A Distributed Localization Hierarchy For An AUV Swarm

Zhuoyuan Song, Student Member, IEEE, and Kamran Mohseni, Member, IEEE

Abstract—Localization of teams of autonomous underwater vehicles (AUVs) still remains as a challenge in large-scale ocean currents. In this study, moving references, drifting under the influence of the ocean background flow, were employed in order to improve the cooperative localization (CL) performance of an AUV swarm in harsh ocean flows. More capable AUVs (dubbed as mother AUVs) with less localization error were utilized as moving references for improving localization error of less capable AUVs (called daughter AUVs). Limitations of a previously proposed modified extended Kalman filter (MEKF) were identified. A particle filter (PF) based algorithm was proposed to address those issues. The performance of the PF algorithm was compared with the MEKF algorithm in several simulated examples including CL in an N-vortex background flow field. Both algorithms can effectively avoid the diverging behavior of localization error in pure CL. The PF algorithm is more robust in choosing the better localized AUV. With a large number of particles, the PF algorithm outperforms the MEKF algorithm at the expense of computational efforts.

I. INTRODUCTION

Due to the fast attenuation of radio frequency (RF) signals in water, the global positioning system (GPS) is not available in underwater areas, which makes localization for autonomous underwater vehicles (AUVs) more difficult [1]. Recent studies have been focused on methods including vision-based simultaneous localization and mapping (SLAM), long baseline (LBL) related techniques, etc. [2]–[4]. However, to the best of our knowledge, in most proposed localization methods, the background flow has been either ignored or treated as small disturbances. In pelagic missions, small AUVs are susceptible to strong ocean currents. They are not capable enough to fight against the background flow to approach localization references and correct their dead-reckoning (DR) errors. Ignoring this fact in the development of localization algorithm for small AUVs leads to failures in localization and missions.

The concept of cooperative localization (CL) using multiple mobile agents has been proposed by Fox et al. to increase the robustness of multi-agent localization through intra-agent measurements and data exchange [5]. In addition, due to the constraint of power supply and the lack of long-term localization references, the coverage of small AUVs is limited, which needs to be remedied by increasing the number of AUVs. Multi-agent cooperation is not only helpful but also necessary in small AUV swarms. In the past decade, CL has been widely applied to applications where GPS is not available including indoor and underwater applications [6]–[9]. However, at least one localization reference is required to bound the localization error of the entire mobile network.

With the capability to model the ocean circulation and simulate the background flow, relationships between optimal trajectories and Lagrangian coherent structures (LCSs) in a background flow have been studied in several investigations [10], [11]. A procedure for trajectory planning based on LCS is outlined by Lipinski and Mohseni in [12]. LCS-based trajectories enable AUVs to navigate as drifters and avoid moving against strong flow for most of their running time, which ensures both navigation accuracy and endurance.

Meanwhile, the rapid growth in ocean modeling and the improvement of computational resources have made it possible to establish operational eddy resolution ocean system that can provide approximately 120 hours’ prediction of the statuses of currents. Real-time oceanographic data collected by low-flying aircrafts, fleets of underwater gliders and AUVs are delivered and assimilated to the ocean model in order to correct and improve the simulation and prediction [13]. Among those spatially changing oceanographic characteristics, the flow velocity is relatively distinguishable among different locations and easy to be measured by AUVs. Before the deployment of AUVs in the ocean, a series of vector field maps with time stamps can be predicted by the ocean model of the area in which AUVs are going to navigate. With these maps downloaded into AUVs before tasks, a vector field based localization method can further help those AUVs navigating along LCS based trajectories and keep bounded localization error.

Our previous studies have shown that the CL performance of an AUV swarm can be improved by utilizing some localization error bounded AUVs as moving references [14], [15]. A modified extended Kalman filter (MEKF) based algorithm was proposed to address the distributed mother-daughter cooperation hierarchy. In this paper, we revisit the problem and propose a particle filter (PF) based algorithm in order to overcome some limitations of MEKF including choosing the best localized DAUV as a sub-optimal reference in CL and further decreasing localization error upper bounds. The rest of the paper is organized as follows. Section II mathematically states the problem and summarizes the MEKF algorithm. Some known limitations are discussed. A particle filter (PF) based algorithm is presented in Section III. In section IV, the validity of the PF algorithm is verified in simulations with the N-vortex flow field on the surface of a
sphere. Under the same simulation settings, we compare the performance of the PF algorithm with the MEKF algorithm. The effect of the number of particles over the localization error bounds is further investigated. Finally, section V concludes the paper and some future research focuses are outlined.

II. MEKF-BASED ALGORITHM

An AUV swarm can be divided into two sub-groups based on their capabilities and functionalities, i.e. mother AUVs (MAUVs) and daughter AUVs (DAUVs) (Fig. 1). We assume that MAUVs’ localization error is bounded. They follow the flow adapted paths to keep their path tracking accuracy, use the background velocity field prediction as navigation maps to further improve their localization and interface with surface vehicles for data up-link and localization error correction. We also assume that all AUVs have two-way communication abilities with neighboring AUVs in their communication scopes. DAUVs are equipped with low-cost inertial measurement units (IMUs) and scope-limited range and bearing sensors but they suffer from DR drifting.

Due to the stochastic nature of noises associated with control inputs and measurements, we represent the problem using the probability theory. The collection of MAUVs’ locations at time $k$ is denoted as $M_k$, where $M_k = \{M_{1,k}, M_{2,k}, \ldots, M_{m,k}\}$ and $m$ is the total number of MAUVs. For DAUV-$i$, whose state at time $k$ is $D_{i,k}$, the collection of all the other DAUVs’ locations at time $k$ is $D_{-i,k} = \{D_n | n = 1, 2, \ldots, d \cap n \neq i\}$ and $d$ is the total number of DAUVs. The state of DAUV-$i$, $D_{i,k}$, can be inferred based on a probabilistic function given the previous state $D_{i,k-1}$ and the control input $u_{k}$, which can be an IMU measurement from time step $k-1$ to $k$. A relative position measurement at time $k$ is denoted as $z_{k}$. The goal is to track the joint probability distribution of all AUVs’ locations given all measurements and control inputs

$$P(D_{i,k}, D_{-i,k}, M_k | Z^k, U^k).$$ \hfill (1)

It has been proved that this distribution can be determined recursively if initial locations of all AUVs are known [14]. By assuming observation independency and the Markov property as well as using Bayes’ rule and the theorem of total probability, the target probability distribution can be calculated through a production of decomposed probability distributions

$$P(D_{i,k}, D_{-i,k}, M_k | Z^k, U^k) = \eta \cdot P(c.u.) \cdot P(c.p.) \cdot P(d.u.) \cdot P(d.p.),$$ \hfill (2)

where $\eta$ is the normalization constant. The original problem is factorized into two sub-problems, i.e. a dynamic simultaneous localization and mapping (DSLAM) problem, where “landmarks” are moving AUVs in this case, and a CL problem. Both sub-problems are recursive. In (2), the CL Prediction is based on

$$P(c.p.) = \int P(D_{-i,k} | D_{i,k-1}) P(D_{i,k-1} | Z^{D,k-1}, U^{k-1}) dD_{i,k-1};$$ \hfill (3)

the CL Updating is based on

$$P(c.u.) = P(z_k^D | D_{i,k}, D_{-i,k});$$ \hfill (4)

the DSLAM Prediction can be completed given

$$P(d.p.) = \int \int P(D_{i,k} | u_k, D_{i,k-1}) P(M_k | M_{k-1}) P(D_{i,k-1} | Z^{M,k-1}, U^{k-1}) dM_{k-1} dD_{i,k-1};$$ \hfill (5)

and the DSLAM Updating is performed based on

$$P(d.u.) = P(z_k^M | D_{i,k}, M_k).$$ \hfill (6)

After the decomposition, recursive structures of both sub-problems are explicit. The two-step DSLAM recursion is very similar to the original SLAM problem with the only difference that localization features in DSLAM are localization error bounded mobile MAUVs instead of static physical features. Nevertheless, the CL problem may introduce issues owing to the fact that this cooperative localization structure is fully distributed. When applying Kalman filter based methods, the updating of covariance matrices need to be carefully addressed since DAUVs are not always in the communication scope of one another. However, out-of-date terms in covariance matrices corresponding to those out-of-scope DAUVs will introduce estimation errors if they are not dealt with properly.

Considering potential issues that may be caused by the distributed structure, we proposed the modified extended Kalman filter (MEKF). The MEKF filters out terms in covariance matrices that are not timely updated due to the unavailability of corresponding DAUVs and exchanges necessary information among AUVs to keep terms needed in the estimation updated correctly. Essential steps used in the MEKF are listed as follows. Interested readers are referred to [15] for more details.

The updating of the vehicle state $\mu$ and covariance matrix $\Sigma$ are performed based on

$$\mu_{k+1} = f(\mu_k, u_{k+1}, \omega_{k+1}),$$ \hfill (7)

$$\Sigma_{k+1} = F_{\mu} \Sigma_{k} F_{\mu}^T + F_\omega Q_{k+1} F_\omega^T,$$ \hfill (8)
where $u$ is the control input, $\omega$ is the associated noise with the covariance $Q$, and $F_\mu$ and $F_\omega$ are control Jacobian matrices. The measurement is corrupted by the noise $\nu$ and is performed given
\[
\tilde{z}_{k+1} = h \left( \mu_{k+1}, \nu_{k+1} \right).
\]
The modified covariance matrix is
\[
\Sigma_{k+1}^{-} = \Xi \Sigma_{k+1}^{\ast} \Xi^T.
\]
$\Xi$ is the selection matrix with diagonal submatrices
\[
\xi_j = \begin{cases} I_{3\times3} & \text{for } n = D_i \cup \{ n \neq D_i \mid r_n \leq S \} \\ 0_{3\times3} & \text{for } \{ n \neq D_i \mid r_n > S \} \end{cases},
\]
where $n \in \{D_1, \ldots, D_d, M_1, \ldots, M_m\}$, and 0 anywhere else. $r_n$ is the Euclidean distance between two DAUVs and $S$ is the communication scope. Finally, the MEKF updating can be carried out using the modified covariance matrix through
\[
K_{k+1} = \Sigma_{k+1}^{-} H_{\mu}^T (H_{\mu} \Sigma_{k+1}^{-} H_{\mu}^T + R_{k+1})^{-1},
\]
\[
\mu_{k+1} = \mu_{k+1} + K_{k+1} [z_{k+1} - h(\mu, 0)],
\]
\[
\Sigma_{k+1} = \Sigma_{k+1}^{-} - K_{k+1} H_{\mu} \Sigma_{k+1}^{-}.
\]
where $K$ is called the Kalman gain, $H_\mu$ is the measurement Jacobian matrix and $R$ is the covariance of the measurement noise.

Since the algorithm is fully distributed and intra-AUV measurements and data exchange take place locally, selection matrices are designed to filter out terms that are not updated. AUVs cannot predict locations of the others. Terms in state vectors and covariance matrices relating to other vehicles are not updated until data exchange takes place when corresponding vehicles are detected. To correctly update the cross-correlation terms in covariance matrices, we introduce the Jacobian multipliers $F_{ji} = F_{x_j}^{k+1} F_{x_j}^{k+1-1} \cdots F_{x_j}^{k+2} F_{x_j}^{k+1}$, which is the collection of Jacobian multipliers used in the motion updating of AUV-$j$ since its last meet with AUV-$i$.

As shown later on, the MEKF is effective in keeping DAUVs’ localization error bounded through utilizing MAUVs as moving references. However, due to inevitable model linearization in performing the MEKF, large linearization error will be an issue when nonlinear vehicles dynamics are considered. Unfortunately, dynamics of AUVs are highly nonlinear, which greatly limits the performance of the MEKF. Meanwhile, when DAUVs are homogeneous with similar noise characteristics, covariance matrices of different DAUVs are very close to each other. When DAUVs perform CL, evaluating the best localized DAUV based on trace of covariance matrices is no longer facile. Without considering better localized DAUVs as sub-optimal references in CL, the performance of the entire localization hierarchy is degraded.

III. PF-BASED ALGORITHM

The PF is one of the Monte Carlo methods [16]. Its fundamental difference from the EKF is that the state probability distribution is represented by a set of “particles” instead of the raw and central moments. As a result, the PF can represent any probability distributions including those that are non-Gaussian, multi-modal. It doesn’t suffer from loss of inaccuracy caused by model linearization or estimation, which makes it suitable for applications with highly nonlinear motion models. Many investigations have been done in regard to solving single-robot localization problems using the PF [17]. It has also been applied to multi-robot localization problems [5], [18], SLAM problems [19] and underwater localization problems [20].

To implement the PF to the proposed problem, we assume known data association, i.e. the correspondence between measurements and identifiers of neighboring AUVs are known. Each DAUV utilizes a set of particles to track its own location estimate. Unlike the MEKF algorithm, in the PF algorithm, DAUVs do not track MAUVs’ location changes through incorporating their Jacobian multipliers during each DSLAM correction phase. As a result, locations of MAUVs can be considered as another set of given parameters, along with the measurement history $Z^k$ and the control input history $U^k$. Moreover, the PF tracks the entire path history of a given DAUV. For any DAUV-$i$, the posterior probability distribution can be expressed as
\[
P(D_i^k, D_{-i}^k | M^k, Z^k, U^k),
\]
in which, again, the superscript $k$ indicates that the corresponding parameter is a collection from time step 1 to time step $k$. Based on the chain rule of conditional probability, this posterior distribution can be calculated through
\[
P(D_i^k, D_{-i}^k | M^k, Z^k, U^k) = P(D_i^k | M^k, Z^k, U^k) P(D_{-i}^k | D_i^k, M^k, Z^k, U^k).
\]
(15)

If locations of DAUV-$i$ and all MAUVs are calculated or estimated, given the measurement history and the control history, paths of the rest of DAUVs can be independently estimated, such that
\[
P(D_{-i}^k | D_i^k, M^k, Z^k, U^k) = \prod_{n=D_i \cup D_i \setminus D_i} (n^k | D_i^k, M^k, Z^k, U^k).
\]
(16)

In (15), the first term can be further decomposed based on the Bayes’ rule to yield
\[
P(D_i^k | M^k, Z^k, U^k) = \frac{P(M^k | D_i^k, Z^k, U^k) P(D_i^k | Z^k, U^k)}{P(M^k | Z^k, U^k)},
\]
(17)
where $P(D_i^k | Z^k, U^k)$ is the posterior probability of DAUV-$i$’s path based on its motion model. We define weights of particles drawn from the particle set of the motion model as
\[
W_{M, D_i} = \frac{P(M^k | D_i^k, Z^k, U^k)}{P(M^k | Z^k, U^k)}.
\]
These weights are determined given the discrepancies between MAUVs’ locations calculated from their motion models, as expressed by the denominator, and estimated locations based on the location of DAUV-$i$ and DAUV-MAUV measurements, represented by the numerator.
In addition, terms in (16) can be rewritten in a similar manner by applying Bayes’ rule and the chain rule to yield

$$P(n^k|D^k_i,M^k,Z^k,U^k) = W_{Mn} \cdot W_{Dn} \cdot P(n^k|Z^k,U^k),$$

(18)

where $W_{Mn}$ and $W_{Dn}$ are weights of particles drawn from the particle set from the motion model of vehicle $n$ and

$$W_{Mn} = \frac{P(M^k | n^k, Z^k, U^k)}{P(M^k | Z^k, U^k)} ,$$

$$W_{Dn} = \frac{P(D^k_i | n^k, M^k, Z^k, U^k)}{P(D^k_i | M^k, Z^k, U^k)} .$$

The completed posterior probability calculation can be expressed as

$$P(D^k_i, M^k, Z^k, U^k) = W_{MDi} \cdot P(D^k_i|Z^k,U^k) \times \prod_{n=D_i}^{D_a} W_{Mn} \cdot W_{Dn} \cdot P(n^k|Z^k,U^k).$$

(19)

By observing (19), one can notice that, for each DAUV, the updating can be accomplished by using one PF. Each particle stores a path estimate and its weight

$$\text{Particle-}i: \{D^k_i, n^k, w^k_{i,D_i}\}.$$ 

Intra-DAUV measurements and DAUV-MAUV measurements are incorporated by assigning different weights to particles. In general, the weight of Particle-$i$ of DAUV-$i$ at time step $k$ is calculated based on

$$w_{k,D_i}^i \propto \frac{1}{\sqrt{2\pi R_k}} \exp \left[ -\frac{1}{2} (z_k - \hat{z}_k)^T R_k^{-1} (z_k - \hat{z}_k) \right] ,$$

where $R_k$ is the covariance of the measurement noise. Algorithm 1 describes the structure of the PF algorithm. During CL correction phases, we choose the DAUV whose particle set has the smallest variance as the best localized neighboring DAUV, which will serve as a sub-optimal reference for other DAUVs. The weighted mean of state of the best localized DAUV is transmitted to other DAUVs for their calculation of particle weights. To avoid large number of particles with small weights, a systematic resampling scheme (line 19-23) is applied after each measurement when the effective sample size (ESS) drops below a certain percentage of the number of particles $N$, where

$$\text{ESS} = \frac{N}{1 + c^2_v} \quad \text{and} \quad c^2_v = \frac{1}{N} \sum_{i=1}^{N} (w^i - 1)^2 .$$

(20)

IV. SIMULATIONS AND RESULTS

The flow generated by vortices on a sphere is widely used in geophysical fluid dynamics when considering large-scale atmospheric or oceanographic flows with coherent structures that persist over a long period of time and move over large distances. This flow pattern is applied as a simplified model of global ocean currents by ignoring interactions with lands [21]. In this model, ocean vortices are simulated by moving point vortices. The resulting flow pattern can be used in meteorological studies including hurricane simulations and ocean current simulations [22]. In this work, we focus on a special solution discussed by Kidambi and Newton in [23], where three vortices are used for simplicity but without loss of generality. We nondimensionalize the problem with respect to the radius of Earth, $R = 6.371 \times 10^6 \text{ m}$ and the circulation $\Gamma = 8 \times 10^6 \text{ m}^2/\text{s}$. If a fully passive AUV is placed at the north pole of the sphere, the resulting trajectory under the impact of the background flow is shown in Fig. 2.

Initial locations of all DAUVs are chosen randomly on the sphere. The measurement frequency is selected based on $T_H/T_M = 33.3$, where $T_H$ is the hydrodynamic time-scale, which can be determine from the length-scale and the circulation-scale, and $T_M$ is the measurement time-scale. By ignoring the dynamics of vehicles, we make AUVs in our simulations to be fully flow driven for simplicity. In the first simulation, the MEKF algorithm runs for 1 normalized time period. The motion updating period is set to 0.0005 and DAUVs take measurements every 10 time steps. Figure 3a shows the localization error of each of the three DAUVs when there is no MAUV and intra-AUV interaction is disabled. DR error keeps increasing as time evolves and DAUVs will finally become lost. The performance of CL is
The feasibility of improving the performance of cooperative localization by utilizing localization error bounded references through a mother-daughter cooperative underwater localization hierarchy was revisited. We first outlined a previously proposed probabilistic formulation of the problem based on the modified extended Kalman filter (MEKF). In order to fit the distributed structure, the selection matrix and the Jacobian multiplier were introduced to ensure the correctness of intra-AUV data fusion. Some limitations of the MEKF algorithm in applications with highly nonlinear dynamics and homogeneous vehicle groups were identified.

V. CONCLUSIONS

Although the PF algorithm achieves similar performance as the MEKF algorithm, one may have noticed that the latter outperforms the former in terms of localization error upper bounds, which is because that the number of particles are fairly small to capture correct vehicle locations. As we increase the number of particles to 50 and 100 (Fig. 5), the PF algorithm yields similar or even better results than the MEKF algorithm. In general, it’s a trade-off between desired localization error bounds and the computational cost. In the MEKF algorithm, each DAUV performs estimation correction with no more than one neighboring AUV, either MAUV or DAUV, and each DAUV partially tracks the status of all the other AUVs, which makes its computational complexity roughly proportional to $d^2$. In the PF algorithm, since each particle needs to be updated and corrected with no more than one neighboring AUV, the computation complexity is proportional to $Nd$. As a result, as the number of DAUVs increases, the PF algorithm can generate better performance with similar computational effort as the MEKF algorithm.
A particle filter (PF) based algorithm was proposed in this paper to address these limitations. The original target probability distribution was modified and further decomposed in order to apply the PF algorithm. The performance of the PF algorithm was compared with the MEKF algorithm in several simulated examples including CL in an N-vortex background flow field. Simulation results showed that the PF algorithm yielded similar performance as the MEKF algorithm when the number of particles for each DAUV was properly chosen. A larger number of particles resulted in better results at the expense of computational complexity. As the number of DAUVs increases, the amount of additional computation introduced by the PF algorithm compared to the MEKF algorithm decreases and better performance can be expected by using the PF algorithm.

Incorporating the background velocity measurement in localization of MAUVs needs further investigations. The dynamics of AUVs in the background flow will be further studied and included in our localization algorithm. Experiments are needed to validate both algorithms and identify further limitations.

REFERENCES