Extended Observation-Cooperative SLAM for Unmanned Surface Vehicles

Gayas Asaad, Alireza Babaei, Mohammad H. Ferdowsi, Hossein Bolandi

Abstract— This paper presents a cooperative approach to perform simultaneous localization and mapping (SLAM) algorithm based on additional, indirect observations of the features (landmarks). These additional observations are made with helping of collaborating vehicles by sharing their own observations of the features, which extends the observed area for each vehicle, so the proposed approach is called the Extended Observation-Cooperative SLAM (EO-CSLAM) algorithm. For implementing this approach, the paper adopts extended Kalman filter-simultaneous localization and mapping (EKF-SLAM) solution, expanding it to the cooperative case and to be applicable with unmanned surface vehicles (USVs) in marine environments. The proposed algorithm is first elucidated mathematically verifying its efficiency, and then, the performance gain is evaluated using simulations conducted for USVs with radar sensors. Simulation results show that the EO-CSLAM provides noticeable improvement versus the single-vehicle SLAM (or Mono-SLAM) in terms of localization accuracy and mapping performance. The extended observation (EO) principle allows implementing this approach with SLAM solutions other than EKF method, and in other application fields.

Index Terms— Autonomous Navigation, Cooperative SLAM, Unmanned Surface Vehicles.

1 INTRODUCTION

In the last two decades, unmanned surface vehicles (USVs) have seen growing spread in a wide spectrum of missions, including harbor surveillance, underwater mapping, water quality detection, and military tasks. In such applications, exploiting multiple vehicles in a cooperative manner provides more efficiency in terms of accuracy, surveillance area, sharing resources, reliability, flexibility and faster achievement, especially in unknown, unstructured, and large-scale surroundings such as marine environments.

Whatever the task of the USV as an autonomous vehicle is, the accurate localization has an essential role for efficient achievement and accurate results. For instance, in underwater environmental modelling, accurate navigation is very important to provide sensors with referential transformations matrix and high-accuracy pose [1]. On the other hand, while GPS can be used for localization, its data can be inaccurate or inaccessible due to many possible reasons, such as atmospheric changes, noisy environments, multi-path errors, deliberate jamming, spoofing or confined areas where observing sufficient number of satellites can be difficult [2]. To solve these problems, simultaneous localization and mapping (SLAM) framework can be a proper alternative or incorporated to GPS [3]. Moreover, exploiting SLAM in a cooperative approach can provide more improvement in localization accuracy of USVs, in addition to the mentioned advantages of cooperative manners.

- *Gayas Asaad is currently a doctoral student in Department of Electrical Engineering, Malek ashtar University of Technology, Iran. E-mail: gayjaw@yahoo.com*
- *Dr. Alireza Babaei, Department of Aerospace, Amirkabir University of Technology, Iran. E-mail: arbabaei@aut.ac.ir*
- *Dr. Mohammad H. Ferdowsi, Department of Electrical Engineering, Malek ashtar University of Technology, Iran. E-mail: ferdowsi@mut.ac.ir*
- *Dr. Hossein Bolandi, School of Electrical Engineering, Iran University of Science and Technology. . E-mail: h_bolandi@iust.ac.ir*

The SLAM algorithm has been used with a single USV (Mono-SLAM) in various applications using various sensors. Kalyan *et al.* utilized a USV with an imaging payload in the form of underwater blazed array sonar within EKF-SLAM framework [4]. University of Michigan designed a USV with a 3D hybrid camera-LIDAR vision system [5]; it has been shown that feature-based EKF-SLAM algorithm is really suitable for USVs because of the sparsity of obstacles on the water plane. Leedekerken *et al.* examined concurrent mapping above and below the water in large scale marine environments via SLAM [2]; their method allowed addressing the technical difficulties of GPS privation. The USV was equipped with sonar, LIDAR, camera, and radar. Radar sensor had been used in EKF-SLAM scenario [3] in a main work presenting a fundamental solution to SLAM problem. Furthermore, a notable research, using X-band marine radar for SLAM with a single USV, has been done by Mullane *et al.* [6] as a precious contribution to radar based navigation through a feature-based SLAM framework. The point features have been extracted through an automated extraction routine based on a probabilistic landmark detector.

For the cooperative SLAM implementation with USVs, a Multi-USV-based CSLAM approach has been proposed in [7] using laser sensors. The research adopted the Constrained Local Submap Filter (CLSF) approach, which had been presented in [8] and [9] to improve computational efficiency and data association. In CLSF approach, local submaps are fused periodically into a single global map; the common (duplicated) feature estimates are processed by a constraining operation as a weighted projection to produce a recovered estimate for each common feature. Therefore, in the CLSF-based cooperative SLAM, the performance gain (versus the single-USV case) depends only on the common features between the local submaps, while the non-common features do not contribute to the improvement due to the uncorrelated nature of local submaps. Consequently, in the case of absence

of common features in the overlapped areas, which is a possible situation in large-scale environments, there will not be improvement in localization accuracy and mapping performance, and CLSF-based CSLAM will act the same level of accuracy of Mono-SLAM or lower.

This paper proposes a cooperative SLAM approach which allows vehicles to improve localization accuracy and mapping performance even with no common features, profiting from all observed features, which makes it a proper method for USVs using radar sensors. In the proposed approach, called Extended Observation-Cooperative SLAM (EO-CSLAM), the vehicles share observations and control data, and then, use the shared information together with vehicle-vehicle (v-v) observations to obtain additional correlated observations, which improves vehicle and map estimates. This improvement, in addition to the general benefits usually gained from cooperative frameworks, makes the proposed approach a proper choice for several tasks, such as harbour surveillance, exploration, environmental security etc, besides the extended observation (EO) principle allows implementing this approach with SLAM solutions other than EKF method.

This paper is organized as follows: Section 2 defines the general framework for the EO-CSLAM approach, provides theoretical explanation of the algorithm adopting EKF-SLAM method, and proves the algorithm mathematically, while section 3 evaluates its utility and performance gain using simulations. Finally, conclusions are presented in section 4.

2 Design of the EO-CSLAM algorithm

In the main structure of the SLAM problem, the correlation between features (or landmarks) plays an essential role in the state estimate convergence. This correlation is a result of the common error in estimated vehicle location [3], where features' estimates depend on the observations made by the vehicle from its location. Moreover, the update steps depend on the observation model which is, in turn, a function of estimated vehicle location and features estimates. In other words, the necessary correlation between features stems from the observation process. Consequently, if a vehicle gets an estimate of a feature observed by another vehicle, this estimate will be uncorrelated with its own features estimates, and if this estimated feature is non-common between the two vehicles, it cannot improve the estimates of the vehicle location and other features. For instance, Fig. 1 shows three features as samples of different situations: the common feature $\mathbf{f}_0$ places in the intersection area of the two fields of view (FOV) of vehicles sensors, the feature $\mathbf{f}_1$ is visible only for the vehicle b, and the feature $\mathbf{f}_2$ is visible only for the vehicle a. As the two features, $\mathbf{f}_0$ and $\mathbf{f}_2$, are observed by the same vehicle (a), their two location estimates $\hat{\mathbf{f}}_0^a$ and $\hat{\mathbf{f}}_2^a$, respectively, are correlated with each other and with the location of vehicle a, while both of them (i.e. $\hat{\mathbf{f}}_0^a$ and $\hat{\mathbf{f}}_2^a$) are independent from $\hat{\mathbf{f}}_0^b$, $\hat{\mathbf{f}}_2^b$ and b. So, if the estimate $\hat{\mathbf{f}}_2^a$ (generated by a) is given to b, it can be added to its map but cannot improve localization accuracy of vehicle b or its map because there is no correlation between $\hat{\mathbf{f}}_2^a$ and vehicle b, in addition to that the feature $\mathbf{f}_2$ is non-common between a and b. In such a case, if a cooperative SLAM is performed via CLSF method, the improvement in localization accuracy of a and b is obtained by consolidating the two estimates, $\hat{\mathbf{f}}_0^a$ and $\hat{\mathbf{f}}_0^b$, into a single estimate of the common feature $\mathbf{f}_0$ to which both estimates belong. So, if a feature such as $\mathbf{f}_0$ were not found, localization accuracy of vehicles would be similar to that of Mono-SLAM.

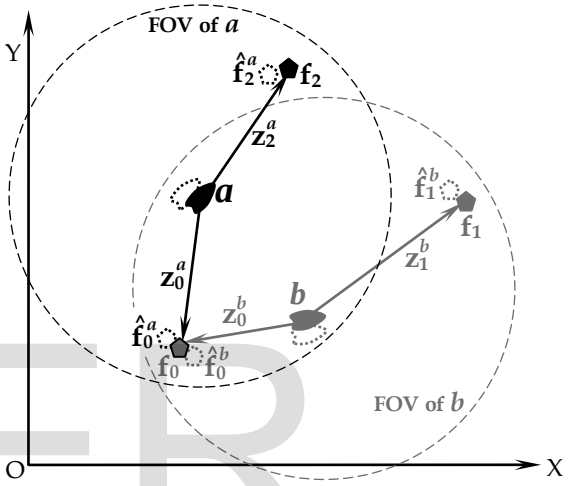

Fig. 1: Two vehicles (a, b) performing SLAM independently, the estimates $\hat{\mathbf{f}}_0^a$ and $\hat{\mathbf{f}}_0^b$ are uncorrelated even that they belong to the same feature $\mathbf{f}_0$.

This section describes a cooperative approach which allows vehicles to profit from feature observations of each other to improve map and vehicle estimates even if those features are non-common. In other words, the proposed approach allows a and b to profit from observations of non-common features, like $\mathbf{f}_1$ and $\mathbf{f}_2$, to improve localization accuracy and mapping performance with or without common features like $\mathbf{f}_0$.

Subsection 2.1 describes the framework in which the proposed algorithm (EO-CSLAM) will be performed. Subsection 2.2 provides the theoretical foundation of the algorithm, while subsection 2.3 mathematically proves the algorithm convergence.

2.1 EO-CSLAM framework

To consider a general framework for the EO-CSLAM algorithm, it is assumed that a team of vehicles perform a collective task in an unknown environment. While moving through the environment, the vehicles use their sensors to obtain relative observations (measurements) of the features and vehicles within their fields of view (FOV). Assuming that the collaborating vehicles have the ability to share required information, each vehicle shares its observations and control signals with the collaborating vehicles.

For the marine environments case considered in this paper, the collaborating vehicles are a team of USVs with radar sensors. To demonstrate the algorithm, let's consider the case of two USVs denoted a and b as shown pictorially in Fig.2 involving only three features for simplifying. The features on sea surface can be artificial and/or naturally occurring elements [6], [9].Considering the surface planar motion, the start points of vehicles are firstly stored as initial positions with respect to a single global reference frame (XOY). Each vehicle initializes its global map in this frame with zero initial position uncertainty and then, continues (through the unknown environment) observing the features and vehicles in its FOV. The relative observations and control data are shared between the vehicles; the shared features' observations are used with the vehicle-vehicle (v-v) observations to generate additional correlated observations, such as $\vec{\mathbf{z}}_1^{ab} = \vec{\mathbf{z}}_b^a + \vec{\mathbf{z}}_1^b$ in Fig.2, while the shared control data are used by each USV to estimate and update the second vehicle's location. The additional correlated observations are called extended observations (EO). Next subsection explains the details of the EO-CSLAM algorithm.

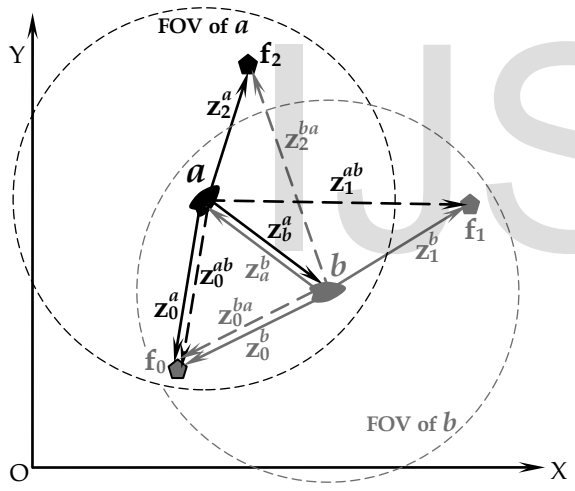

Fig. 2: EO-CSLAM by two USVs (a, b), solid arrows refer to local observations (of features in the FOV) while dashed arrows refer to extended observations, such as $\mathbf{z}_1^{ab}$.

2.2 EO-CSLAM algorithm formulation

First, the main quantities and their used symbols are defined together with motion and observation models.

The true global pose of a vehicle v $(v = a,b)$ is denoted $\mathbf{p}_k^v = [\mathbf{q}_k^{vT}, \phi_k^v]^T$ (Fig. 3) involving its position $\mathbf{q}_k^v = [x_k^v, y_k^v]^T$ and heading ϕ_k^v at the kth instant $t_k = k.T_c$, where T_c denotes the time interval between control signals and $[\cdot]^T$ refers to transpose, so the k th true global pose of a is:

$$\mathbf{p}_k^a = [\mathbf{q}_k^{aT}, \phi_k^a]^T = [x_k^a, y_k^a, \phi_k^a]^T \quad (1)$$

The true global position of the l^{th} feature is denoted $\mathbf{f}_l = [x_l, y_l]^T$, and its estimate generated by vehicle v at the t_k instant is $\hat{\mathbf{f}}_{l,k}^v = [\hat{x}_{l,k}^v, \hat{y}_{l,k}^v]^T$. If $\mathbf{m}_k^v$ denotes the true map of a vehicle v up to t_k, the generated (or estimated) map will be $\hat{\mathbf{m}}_k^v = [\hat{m}_{1,k}^{vT}, \hat{m}_{2,k}^{vT}, .., \hat{m}_{Nv,k}^{vT}]^T$ containing Nv feature estimates (e.g. Na for vehicle a). In fact, each vehicle generates its own map in a different order according to the sequence of observed features, i.e., when the feature $\mathbf{f}_l$ is observed by vehicle v for the first time, its estimate $\hat{m}_{i,k}^v \equiv \hat{\mathbf{f}}_{l,k}^v$ will be added to $\hat{\mathbf{m}}_k^v$, and will be the i^{th} estimate in this map, where $i = Nv+1$ and in general $i \neq l$. The vector $\hat{\mathbf{s}}_k^v = [\hat{\mathbf{p}}_k^{vT}, \hat{\mathbf{m}}_k^{vT}]^T$ represents the estimate of the true state vector $\mathbf{s}_k^v = [\mathbf{p}_k^{vT}, \mathbf{m}_k^{vT}]^T$ with covariance $\mathbf{C}_k^v$. This state covariance comprises: $\mathbf{C}_{ppk}^v$ the covariance matrix of $\hat{\mathbf{p}}_k^v$, $\mathbf{C}_{MMk}^v$ the covariance matrix of $\hat{\mathbf{m}}_k^v$, and $\mathbf{C}_{pMk}^v$ the cross-covariance between v and $\mathbf{m}_k^v$ as follows:

$$\mathbf{C}_k^v = \begin{bmatrix} \mathbf{C}_{ppk}^v & \mathbf{C}_{pMk}^v \\ \mathbf{C}_{pMk}^{vT} & \mathbf{C}_{MMk}^v \end{bmatrix} \quad (2)$$

$$\mathbf{C}_{ppk}^v = \begin{bmatrix} \mathbf{C}_{qqk}^v & \mathbf{C}_{q\phi k}^v \\ \mathbf{C}_{q\phi k}^{vT} & \sigma_{\phi,k}^{2,v} \end{bmatrix} \quad (3)$$

where $\mathbf{C}_{qqk}^v$ is the covariance matrix of $\hat{\mathbf{q}}_k^v$, and $\sigma_{\phi,k}^{2,v}$ the heading variance of v.

The motion model of the SLAM is usually written as:

$$\mathbf{s}_{k+1}^v = f(\mathbf{s}_k^v, \mathbf{u}_k^v, \mathbf{w}_k^v) \quad (4)$$

where $\mathbf{u}_k^v$ denotes the control vector applied at time t_k to drive the vehicle to state $\mathbf{s}_{k+1}^v$ at time t_{k+1}, and $\mathbf{w}_k^v$ is zero mean uncorrelated Gaussian motion disturbances with covariance $\mathbf{Q}_k^v$. The observation model:

$$\mathbf{z}_{ik}^v = [r_{ik}^v, \theta_{ik}^v]^T = h(\mathbf{s}_k^v, \mathbf{m}_k^v) + \mathbf{v}_k \quad (5a)$$

where $\mathbf{v}_k = [n_k^r, n_k^\theta]^T \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ is a vector of additive zero mean uncorrelated Gaussian observation errors (n_k^r and n_k^θ) in the feature range and bearing (respectively), see Fig.3. Thus:

$$\mathbf{z}_{ik}^v = \begin{bmatrix} r_{ik}^v \\ \theta_{ik}^v \end{bmatrix} = \begin{bmatrix} \sqrt{(x_l - x_k^v)^2 + (y_l - y_k^v)^2} \\ \tan^{-1}\left(\dfrac{y_l - y_k^v}{x_l - x_k^v}\right) - \phi_k^v \end{bmatrix} + \begin{bmatrix} n_k^r \\ n_k^r \end{bmatrix} \quad (5b)$$

Now, EKF steps of EO-CSLAM are performed as follows (considering the vehicle a as an example):

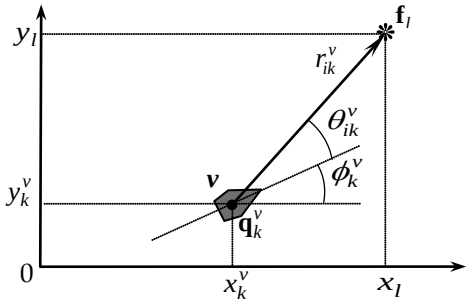

Fig. 3: Vehicle v is observing feature $\mathbf{f}_l$ which represents the i^{th} feature in its map.

Initialization:

$$\hat{\mathbf{s}}_0^{a+} = \hat{\mathbf{p}}_0^{a+} = [\hat{x}_0^{a+}, \hat{y}_0^{a+}, \hat{\phi}_0^{a+}]^T \tag{6}$$

with covariance $\mathbf{C}_0^{a+} = \mathbf{C}_{pp0}^{a+}$, where no features have been mapped yet ($\mathbf{m}_0^a = \emptyset$). The vehicle a stores the initial data of both vehicles, i.e. ($\hat{\mathbf{s}}_0^{a+}$, $\mathbf{C}_0^{a+}$) and ($\hat{\mathbf{s}}_0^{b+}$, $\mathbf{C}_0^{b+}$). In the following, the notation $(\cdot)_k^-$ refers to predicted values, while $(\cdot)_k^+$ refers to updated values, for $k = 1,2,3...$

Prediction:

$$\hat{\mathbf{s}}_k^{a-} = f(\hat{\mathbf{s}}_{k-1}^{a+}, \mathbf{u}_{k-1}^a, \mathbf{0}) \tag{7}$$

$$\mathbf{C}_k^{a-} = \mathbf{F}_{k-1}^a \mathbf{C}_{k-1}^{a+} \mathbf{F}_{k-1}^{aT} + \mathbf{L}_{k-1}^a \mathbf{Q}_{k-1}^a \mathbf{L}_{k-1}^{aT} \tag{8}$$

where $\mathbf{F}_{k-1}^a$ and $\mathbf{L}_{k-1}^a$ are the Jacobeans of motion model [10]:

$$\mathbf{F}_{k-1}^a = \partial f / \partial \mathbf{s}\big|_{\hat{\mathbf{s}}_{k-1}^{a+}} \tag{9}$$

$$\mathbf{L}_{k-1}^a = \partial f / \partial \mathbf{w}\big|_{\hat{\mathbf{s}}_{k-1}^{a+}} \tag{10}$$

As control data $\mathbf{u}_{k-1}^v$ are shared, and having ($\hat{\mathbf{s}}_{k-1}^{b+}$, $\mathbf{C}_{k-1}^{b+}$), the vehicle a obtains $\hat{\mathbf{s}}_k^{b-}$ and $\mathbf{C}_k^{b-}$ of b using (7) and (8), so the v-v observation $\mathbf{z}_b^a$ is distinguished via data association (being assigned to $\hat{\mathbf{q}}_k^{b-}$ of $\hat{\mathbf{s}}_k^{b-}$), and the shared local observations $\mathbf{z}_{ik}^v$ are used to obtain extended observations as follows (dropping the time index k for simplifying):

For each feature $\mathbf{f}_l = [x_l, y_l]^T$ observed by b, see Fig. 4, the vehicle b transmits the local observation $\mathbf{z}_j^b = [r_j^b, \theta_j^b]^T$ to a, the vehicle a now computes the extended observation of $\mathbf{f}_l$ using an extended observation model h_e as follows:

$$\mathbf{z}_j^{ab} = [r_e, \theta_e]^T = h_e(\mathbf{z}_b^a, \phi^a, \mathbf{z}_j^b, \phi^b) \tag{11}$$

where $\mathbf{z}_b^a = [r_a, \theta_a]^T$ is the v-v observation of b by a, thus

$$\mathbf{z}_j^{ab} = h_e(\mathbf{z}_b^a, \phi^a, \mathbf{z}_j^b, \phi^b) = \begin{bmatrix} r_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \sqrt{(dx)^2 + (dy)^2} \\ \tan^{-1}(dy/dx) - \phi^a \end{bmatrix} \tag{12}$$

where $dx = \tilde{x}_a + \tilde{x}_j = r_a.c_a + r_j^b.c_j$

$dy = \tilde{y}_a + \tilde{y}_j = r_a.s_a + r_j^b.s_j$

and $c_a = \cos(\phi^a + \theta_a)$, $s_a = \sin(\phi^a + \theta_a)$

$c_j = \cos(\phi^b + \theta_j^b)$, $s_j = \sin(\phi^b + \theta_j^b)$

As the true values of headings ϕ^a and ϕ^b are not available, their estimated values are used in (11), where the estimate $\hat{\phi}^{b-}$ and its variance $\sigma_\phi^{2,b-}$ are extracted from $\hat{\mathbf{s}}^{b-}$ and $\mathbf{C}^{b-}$ respectively:

$$\mathbf{z}_j^{ab} = [r_e, \theta_e]^T = h_e(\mathbf{z}_b^a, \hat{\phi}^{a-}, \mathbf{z}_j^b, \hat{\phi}^{b-}) \tag{13}$$

However, extended observation error covariance $\mathbf{R}_j^e$ for each $\mathbf{f}_l$ is not the same as $\mathbf{R}$ (the main observation error covariance of $\mathbf{z}_b^a$ and $\mathbf{z}_j^b$), so the covariance matrix $\mathbf{R}_j^e$ of $\mathbf{z}_j^{ab}$ can be obtained based on (11) as follows:

$$\mathbf{R}_j^e = \mathbf{J}_a \mathbf{R} \mathbf{J}_a^T + \mathbf{J}_\phi^a \sigma_{\phi,k}^{2,a} \mathbf{J}_\phi^{aT} + \mathbf{J}_j \mathbf{R} \mathbf{J}_j^T + \mathbf{J}_\phi^b \sigma_{\phi,k}^{2,b} \mathbf{J}_\phi^{bT} \tag{14}$$

where $\mathbf{J}_\times^\times$, the Jacobeans of h_e, are obtained from (12) as follows:

$$\mathbf{J}_a = \frac{\partial h_e}{\partial \mathbf{z}_b^a} = \begin{bmatrix} \partial r_e/\partial r_a & \partial r_e/\partial \theta_a \\ \partial \theta_e/\partial r_a & \partial \theta_e/\partial \theta_a \end{bmatrix} = \frac{1}{r_e} \begin{bmatrix} c_a.dx + s_a.dy & r_a(c_a.dy - s_a.dx) \\ (s_a.dx - c_a.dy)/r_e & r_a(c_a.dx + s_a.dy)/r_e \end{bmatrix} \tag{15a}$$

$$\mathbf{J}_\phi^a = \frac{\partial h_e}{\partial \phi^a} = \begin{bmatrix} \partial r_e/\partial \phi^a \\ \partial \theta_e/\partial \phi^a \end{bmatrix} = \begin{bmatrix} r_a(c_a.dy - s_a.dx)/r_e \\ [r_a(c_a.dx + s_a.dy)/r_e^2] - 1 \end{bmatrix} \tag{15b}$$

$$\mathbf{J}_j = \frac{\partial h_e}{\partial \mathbf{z}_j^b} = \begin{bmatrix} \partial r_e/\partial r_j^b & \partial r_e/\partial \theta_j^b \\ \partial \theta_e/\partial r_j^b & \partial \theta_e/\partial \theta_j^b \end{bmatrix} = \frac{1}{r_e} \begin{bmatrix} c_j.dx + s_j.dy & r_j^b(c_j.dy - s_j.dx) \\ (s_j.dx - c_j.dy)/r_e & r_j^b(c_j.dx + s_j.dy)/r_e \end{bmatrix} \tag{15c}$$

$$\mathbf{J}_\phi^b = \frac{\partial h_e}{\partial \phi^b} = \begin{bmatrix} \partial r_e/\partial \phi^b \\ \partial \theta_e/\partial \phi^b \end{bmatrix} = \begin{bmatrix} r_j^b(c_j.dy - s_j.dx)/r_e \\ r_j^b(c_j.dx + s_j.dy)/r_e^2 \end{bmatrix} \tag{15d}$$

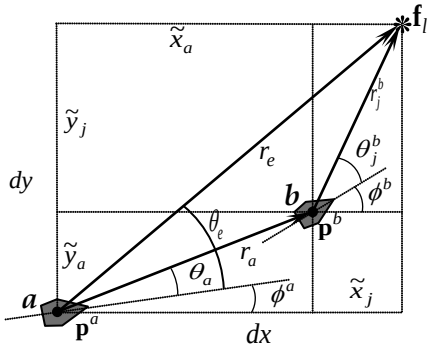

Fig. 4: Extended observation of $\mathbf{f}_l$ by a

Therefore, in addition to its main local observations, such as $\mathbf{z}_0^a$ and $\mathbf{z}_2^a$ in Fig. 2, the vehicle a can obtain two types of extra observations, additional observations of the common

features, like $\mathbf{z}_0^{ab}$ of $\mathbf{f}_0$, and observations of the out-of-FOV features, like $\mathbf{z}_1^{ab}$ of $\mathbf{f}_1$.

For all these types of observation, data association is performed: for each observation associated with previous mapped feature, the innovation $\boldsymbol{\nu}_{ik}^a$ and its covariance $\mathbf{E}_{ik}$ are computed as follows:

For local observations $\mathbf{z}_{ik}^a$:

$$\boldsymbol{\nu}_{ik}^a = \mathbf{z}_{ik}^a - \hat{\mathbf{z}}_{ik}^a = \mathbf{z}_{ik}^a - h(\hat{\mathbf{s}}_k^{a-}, \hat{\mathbf{m}}_k^a) \tag{16a}$$

$$\mathbf{E}_{ik} = \mathbf{H}_{ik}^a \mathbf{C}_k^{a-} \mathbf{H}_{ik}^{aT} + \mathbf{R}_k \tag{17a}$$

For extended observations $\mathbf{z}_{ik}^{ab}$:

$$\boldsymbol{\nu}_{ik}^a = \mathbf{z}_{ik}^{ab} - \hat{\mathbf{z}}_{ik}^{ab} = \mathbf{z}_{ik}^{ab} - h(\hat{\mathbf{s}}_k^{a-}, \hat{\mathbf{m}}_k^a) \tag{16b}$$

$$\mathbf{E}_{ik} = \mathbf{H}_{ik}^a \mathbf{C}_k^{a-} \mathbf{H}_{ik}^{aT} + \mathbf{R}_{ik}^e \tag{17b}$$

where $\mathbf{H}_{ik}^a = \partial h / \partial \mathbf{s}$ is the Jacobean of the observation model h:

$$\mathbf{H}_{ik}^a = \left[\left.\frac{\partial h}{\partial \mathbf{p}}\right|_{\hat{\mathbf{p}}_k^{a-}} \mathbf{0}_1 \;..\; \mathbf{0}_{i-1} \;\; \left.\frac{\partial h}{\partial \mathbf{f}}\right|_{\hat{\mathbf{f}}_{ik}^a} \mathbf{0}_{i+1} \;..\; \mathbf{0}_{Nak} \right] \tag{18}$$

Kalman gain:

$$\mathbf{K}_{ik}^a = \mathbf{C}_k^{a-} \mathbf{H}_{ik}^{aT} \mathbf{E}_{ik}^{-1} \tag{19}$$

Update:

$$\hat{\mathbf{s}}_k^{a+} = \hat{\mathbf{s}}_k^{a-} + \mathbf{K}_{ik}^a \boldsymbol{\nu}_{ik}^a \tag{20}$$

$$\mathbf{C}_k^{a+} = \mathbf{C}_k^{a-} - \mathbf{K}_{ik}^a \mathbf{E}_{ik} \mathbf{K}_{ik}^{vT} = (\mathbf{I} - \mathbf{K}_{ik}^a \mathbf{H}_{ik}^a) \mathbf{C}_k^{a-} \tag{21}$$

while for each observation belongs to a new observed feature, the location estimate $\hat{\mathbf{f}}_{ik}^a$ is computed as follows:

$$\hat{\mathbf{f}}_{ik}^a = g(\hat{\mathbf{p}}_k^{a-}, \mathbf{z}_{ik}) = \begin{bmatrix} \hat{x}_k^{a-} + r_{ik}.\cos(\hat{\phi}_k^{a-} + \theta_{ik}) \\ \hat{y}_k^{a-} + r_{ik}.\sin(\hat{\phi}_k^{a-} + \theta_{ik}) \end{bmatrix} \tag{22}$$

where $\mathbf{z}_{ik} = [r_{ik}, \theta_{ik}]^T$ represents $\mathbf{z}_{ik}^a$ *or* $\mathbf{z}_{ik}^{ab}$, and then, the new estimate is combined to $\hat{\mathbf{m}}_k^a$ within $\hat{\mathbf{s}}_k^{a-}$, while $\mathbf{C}_k^{a-}$ is augmented, thus

$$\hat{\mathbf{s}}_k^{a+} = \left[(\hat{\mathbf{p}}_k^{a-})^T \;,\; \hat{\mathbf{m}}_k^{aT} \;,\; \hat{\mathbf{f}}_{ik}^{aT} \right]^T \tag{23a}$$

$$\mathbf{C}_k^{a+} = \begin{bmatrix} \mathbf{C}_{ppk}^{a-} & \mathbf{C}_{pMk}^{a-} & (\mathbf{J}_p^a \mathbf{C}_{ppk}^{a-})^T \\ \mathbf{C}_{pMk}^{a-\,T} & \mathbf{C}_{MMk}^{a-} & (\mathbf{J}_p^a \mathbf{C}_{pMk}^{a-})^T \\ \mathbf{J}_p^a \mathbf{C}_{ppk}^{a-} & \mathbf{J}_p^a \mathbf{C}_{pMk}^{a-} & \mathbf{C}_{iik}^{a+} \end{bmatrix} \tag{23b}$$

where

$$\mathbf{J}_p^a = \frac{\partial g}{\partial \mathbf{p}_k^a} = \begin{bmatrix} 1 & 0 & -r_{ik}.\sin(\phi_k^a + \theta_{ik}) \\ 0 & 1 & r_{ik}.\cos(\phi_k^a + \theta_{ik}) \end{bmatrix} \tag{24}$$

and $\mathbf{C}_{iik}^{a+}$ is computed for local observations using $\mathbf{R}_k$:

$$\mathbf{C}_{iik}^{a+} = \mathbf{J}_p^a \mathbf{C}_{ppk}^{a-} \mathbf{J}_p^{aT} + \mathbf{J}_z^a \mathbf{R}_k \mathbf{J}_z^{aT} \tag{25a}$$

and using $\mathbf{R}_{ik}^e$ for extended observations

$$\mathbf{C}_{iik}^{a+} = \mathbf{J}_p^a \mathbf{C}_{ppk}^{a-} \mathbf{J}_p^{aT} + \mathbf{J}_z^a \mathbf{R}_{ik}^e \mathbf{J}_z^{aT} \tag{25b}$$

where

$$\mathbf{J}_z^a = \frac{\partial g}{\partial \mathbf{z}_{ik}} = \begin{bmatrix} \cos(\phi_k^a + \theta_{ik}) & -r_{ik}.\sin(\phi_k^a + \theta_{ik}) \\ \sin(\phi_k^a + \theta_{ik}) & r_{ik}.\cos(\phi_k^a + \theta_{ik}) \end{bmatrix} \tag{26}$$

and so, the next period $(k+1)$ starts with (7) and (8) continuing with the same manner.

2.3 Proof of the EO-CSLAM algorithm

This subsection firstly explains the correlated nature of the feature estimates generated from extended observations via EO-CSLAM algorithm, and shows, in contrast, the uncorrelated nature of the estimates when they are taken from other vehicles, and then, proves the convergence of the proposed algorithm.

Considering equation (23b) of the state covariance when augmenting a new feature estimate, it can be seen that the correlation (between the new estimate and both the vehicle and previous mapped features) depends on the Jacobean $\mathbf{J}_p^a$ of the function $g(\mathbf{p}_k^a, \mathbf{z}_{ik})$. This function is actually the inverse function of the observation model, i.e. $g(\mathbf{p}_k^a, \mathbf{z}_{ik}) = h^{-1}(\mathbf{s}_k^v, \mathbf{m}_k^v)$, see Fig. 3 and equations (5a,5b, 22). In fact, g transforms the polar form of $\mathbf{z}_{ik}$ to an estimate $\hat{\mathbf{f}}_{ik}^a$ in a Cartesian form, while h works vice versa, that is, it converts the estimate $\hat{\mathbf{f}}_{ik}^a \supset \hat{\mathbf{m}}_k^v$ to an estimated observation $\hat{\mathbf{z}}_{ik}$ in the polar form. This explains what previously mentioned: the necessary correlation (between features) stems from the observation process. Thus, since the function g is applied according to (22) on both the observation types, local ($\mathbf{z}_{ik}^a$) and extended ($\mathbf{z}_{ik}^{ab}$), the feature estimates resulting from extended observations will be also correlated as those from the local. While, on the other hand, assuming that the vehicle a gets a ready estimate $\hat{\mathbf{f}}_{jk}^b$ (generated by the vehicle b) with its covariance $\mathbf{C}_{jjk}^{b+}$, the augmentation process (23 a, b) will be done as follows:

$$\hat{\mathbf{s}}_k^{a+} = [\hat{\mathbf{s}}_k^{a-}, \hat{\mathbf{f}}_{jk}^{bT}]^T = \left[(\hat{\mathbf{p}}_k^{a-})^T \;,\; \hat{\mathbf{m}}_k^{aT} \;,\; \hat{\mathbf{f}}_{jk}^{bT} \right]^T \tag{27a}$$

$$\mathbf{C}_k^{a+} = \begin{bmatrix} \mathbf{C}_{ppk}^{a-} & \mathbf{C}_{pMk}^{a-} & \mathbf{0} \\ \mathbf{C}_{pMk}^{a-\,T} & \mathbf{C}_{MMk}^{a-} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_{jjk}^{b+} \end{bmatrix} \tag{27b}$$

where no observation process is done ($\mathbf{J}_p^a = \mathbf{0}$); and now, if a new estimate $\hat{\mathbf{f}}_{ik}^a$ is generated by an observation process, the result of applying (23 a, b) again on (27 a, b), will be

$$\hat{\mathbf{s}}_k^{a+} = \left[(\hat{\mathbf{p}}_k^{a-})^T \;,\; \hat{\mathbf{m}}_k^{aT} \;,\; \hat{\mathbf{f}}_{jk}^{bT} \;,\; \hat{\mathbf{f}}_{ik}^{aT} \right]^T \tag{28a}$$

$$\mathbf{C}_k^{a+} = \begin{bmatrix} \mathbf{C}_{ppk}^{a-} & \mathbf{C}_{pMk}^{a-} & \mathbf{0} & (\mathbf{J}_p^a \mathbf{C}_{ppk}^{a-})^T \\ \mathbf{C}_{pMk}^{a-\,T} & \mathbf{C}_{MMk}^{a-} & \mathbf{0} & (\mathbf{J}_p^a \mathbf{C}_{pMk}^{a-})^T \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_{jjk}^{b+} & \mathbf{0} \\ \mathbf{J}_p^a \mathbf{C}_{ppk}^{a-} & \mathbf{J}_p^a \mathbf{C}_{pMk}^{a-} & \mathbf{0} & \mathbf{C}_{iik}^{a+} \end{bmatrix} \tag{28b}$$

It is clear that the estimate $\hat{\mathbf{f}}_{ik}^a$ (augmented from an observed feature) is correlated with the vehicle and previous map, while the estimate $\hat{\mathbf{f}}_{jk}^b$ (generated by another vehicle) remains uncorrelated. Furthermore, considering the structure of Jacobian $\mathbf{H}_{ik}^a$ of the observation model h in (18), it can be noted that all its entries are zeros except the columns corresponding to the vehicle ($\partial h/\partial \mathbf{p}$) and the observed feature ($\partial h/\partial \mathbf{f}$). So, as a result of using $\mathbf{H}_{ik}^a$ for computing $\mathbf{E}_{ik}$ and $\mathbf{K}_{ik}^a$, this structure makes the correlated estimate $\hat{\mathbf{f}}_{ik}^a$ (of the observed feature) the only estimate that affects the state $\hat{\mathbf{s}}_k^{a-}$ (20) and covariance $\mathbf{C}_k^{a-}$ (21).

The convergence of the ordinary EKF-SLAM has been theoretically proved and practically validated by the radar-based demonstration in the fundamental EKF-SLAM solution in [3]. Considering the notation differences, it has been shown that this convergence stems from the positive semidefinite (*psd*) nature of the matrices ($\mathbf{C}_0^{v+}$, $\mathbf{Q}^v$ and $\mathbf{R}$).

The main property of *psd* matrices states: If $A \in R^{n\times n}$ is *psd*, then, for any matrix $B \in R^{m\times n}$, BAB^T is *psd*. Consequently, from (14), since $\mathbf{R}$ is *psd* and $\sigma_{\phi,k}^{2,v} \geq 0$, the covariance matrix $\mathbf{R}_j^e$ of the extended observation is also *psd*, and so will be all the matrices $\mathbf{C}_k^{a-}$ in (8), $\mathbf{E}_{ik}$ in (17 a, b), and $\mathbf{K}_{ik}^a \mathbf{E}_{ik} \mathbf{K}_{ik}^{vT}$ in (21) are all *psd*, so from (21)

$$\det[\mathbf{C}_k^{a+} = \mathbf{C}_k^{a-} - \mathbf{K}_{ik}^a \mathbf{E}_{ik} \mathbf{K}_{ik}^{vT}] \leq \det \mathbf{C}_k^{a-} \tag{29}$$

This ensures that the total uncertainty of the state estimate does not increase during an update, where the determinant of $\mathbf{C}_k^{a+}$, the state covariance matrix, is a measure of total uncertainty in the state estimate $\hat{\mathbf{s}}_k^{a+}$. Since the covariance matrix $\mathbf{R}_{ik}^e$ of the extended observation is psd as $\mathbf{R}$ of the local observation, the EO-CSLAM approach keeps this convergence.

This result agrees with convergence properties of both Kalman filter and SLAM algorithm, i.e., as the vehicle moves through the environment taking observations, the estimation error reduces monotonically [3]. This reduction, as explained above, requires that the estimates to be correlated. Thus, since the EO-CSLAM approach provides collaborating vehicles with additional correlated estimates, more decrease is gained in the estimation errors.

In addition to this mathematical proof, section 3 provides simulations conducted to verify the validity, utility, and performance gain of the EO-CSLAM.

3 Simulation Results

This section verifies the validity of EO-CSLAM algorithm using simulation. At first, ordinary single-USV SLAM (Mono-SLAM) is performed for obtaining independent localization results for every USV. Secondly, the EO-CSLAM algorithm is performed and its results are compared with Mono-SLAM. And then, the CLSF-based CSLAM algorithm, from [7], is performed in order to compare the performance of the two approaches, EO-CSLAM and CLSF, in two cases: when there are common features in the overlapped area between the FOVs of USVs, and the case of absence of such common features.

In order to perform SLAM algorithm simulation, we have designed a simple virtual marine environment so that allows inserting point features and pre-planned trajectories to be followed by the USVs. Fig. 5 shows an example of a $10\text{km} \times 10\text{km}$ area with an arbitrary coast line and ten inserted features. The point features are assumed to be outputs of signal processing algorithms [3] and extraction routines that detect the point targets while suppress land reflections [6]. Two USVs (a and b) were inserted and driven along two closed trajectories of about 1.7 km-long. Dashed circles show the range/field-of-view FOV of each vehicle at the start time. The 5km-range of radar sensor was chosen agreeing with actual detection capability on the sea surface such as in [6].

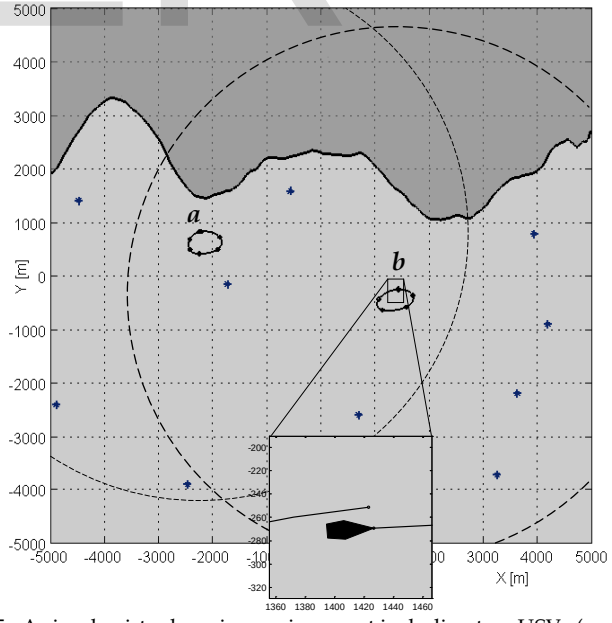

Fig. 5: A simple virtual marine environment including two USVs (*a* and *b*) are taken on two closed trajectories (zooming in on *b* at the start point).

First, a Mono-SLAM is performed for the case of Fig. 5 by each vehicle independently. It can be noted, from Fig. 5, that the vehicle a, through its tour, can observe six features from ten, while b can observe eight features (the eighth feature become

inside FOV during the vehicle tour). Fig. 6 illustrates the positional estimation errors in X and Y of both vehicles, a and b, together with 95% (2σ) confidence limits in the estimation error, in addition to norms of position covariances. The errors in the vehicle estimates are within the 2σ covariance bounds showing that the SLAM filter is well tuned. Once a vehicle moves through the unknown environment, the motion disturbances, $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$, adds uncertainty to its position estimate, and the vehicle covariance grows, but as it continues taking observations, the error in the feature estimates reduces monotonically, what restricts the vehicle positional errors.

Next, the EO-CSLAM is performed for the same case of Fig.5. Considering the norm of covariance as a total criterion for SLAM performance, Fig. 7 illustrates a comparison of norms of position covariance in both cases, Mono-SLAM and EO-CSLAM. It is obvious that the uncertainty in the vehicle position estimate via EO-CSLAM decreases noticeably versus Mono-SLAM. This improvement is due to the additional correlated feature estimates generated by each vehicle through the extended observation way. Fig. 7 shows also the improvement ratio (IR) for each USV, which represents the reduction in the uncertainty as a percentage of the Mono case, defined as

$$IR = \frac{\| \mathbf{C}_{qq} \|_{Mono} - \| \mathbf{C}_{qq} \|_{EO}}{\| \mathbf{C}_{qq} \|_{Mono}} \times 100 \ \ \% \qquad (30)$$

where $\| \mathbf{C}_{qq} \|$ is the norm of vehicle position covariance. In other words, the EO-CSLAM approach has removed about 40% of the localization errors for the vehicle a, and more than 30% of that for b.

In order to analyse and compare the mapping performance of EO-CSLAM versus Mono-SLAM, the global map estimated by the vehicle b is illustrated in Fig. 8. Due to the large-scale map, and in order to clarify the difference between the two cases, close-up views of three features are also shown in the figure. Each numbered point is, actually, two ellipses, one of them represents the 95% confidence area of the feature estimate via EO-CSLAM (at the final instant), while the second is for the same feature, but via Mono-SLAM. However, the close-up view of the 10th feature shows only an ellipse of the EO case; where the features 9 and 10 were always out of FOV of b, see Fig. 5. So, while the vehicle b has mapped eight features of the ten via Mono-SLAM, it was able to map all the ten features by EO-CSLAM approach. Furthermore, from the close-up ellipses of the 1st and 5th features, it can be observed that the location covariances of EO-CSLAM are smaller (better) than those for Mono-SLAM. This improvement can be seen more obviously in Fig. 9; instantaneous comparison of two features' norm of covariance is shown for the two methods. The shown numbers are the lower bounds to which the feature estimates have converged. It is obvious that the estimation errors in the EO case are smaller than those of Mono.

Moreover, while the vehicle b in Mono-SLAM case does not see the 8th feature until $t_k = 815.17$ sec (Fig 9), it observes this feature indirectly using EO-CSLAM from the beginning with helping of a; this can be seen also in Fig.5 where the 8th feature falls out of FOV at the start time, but it is near enough to become inside during the vehicle tour.

Consequently, in addition to improving the localization accuracy, EO-CSLAM approach allows the collaborating USVs to map more features with more accuracy than the Mono-SLAM case.

After verifying the validity of the EO-CSLAM algorithm and its superiority on the Mono-SLAM, the CLSF-based CSLAM algorithm is performed for the same case of Fig. 5, which contains common features in the overlapped area between the two FOVs. Fig. 10 compares the norms of position covariances for the three methods. It can be noted that the EO-CSLAM algorithm, in the first and last parts of the tour duration, produces lower (better) uncertainty than CLSF, while CLSF approach produces uncertainty lower than EO-CSLAM in most of the middle periods. At the end of each period, a sharp reduction occurs in the errors of CLSF; this reduction results from the periodic minimization of the constraints (the duplicated estimates of common features), and during each period, the errors grow again where vehicles start new SLAM operation in each period. So, the performance gain of CLSF cooperative approach requires common features existence.

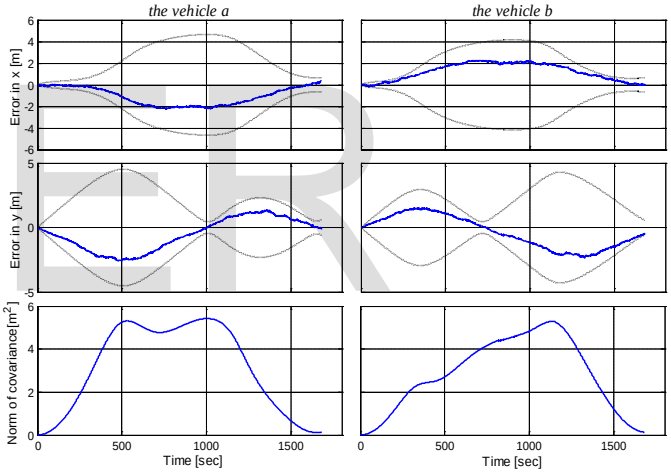

Fig. 6: The two USVs' positional estimation errors in X and Y, together with 95% confidence bounds (dotted lines), for Mono-SLAM, in addition to norms of position estimate covariance.

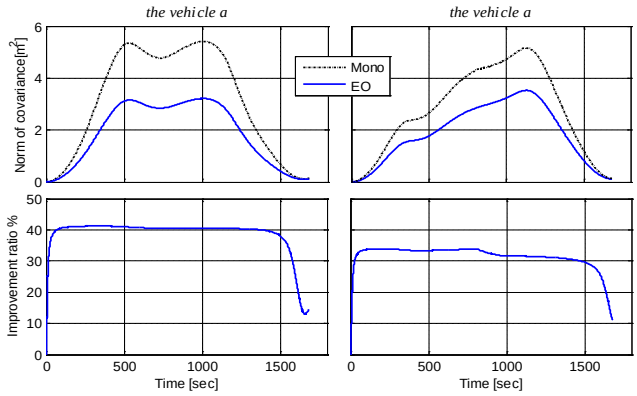

Fig. 7: The comparison of the two vehicles' norm of position covariance in Mono-SLAM case with EO-CALM together with the improvement ratio.

For comparing the performance in the case of absence of any common feature, the three approaches (Mono, EO, and CLSF) are performed for the case of Fig. 11; the case is identical to that of Fig. 5 except that the common features has been moved to be non-common. The resulting norms of position covariances are shown in Fig. 12. As expected previously, while the EO-CSLAM provides noticeable performance improvement versus Mono-SLAM, the CLSF cooperative method yields the same level of accuracy of Mono-SLAM or lower (higher uncertainty). This is because no common features between the USVs to make the required constraints for CLSF. Additionally, as explained in subsection 2.3, the shared, uncorrelated feature estimates cannot improve the estimation of vehicle's location. On the other hand, due to the correlated estimates generated by the extended observation, the EO-CSLAM approach allows vehicles to improve map and vehicle estimates even if features were non-common.

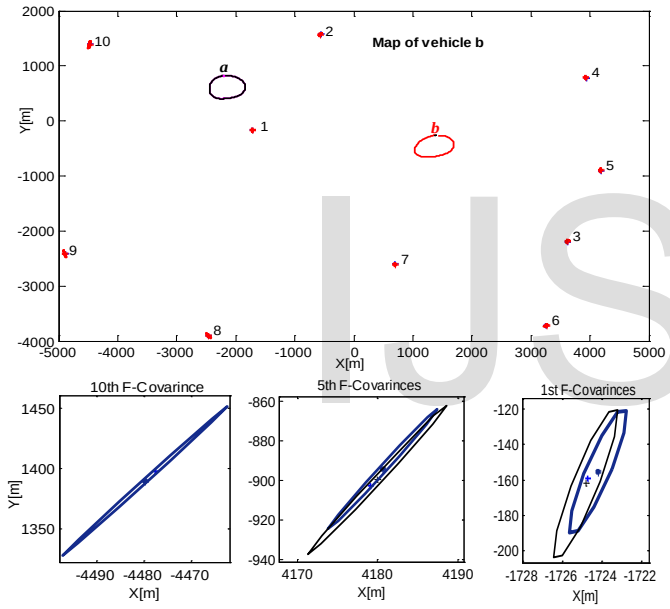

Fig. 8: The map of vehicle *b*, and close-up of three features: first, fifth, and tenth, showing their 95% (2σ) ellipses for Mono (light-line ellipses) and EO-CSLAM (bold ellipses).

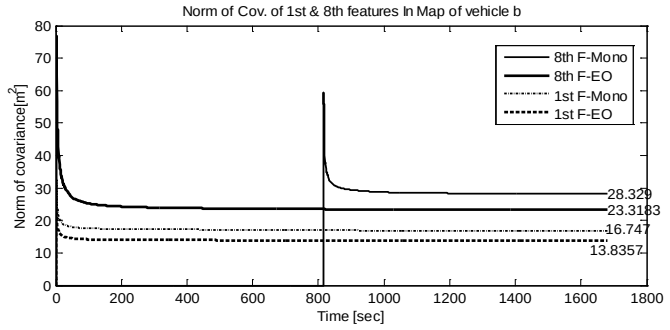

Fig. 9: The comparison of norm of covariance for two features (F) in the Mono-SLAM case with EO-CSLAM.

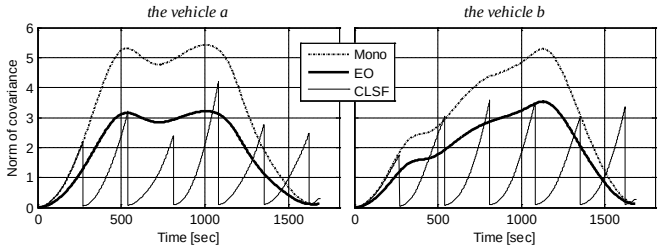

Fig. 10: The comparison of the two vehicles' norm of position covariance in Mono-SLAM, EO-CALM and CLSF-CSLAM with common features.

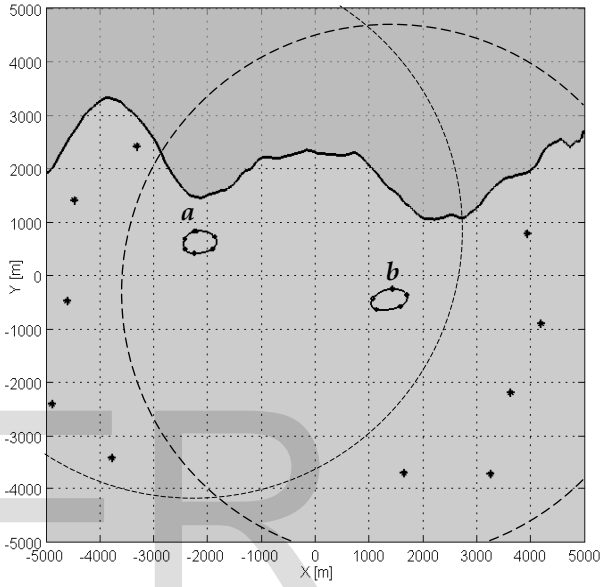

Fig. 11: A testing case with the same previous trajectories, the common features has been moved, so there are no longer common features.

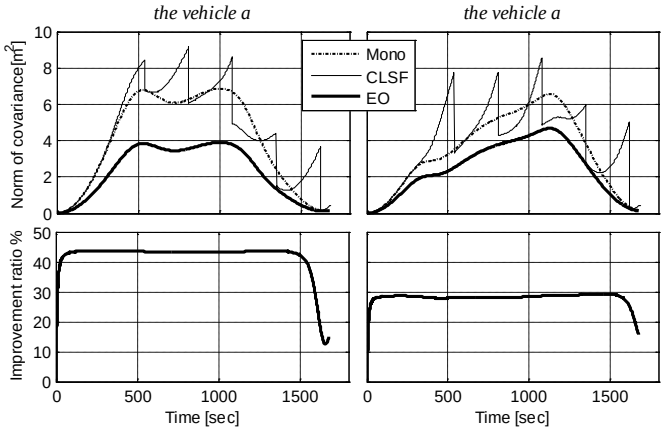

Fig. 12: The comparison of the two vehicles' norm of position covariance in Mono-SLAM, EO-CALM and CLSF-CSLAM without common features, and also the improvement ratio of EO-CALM versus Mono-SLAM.

4 CONCLUSIONS

This paper has proposed an efficient approach to cooperative simultaneous localization and mapping (CSLAM), called extended observation-cooperative SLAM (EO-CSLAM). The proposed algorithm depends on the fact that the necessary cross-correlation between the vehicle and features, required for SLAM convergence, arises from the observation process.

First, the formulation of EO-CSLAM algorithm has been demonstrated and mathematically verified, and then, its validity has been evaluated using simulations for the case of unmanned surface vehicles (USVs) with radar sensors. Simulation results have shown that the EO-CSLAM algorithm improves localization accuracy and mapping performance (compared with Mono-SLAM) and allows the collaborating vehicles to map larger areas with more accurate estimation of features' locations.

It has been shown that the EO-CSLAM algorithm does not required common features to provide performance gain. Therefore, in the case of absence of common features between vehicles, EO-CSLAM acts better than Constrained Local Submap Filter (CLSF) cooperative SLAM, which needs such common features (constraints) to provide accuracy better than Mono-SLAM.

REFERENCES

[1] H. Ferreira, C. Almeida, A. Martins, J. Almeida, A. Dias, G. Silva, E. Silva "Environmental modeling with precision navigation using ROAZ autonomous surface vehicle," in *Proc. IROS – Intelligent Robots and Systems, Workshop on Robotics for Environmental Monitoring*, Porto, Portugal, 2012.

[2] J. C. Leedekerken, M. F. Fallon and J. J. Leonard. "Mapping Complex Marine Environments with Autonomous Surface Craft", Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, 2010.

[3] G. Dissanayake, P. Newman, S. Clark, H.F. Durrant-Whyte, and M. Csorba. "A solution to the simultaneous localisation and map building (SLAM) problem," *IEEE Trans. Robotics and Automation,* 17(3):229-241, 2001H. Poor, "A Hypertext History of Multiuser Dimensions," *MUD History,* http://www.ccs.neu.edu/home/pb/mud-history.html. 1986. (URL link *include year)

[4] B. Kalyan, K. Lee, S. Wijesoma, and N. Patrikalakis. "A clutter rejection filter for Sonar feature based navigation in Marine environments," in Proc IEEE, Oceans'11, 2011. paper 110422-083, pp. 1-6.

[5] E. Rossetti, A. Beck, D. Witt, D. Devecsery, .R. Wolcott, A. Bonkoski, and A. Prog. "UM::Autonomy Wolvemarine, A Highly Capable Autonomous Surface Vehicle", University of Michigan, Ann Arbor, MI 48109, 2011.

[6] J. Mullane, S. Keller, A. Rao, M. Adams, A. Yeo, F. Hover‡ and N. Patrikalakis. "X-band Radar based SLAM in Singapore's Off-shore," in Proc. IEEE, Int. Conf. Control, Automation, Robotics and Vision, 2010, pp. 398-403.

[7] M. Moratuwage, W. S. Wijesoma,B. Kalyan, J. Dong,and P. Namal Senarathne. "Collaborative Multi-Vehicle Localization and Mapping in Marine Environments", in Proc IEEE, Oceans'10, 2010.

[8] S.B. Williams. "Efficient Solutions to Autonomous Mapping and Navigation Problems", PhD thesis, University of Sydney, Australian Centre for Field Robotics, 2001.

[9] S.B. Williams, G Dissanayake, and H Durrant-Whyte. "An Efficient Approach to the Simultaneous Localisation and Mapping Problem", IEEE Proceedings. ICRA'02- Robotics and Automation, 2002.

[10] Dan Simon, Optimal State Estimation, John Wiley & Sons, Inc., Hoboken, New Jersey, 2006. pp. 407-409.