The Effect of Features Number on Extended Observation-Cooperative SLAM

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

Abstract— This paper investigates the effect of the number of environment features on a cooperative approach of simultaneous localization and mapping (SLAM). The tested cooperative SLAM approach is the Extended Observation-Cooperative SLAM (EO-CSLAM) algorithm which depends on additional, indirect correlated observations of the features (landmarks). The performance gain due to additional correlated observations means that additional features will have similar positive effect. However, as EO-CSLAM adopts extended Kalman filter-simultaneous localization and mapping (EKF-SLAM) solution, the number of environment features will have an important role in the computational burden. Simulation results show that the performance gain provided by EO-CSLAM is more obvious in tha less features cases.

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

1 INTRODUCTION

Whatever the task of 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.

For the cooperative SLAM implementation with USVs, a Multi-USV-based CSLAM approach has been proposed in [5] using laser sensors. The research adopted the Constrained Local Submap Filter (CLSF) approach, which had been presented in [6] and [7] 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 theCLSF-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 isa possible situation in large-scale environments, there will not be improvement in localization accuracy and mappingperformance, and CLSF-based CSLAM will act the same level of accuracy of Mono-SLAM or lower.

In our previous work [8], the Extended Observation-Cooperative SLAM (EO-CSLAM) has been presented, this approach 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 this paper more investigation on this algorithm is presented showing the effect of environment features number on the resulting performance gain and computational complexity.

This paper is organized as follows: Section 2 provides the general framework for the EO-CSLAM approach and its formulation, while section 3 evaluates (using simulations) the effect of features number on the performance gain and computational burden. Finally, conclusions are presented in section 4.

2 EO-CSLAM ALGORITHM

In the 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

- *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*

sensors. To demonstrate the algorithm, let's consider the case of two USVs denoted a and b as shown pictorially in Fig.1 involving only three features for simplifying. The features on sea surface can be artificial and/or naturally occurring elements [4], [5].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.1, 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.

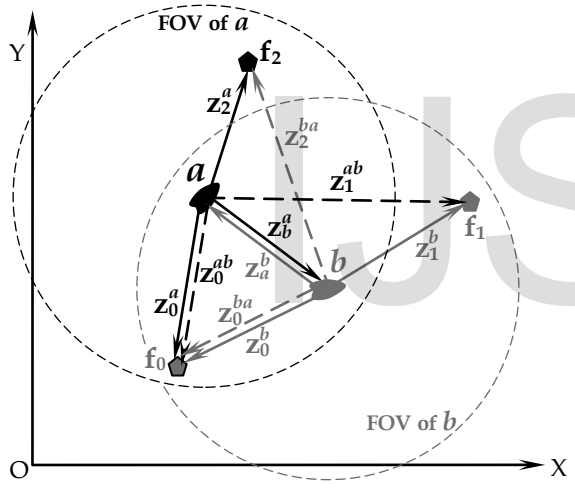

Fig. 1: EO-CSLAM by two USVs (a, b), solid arrows refer to local observations (of features in the FOV) while dashed arrows refer to extended observations, such as $\mathbf{z}_1^{ab}$.

2.1 EO-CSLAM algorithm formulation

The main quantities and their used symbols are defined in [8] together with way of obtaining extened observations. Therefore, it is enough (for this paper's goal) to review the main EKF recursive procedure.

Initialization:

$$\hat{\mathbf{s}}_0^{a+} = \hat{\mathbf{p}}_0^{a+} = [\hat{x}_0^{a+}, \hat{y}_0^{a+}, \hat{\phi}_0^{a+}]^T \tag{1}$$

with covariance $\mathbf{C}_0^{a+} = \mathbf{C}_{pp0}^{a+}$, where no features have been mapped yet ($\mathbf{m}_0^a = \varnothing$). 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:

$$\hat{\mathbf{s}}_k^{a-} = f(\hat{\mathbf{s}}_{k-1}^{a+}, \mathbf{u}_{k-1}^a, \mathbf{0}) \tag{2}$$

$$\mathbf{C}_k^{a-} = \begin{bmatrix} \mathbf{F}_{pk-1}^a \mathbf{C}_{ppk-1}^+ \mathbf{F}_{pk-1}^{aT} + \mathbf{L}_{k-1}^a \mathbf{Q}_{k-1} \mathbf{L}_{k-1}^{aT} & \mathbf{F}_{pk-1}^a \mathbf{C}_{pMk-1}^{a+} \\ \mathbf{C}_{pMk-1}^{a+T} \mathbf{F}_{pk-1}^{aT} & \mathbf{C}_{MMk-1}^{a+} \end{bmatrix} \tag{3}$$

where $\mathbf{F}_{pk-1}^a$ and $\mathbf{L}_{k-1}^a$ are the Jacobeans of motion model [9]:

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 (2) and (3), 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. See [8] for more details.

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 $\mathbf{z}_{ik}^a$:

$$\boldsymbol{\nu}_{ik}^a = \mathbf{z}_{ik}^a - \hat{\mathbf{z}}_{ik}^a = \mathbf{z}_{ik}^a - h(\hat{\mathbf{s}}_k^{a-}, \hat{\mathbf{m}}_k^a) \tag{4a}$$

$$\mathbf{E}_{ik} = \mathbf{H}_{ik}^a \mathbf{C}_k^{a-} \mathbf{H}_{ik}^{aT} + \mathbf{R}_k \tag{5a}$$

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

$$\boldsymbol{\nu}_{ik}^a = \mathbf{z}_{ik}^{ab} - \hat{\mathbf{z}}_{ik}^{ab} = \mathbf{z}_{ik}^{ab} - h(\hat{\mathbf{s}}_k^{a-}, \hat{\mathbf{m}}_k^a) \tag{4b}$$

$$\mathbf{E}_{ik} = \mathbf{H}_{ik}^a \mathbf{C}_k^{a-} \mathbf{H}_{ik}^{aT} + \mathbf{R}_{ik}^e \tag{5b}$$

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

$$\mathbf{H}_{ik}^a = \left[\left. \frac{\partial h}{\partial \mathbf{p}} \right|_{\hat{\mathbf{p}}_k^{a-}} \mathbf{0}_1 \;\; .. \;\; \mathbf{0}_{i-1} \;\; \left. \frac{\partial h}{\partial \mathbf{f}} \right|_{\hat{\mathbf{f}}_{ik}^a} \;\; \mathbf{0}_{i+1} \;\; .. \;\; \mathbf{0}_{Nak} \right] \tag{6}$$

Kalman gain:

$$\mathbf{K}_{ik}^a = \mathbf{C}_k^{a-} \mathbf{H}_{ik}^{aT} \mathbf{E}_{ik}^{-1} \tag{7}$$

Update:

$$\hat{\mathbf{s}}_k^{a+} = \hat{\mathbf{s}}_k^{a-} + \mathbf{K}_{ik}^a \boldsymbol{\nu}_{ik}^a \tag{8}$$

$$\mathbf{C}_k^{a+} = \mathbf{C}_k^{a-} - \mathbf{K}_{ik}^a \mathbf{E}_{ik} \mathbf{K}_{ik}^{vT} = (\mathbf{I} - \mathbf{K}_{ik}^a \mathbf{H}_{ik}^a) \mathbf{C}_k^{a-} \tag{9}$$

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

$$\hat{\mathbf{f}}_{ik}^a = g(\hat{\mathbf{p}}_k^{a-}, \mathbf{z}_{ik}) = \begin{bmatrix} \hat{x}_k^{a-} + r_{ik}.\cos(\hat{\phi}_k^{a-} + \theta_{ik}) \\ \hat{y}_k^{a-} + r_{ik}.\sin(\hat{\phi}_k^{a-} + \theta_{ik}) \end{bmatrix} \tag{10}$$

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

$$\hat{\mathbf{s}}_k^{a+} = \left[\; (\hat{\mathbf{p}}_k^{a-})^T \quad , \hat{\mathbf{m}}_k^{aT} \quad , \quad \hat{\mathbf{f}}_{ik}^{aT} \right]^T \tag{11a}$$

$$\mathbf{C}_k^{a+} = \begin{bmatrix} \mathbf{C}_{ppk}^{a-} & \mathbf{C}_{pMk}^{a-} & (\mathbf{J}_p^a \mathbf{C}_{ppk}^{a-})^T \\ \mathbf{C}_{pMk}^{a-\;T} & \mathbf{C}_{MMk}^{a-} & (\mathbf{J}_p^a \mathbf{C}_{pMk}^{a-})^T \\ \mathbf{J}_p^a \mathbf{C}_{ppk}^{a-} & \mathbf{J}_p^a \mathbf{C}_{pMk}^{a-} & \mathbf{C}_{iik}^{a+} \end{bmatrix} \tag{11b}$$

where

$$\mathbf{J}_p^a = \frac{\partial g}{\partial \mathbf{p}_k^a} = \begin{bmatrix} 1 & 0 & -r_{ik}.\sin(\phi_k^a + \theta_{ik}) \\ 0 & 1 & r_{ik}.\cos(\phi_k^a + \theta_{ik}) \end{bmatrix} \tag{12}$$

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

$$\mathbf{C}_{iik}^{a+} = \mathbf{J}_p^a \mathbf{C}_{ppk}^{a-} \mathbf{J}_p^{aT} + \mathbf{J}_z^a \mathbf{R}_k \mathbf{J}_z^{aT} \tag{13a}$$

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

$$\mathbf{C}_{iik}^{a+} = \mathbf{J}_p^a \mathbf{C}_{ppk}^{a-} \mathbf{J}_p^{aT} + \mathbf{J}_z^a \mathbf{R}_{ik}^e \mathbf{J}_z^{aT} \tag{13b}$$

where

$$\mathbf{J}_z^a = \frac{\partial g}{\partial \mathbf{z}_{ik}} = \begin{bmatrix} \cos(\phi_k^a + \theta_{ik}) & -r_{ik}.\sin(\phi_k^a + \theta_{ik}) \\ \sin(\phi_k^a + \theta_{ik}) & r_{ik}.\cos(\phi_k^a + \theta_{ik}) \end{bmatrix} \tag{14}$$

and so, the next period $(k+1)$ starts with (2) and (3) continuing with the same manner.

2.2 Computational complexity

It has been mentioned that the Constrained Local Submap Filter (CLSF), which is an effective method for reducing the computational burden in small-scale environment, will not be useful for USVs with long-range sensors such as radar where the difference between the local and global map will be small. Therefore, among the methods used for reducing the computational burden of SLAM algorithms, the state augmentation technique has been adopted in the EO-CSLAM algorithm, see (8) and (11 a,b); this technique reduces both the EKF prediction step and the process of adding new feature from calculations with cubic complexity in the number of features to calculations that are linear [10]. However, due the decentralized manner of EO-CSLAM and performing the SLAM procedure in each vehicle for additional observations increases the computational complexity of the algorithm. Section 3 compares the computational burden of EO-CSLAM with that of Mono-SLAM and illustrates the features' number effect on the performance gain.

3 Simulation Results

This section demonstrates (using simulation) the positive role of additional features and observations on the localization accuracy using SLAM, in addition to their negative effect on computational burden, and how will be their role in the improvement of EO-CSLAM approach.

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. 2 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 [4]. 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 [4].

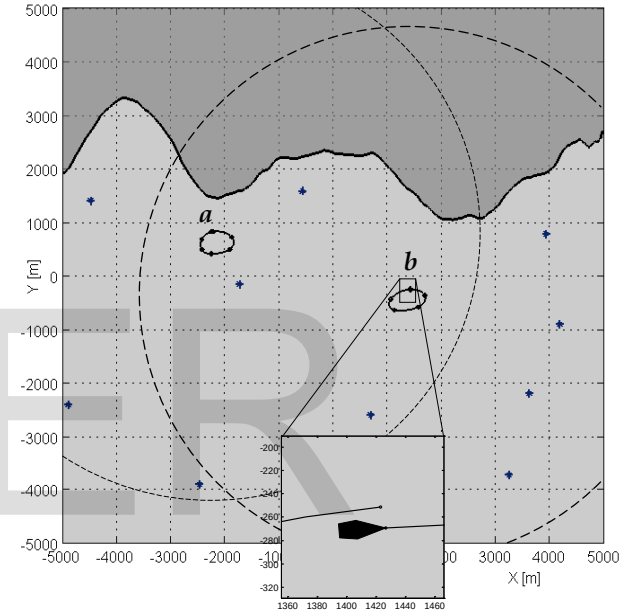

Fig. 2: A simple marine environment including two USVs (*a* and *b*) on two closed trajectories (zooming in on *b* at the start point) with 10 features.

First, both Mono-SLAM and EO-CSLAM have been performed for the case of Fig 2, but with 6 features rather than 10. Considering the norm of positional error covariance as a total criterion for SLAM performance, Fig. 3 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. 3 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

$$IR = \frac{\| \mathbf{C}_{qq} \|_{Mono} - \| \mathbf{C}_{qq} \|_{EO}}{\| \mathbf{C}_{qq} \|_{Mono}} \times 100 \;\; \% \tag{15}$$

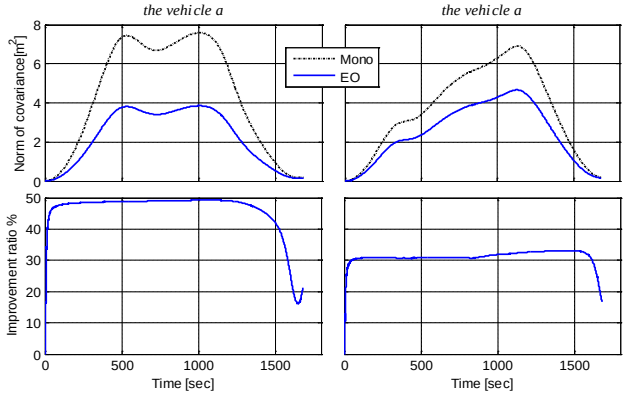

Fig. 3: Comparison of two vehicles' norm of covariance in Mono-SLAM case with EO-CALM, and the improvement ratio for 6-Feature case.

Next, the process has been repeated for 10-feature case and 40-feature case (Fig. 4). Fig. 5 illustrates the comparison for the 10-feature case, and Fig. 6 illustrates the 40-feature case.

Considering tha only the norm of vovariance and caomparing it between the three cases (6, 10, and 40 features), it can be seen that the increment in features number reduces the localization errors for both, Mono-SLAM and EO-CSLAM, and despite that EO-CSLAM is always better than Mono-SLAM, the performance gain will be higher for the less features case; notice that the improvement ratio (*IR*) has reached 50% for the 6-feature case, while for the 10- and 40-feature cases reached about 40% and 30%, respectively. Furthermore, due to the correlation between the vehicle postion eastimate and features locations estimates, the same effect and comparison will be satisfied for the mapping performance.

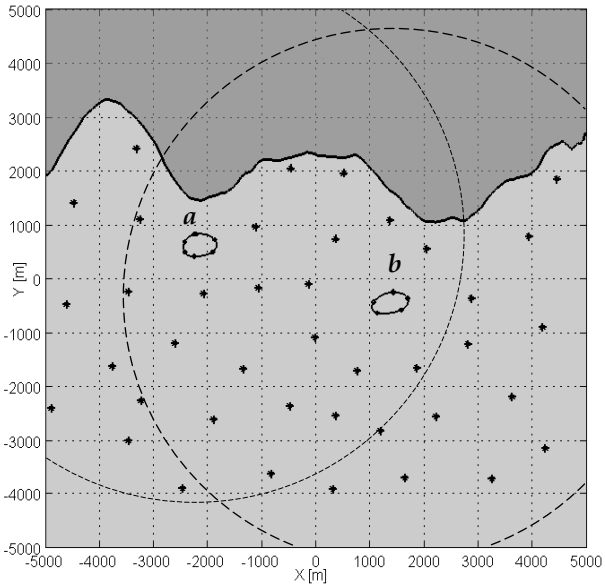

Fig. 4: The same marine environment case with 40 features.

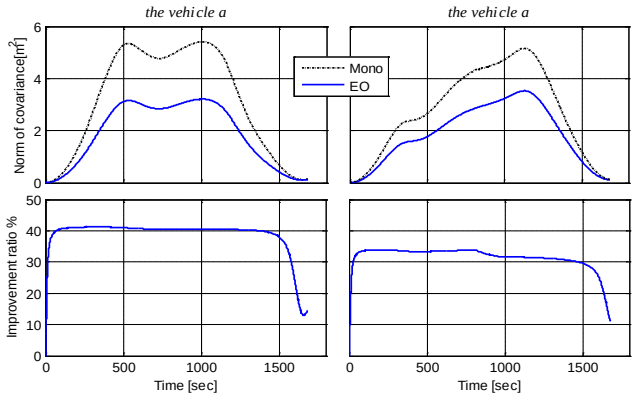

Fig. 5: Comparison of two vehicles' norm of covariance in Mono-SLAM case with EO-CALM, and the improvement ratio for 10-Feature case.

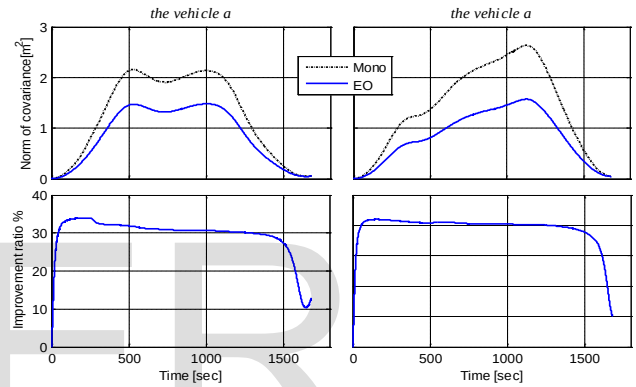

Fig. 6: Comparison of two vehicles' norm of covariance in Mono-SLAM case with EO-CALM, and the improvement ratio for 40-Feature case.

On the other hand, in the purpose of evaluating the computational burden effect of the EO approach and its components, the running time elapsed (in MATLAB) for a single recursion period has been extracted for each method: Mono and EO-CSLAM.

Fig. 7 shows this time for the vehicle b. As expected, the running time increases in the cooperative approaches where a greater amount of information is processed; while Mono-SLAM has the lowermost running time due to the least amount of shared data, EO-CSLAM takes longest time. Thus, since the increment in processed data in SLAM increases the computational burden, more features means longer computating time for both Mono-SLAM and EO-CSLAM, and additionally, the longer computating time is the price of the performance gain obtained using EO-CSLAM.

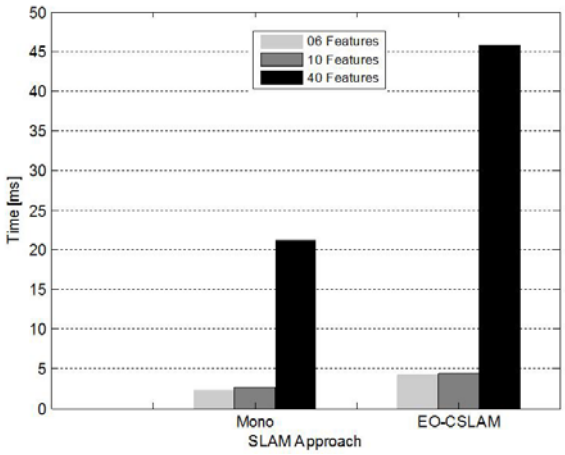

Fig. 7: Running time elapsed (in MATLAB) for a single recursion period for the 10 features' case and vehicle b.

4 Conclusions

This paper has explaind the role of features number in the performance gain of the extended observation-cooperative SLAM (EO-CSLAM). This role appears in two contradictory effects, a positive effect on the localization accuracy and mapping performaence (more features cause more accuracy for both Mono and EO-CSLAM), while the negative effect is the computational burden which incresea with the number of observed feature. While EO-CSLAM is always better than Mono-SLAM, the performance gain will be higher for the cases with less number of environment features.

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] 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.

[5] 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.

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

[7] 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.

[8] G. Asaad, A. Babaei, M. H. Ferdowsi, H. Bolandi, "Extended Observation-Cooperative SLAM for Unmanned Surface Vehicles." International Journal of Scientific & Engineering Research (IJSER), Vol. 4, Issue 8, (pp. 792–800) August 2013.

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

[10] T. Bailey and H. Durrant-Whyte. "Simultaneous localization and mapping (SLAM): Part II, state of the art", IEEE Robotics and Automation Magazine 13(3), 108-117, 2006