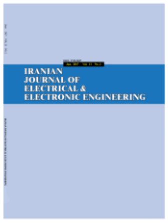
IUST
Iran University of
Science and Technology

Iranian Journal of Electrical and Electronic Engineering

Journal Homepage: ijeee.iust.ac.ir

Research Paper



Effects of Moving Landmark’s Speed on Multi-Robot Simultaneous Localization and Mapping in Dynamic Environments

S. Badalkhani* and R. Havangi*(C.A.)

Abstract: Even when simultaneous localization and mapping (SLAM) solutions have been broadly developed, the vast majority of them relate to a single robot performing measurements in static environments. Researches show that the performance of SLAM algorithms deteriorates under dynamic environments. In this paper, a multi-robot simultaneous localization and mapping (MR-SLAM) system is implemented within a dynamic environment. A probabilistic approach based on extended Kalman filter (EKF) is proposed to detect moving landmarks and consequently improve the performance of SLAM in dynamic environments. The expected landmark area (ELA) is introduced. This concept allows identifying and filtering the moving landmarks. Several experiments are performed varying the speed and number of moving landmarks within the environment to investigate the effect of dynamism level and landmark speed on. The root mean square error (RMSE) is used as a form of measuring the performance of the algorithm. Results show moving landmarks, degrade the performance of classical EKF-SLAM. However, the proposed method is robust to environmental changes and is less affected by the increasing speed of the moving landmarks.

Keywords: Simultaneous Localization and Mapping, Dynamic Environments, Multi-Robot Systems.

1 Introduction

LOCALIZATION or more precisely, self-localization has been one of the fundamental problems in mobile robotics since the beginning. Determine its position within the environment surrounding it from sensor data is a key part of any mobile robot. Even when there is an external positioning system (global positioning system (GPS), local positioning system (LPS), etc.) a self-contained approach is needed to overcome signal faults and improve the precision. On the other hand, to achieve any real movement it is necessary to have a consistent representation of the environment around the robot; a map, which must allow the robot to distinguish between obstacles and freeway to navigate it. This is why localization and mapping are the main blocks on which every mobile robot platform relies.

In the late’90s, many researchers realize that combining the mapping and localization problems into a single one it would result in a convergent estimation problem [1]. Since ever that, SLAM has been the Holy Grail of mobile robotics researchers. Many approaches have been carried out through the years in order to find convergent, fast and reliable solutions to the SLAM problem. However, many of them rely on two well-known and distinctive branches: the EKF based and the particle-filter based. Bailey and Durrant-Whyte established a complete mathematical formulation of the SLAM problem and its classic solutions on their tutorial papers [1, 2]. Even when SLAM solutions have been broadly developed, the vast majority of them relate to a single robot performing measurements in static environments.

In this paper, a solution for the multi-robot SLAM problem in dynamic environments is proposed. The

Iranian Journal of Electrical and Electronic Engineering, 2021.
Paper first received 05 December 2019, revised 25 January 2020, and accepted 31 January 2020.
* The authors are with the Faculty of Electrical and Computer Engineering, University of Birjand, Birjand, Iran.
E-mails: s_badalkhani@birjand.ac.ir and havangi@birjand.ac.ir.
Corresponding Author: R. Havangi.
https://doi.org/10.22068/IJEEE.17.1.1740

proposed method is based on EKF-SLAM and identifies moving landmarks based on probabilistic location constraints and ELA. The ELA is used as a probabilistic index to estimate an approximate location for each landmark at each time frame and decide about inclusion of detected landmarks in EKF correction step. The detected landmarks will have the opportunity to come back to the process, if they stop moving. The communication between individual team members and the leader robot (sink robot), allows the team to share statistics of the moving parts with other team members. An algorithm is responsible for updating the information. Robot and landmark locations at each time frame are used to decide on inclusion of detected landmarks in EKF correction. The main contribution of this research is to introduce a novel probabilistic method which identifies the moving parts of the environment and filters them to reduce the impact of dynamic objects and make SLAM resilient to environmental movements. As the second contribution, the proposed method leverages local maps to detect global changes; hence, it is applicable in multi-robot scenarios. As a result, compared to other methods, the proposed method not only reduces the devastating effects of both dynamism and speed, but also increases the accuracy, robustness, and reliability of the SLAM in harsh environments.

The structure of this paper is as follows. In Section 2, a review of SLAM solutions is presented regarding specific approaches for multi-robots and dynamic environments implementations. Then, in Section 3 the proposed method is introduced. In this section, classic EKF-SLAM is explained in detail. In addition, a way to determine how the speed of dynamic landmarks affects the performance of SLAM is introduced. Then a new method is proposed to reduce the effects of speed of dynamic landmarks in SLAM. In Section 4, the results of experiments are presented. The conclusions reached by the research are presented in Section 5.

2 Related Works

A basic assumption in most current SLAM approaches is that the environment is static. However, some research has been done in the last years regarding dynamic environments. In 2003, the assumption that the errors affecting proprioceptive and exteroceptive sensors are unknown-but-bounded, led to a set-theoretic formulation of the SLAM problem [3]. In 2012, the Bayesian random finite set (RFS) SLAM solution is extended to the collaborative multi-vehicle SLAM (CMSLAM) problem [5]. Then, in 2013, the initial approach is improved by adding moving object tracking (MOT) [6]. It used RFS representation of the feature map and measurements, tracking both static and dynamic features. The corresponding probability density is propagated using Bayes recursion, from which the static feature map and the dynamic feature locations can be estimated. The update phase in the CMSLAM process is carried out using the static feature map only.

In 2015, a robust Graph-based SLAM relying on expectation maximization (EM) algorithms to characterize landmark mobility while establishing the estimations of robot trajectory and the map is proposed [7]. A mobility variable is introduced to scale the effect of every landmark according to how stationary it is. Therefore, moving landmarks are filtered as outliers. In order to evaluate the performance, two datasets of real dynamic environments are used. From then until now, several works keep proposing methods to separate the static part of the environment from the dynamic part, removing the latter from the SLAM algorithm [8, 9].

In 2017, an overview of the latest trends the problem of MR-SLAM is provided [10]. It is focused on a robot operating system (ROS) package designed to solve this problem by enabling the robots to share their maps and merge them over a Wi-Fi network. The approach is tested with G-mapping and hector SLAM well-known packages. In the same year, the adaptive smooth variable structure filter (ASVSF) is introduced as an algorithm that differentiates between the static and dynamic parts of the environment [11]. The ASVSF performance is validated in the real-world and the results confirm the robustness of the approach.

With the birth of convolutional neural networks (CNN) and other deep learning techniques, it is only a matter of time before these tools joined the SLAM field. In 2019, a CNN-based single shot detector (SSD) object detector is proposed to analyze the video stream from the robot camera [12]. Then, it is assigned a characteristic dynamic score based on previous knowledge. Highly dynamic objects score close to 10 and highly static objects score close to 0. Therefore, a moving object such as a person could be scored 9 to 10, while a static table would be scored between 2 to 3 (a table is mainly static but can be moved sometimes). Then, they included a tracking thread that processes only the feature points of dynamic objects. The final implementation of SLAM (based on ORB-SLAM2 [13]) successfully locates and builds an accurate map in real-world tests with a single robot. ORB-SLAM, is a complete SLAM system for monocular, stereo and RGB-D cameras, including map reuse, loop closing, and relocalization which uses Oriented FAST (Features from accelerated segment test) and Rotated BRIEF (Binary Robust Independent Elementary Features) feature detector (ORB) [13]. Very similar approaches using prior knowledge of moving objects, deep learning objects detector, and SLAM-ORB2 as a core can be found in [13-18]. A complete and updated review of visual SLAM in dynamic environment can be read in [19].

3 Multi-Robot SLAM in Dynamic Environments

In this section, the proposed method is presented. For this purpose, a classic EKF-SLAM is explained in detail. Then introduce a way to determine how the speed of dynamic landmarks affects the performance of SLAM algorithms. Finally, a method is introduced to reduce the effects of the speed of dynamic landmarks in SLAM.

3.1 EKF-SLAM

Based on EKF-SLAM formulation, the robot pose and the mapped landmarks form a random state vector called map:

$$X = \begin{bmatrix} R \\ M \end{bmatrix} \tag{1}$$

with

$$R = \begin{bmatrix} x \\ q \end{bmatrix} \text{ and } M = \begin{bmatrix} l_1 \\ \vdots \\ l_n \end{bmatrix} \tag{2}$$

where *R* is the robot state, containing both the robot's position coordinates *x* and its orientation angles *q*. Likewise, *M* is the vector containing the position coordinates of the *n* mapped landmarks ($l_1, \ldots, l_n$). Both *R* and *M* are expressed in the same global reference frame. This formulation assumes point landmarks without orientations. In the EKF framework, the "a posteriori" density is approximated by a Gaussian density with mean $\hat{X} = \begin{bmatrix} \hat{R} & \hat{M} \end{bmatrix}^T$ and covariance matrix *P*. The SLAM keeps this probability density function (PDF) up to date with either the robot movements or landmark observations. The evolution of robot pose during one time step is characterized as:

$$R^+ = f\ (R, u) \tag{3}$$

Here *f*() is the motion function and *u* is the uncertain control signal. This vector of controls permits to modify the robot's trajectory in the state space. A control signal can be a command sent by the computer to the robot or can be determined by proprioceptive sensors (wheel encoders, inertial sensors or even visual odometry). The *u* ~ *N*(*û*; *U*) is assumed to be Gaussian with mean *û* and covariance matrix *U*. Based on EKF-SLAM, the prediction step is described as:

$$\hat{R}^+ = f\ (\hat{R}, \hat{u}) \tag{4}$$

$$\hat{P}^+ = F_R P F_R^T + F_u U F_u^T \tag{5}$$

where *P* and *U* are the covariance matrices, the F_R and F_u are the Jacobian matrices obtained by linearizing function *f* at the current estimated state point *R* and the control *u*:

$$F_R = \left.\frac{\partial f}{\partial R^T}\right|_{\hat{R},\hat{u}}, \qquad F_u = \left.\frac{\partial f}{\partial u^T}\right|_{\hat{R},\hat{u}} \tag{6}$$

Currently, observed landmarks help SLAM make corrections. As the robot moves, it might encounter mapped landmarks, and then the existing map will be corrected based on the new acquired measurements. The measurement from the observed landmark l_i at time *k* is as follows:

$$y_i = h(R, l_i) + \omega \tag{7}$$

where *h*()is the observation matrix and *ω* ~ *N*(0; *W*) is a white Gaussian measurement noise with covariance matrix *W*. In the correction step, the EKF-SLAM takes observations and calculates the difference between real landmark observations (y_i) and its predictions. This allows the algorithm to perform corrections on the prior estimation as follows:

$$z_i = y_i - h\left(\hat{R}, \hat{l}_i\right) \tag{8}$$

$$Z_i = H_i P H_i^T + W \tag{9}$$

$$K_i = P H_i^T . Z_i^{-1} \tag{10}$$

$$\hat{X}^+ = \hat{X} + K_i . z_i \tag{11}$$

$$P^+ = P - K_i Z_i K_i^T \tag{12}$$

where *K* is Kalman gain, *z* and *Z* are the innovation's mean and its covariance matrices, respectively. The innovation *z* is the difference between the actual measurements and the predictions. Equations (10) and (11) constitute the filter update. The $\hat{X}^+$ is the extended state vector after the correction step, with covariance P^+. The Jacobian matrix *H* is defined in a similar way as F_R and F_U by linearizing the function *h* at the current estimated state point $\hat{X}$ as:

$$H_i = \left.\frac{\partial h\left(\hat{R}, \hat{l}_i\right)}{\partial X}\right|_{\hat{X}} \tag{13}$$

As long as the robot moves, the prediction and correction steps are implemented to process the SLAM and build the environment map.

3.2 Landmarks Speed

In order to determine how the speed of dynamic landmarks affects the performance of SLAM, it is necessary to define a relative rate of change that considers both the speed at which the landmarks move and the rate at which they are processed by the sensor. In this case, we refer to as a sensor to the complete perception system that processes the raw data and extracts the references. The sensor data is processed at a rate of *F* frames per second.

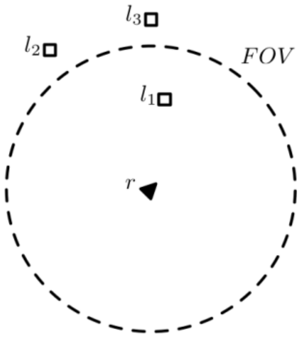


Fig. 1 A single robot surrounded by three landmarks.

Fig.1 shows a robot (r) and three landmarks (l_1, l_2, l_3). The field of the view (FOV) of the robot sensor is defined as a circular region of diameter D surrounding the robot. In this figure, only l_1 is within the FOV of the sensor and the other two landmarks remain unseen to it. If the speed of movement of a landmark is 1D and higher, it will disappear from the current FOV in the next time frame. Therefore, it is not worthwhile to study speeds higher than 1D. Considering the FOV diameter D, and sensor data process rate F, we can define the speed of the dynamic landmarks are as:

$$v^l = \alpha DF \tag{14}$$

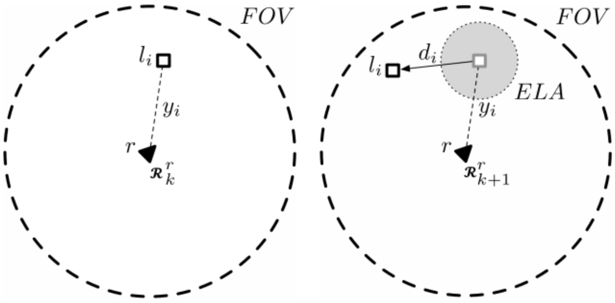
where $0 < \alpha < 1$ is a coefficient that allows studying the whole ranges of speed of interest. $\alpha = 1$ is equivalent to static landmarks and $\alpha = 1$ is equivalent to the maximum speed, where the landmark is detected in a single frame. On the other hand, the movement of the robot must ensure that the overlap between the FOV in one frame and the FOV in the next frame is large enough for the SLAM convergence. This requires the movement speed of the robot to be less than 1D per frame.

3.3 Reducing the negative effects of movement

Traditional EKF-SLAM loses accuracy in dynamic environments. As will be shown in the result section, increasing the speed or the dynamism level quickly deteriorates the performance of EKF-SLAM. In this paper, a new method is proposed to reduce the negative effects of the speed of landmarks and other robots. This will allow the correct implementation of multi-robot SLAM in dynamic environments. In the proposed method, the dynamic landmarks of the environment are identified and eliminated through a probabilistic approach. In order to prevent making corrections based on erroneous measurements of the moving landmarks, a new concept called the expected landmark area (ELA) is introduced. This concept allows identifying and filtering moving landmarks.

3.3.1 Landmarks Identification and Filtering

Following the same notation introduced in the previous sections, the robot r is located in state R^r_k at time instant k. An observation y_i of a single landmark l_i is performed as Fig. 2(a). By applying the inverse observation function g, the position l_i of the observed landmark is as:



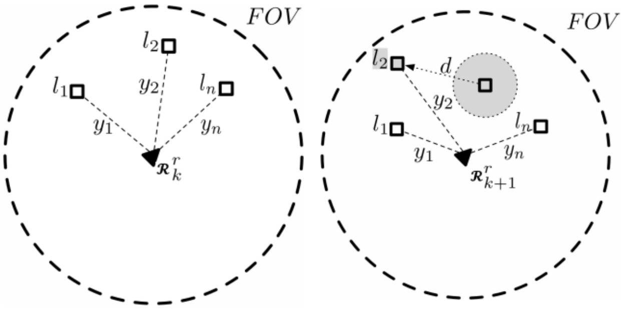
(a) (b)

Fig. 2 a) The robot discovers a single landmark and b) robot moves based on the movement function and the landmark is outside the ELA.

$$l_i = g\left(R^r_k, y_{ik}\right) \tag{15}$$

This location is stored in a map called L-map. The L-map is further discussed in subsection 3.2.2. Then the robot goes on with its exploration of the environment based on motion function f. Next time instant $k + 1$, the robot might face different situations according to its new position R^r_{k+1}. First, the ELA is calculated. The ELA for landmark l_i represents the location where it is expected to be on the global map as Fig. 2(b). By defining a location threshold, it is possible to distinguish whether a landmark is static or dynamic. If the distance d between the actual landmark location l_i and the *ELA* center is greater than threshold d_i, l_i is labeled as a moving landmark. In this case, any previous information about l_i is removed from the L-map and its current location is added. Otherwise, it is a static landmark and its location is updated via EKF-SLAM. Rather than using Euclidean distance, the comparison is made by Mahalanobis distance as will be explained in the next section.

The remaining case happens if both ELA and landmark l_i are not within the FOV. In this case, due to lack of information, the location of the landmark in L-map remains unchanged so it can be used later. The process already described is easily extended to the multiple detected landmarks case adding other considerations. The relative position of a landmark, considering other neighboring landmarks, is used as an index to check the mobility of landmarks. Fig. 3(a) visualizes a robot located in R^r_k at time instant k. The robot makes the measurements y_1, …, y_n, calculates the landmark locations l_1, …, l_n , and stores these locations in L-map. Then, the robot moves and its location changes to R^r_{k+1} at time instant $k + 1$ as Fig. 3(b). When considering multiple landmarks it seems obvious that a constraint network is formed between them. So these constraints must remain almost fixed if the landmarks are static. A moving landmark could be easily detected due to its inconsistent behavior regarding the other



(a) (b)

Fig. 3 a) The robot observes multiple landmarks and b) The robot moves and l_2 is outside its ELA.

surrounding landmarks.

The relative position where landmark l_1 is supposed to be in the next time instant is called $\tilde{l}_i^+$ which $\tilde{l}_i^+$ can be interpreted as the ELA of each landmark observed in time k and expected to be observed in time $k + 1$. The $\tilde{l}_i^+$ is calculated for each landmark in the world reference frame using the other $n - 1$ landmarks positions (mean and covariance). The $\tilde{l}_i^+$ is calculated in a Bayesian way as follows:

$$P\left(\tilde{l}_i^+\right) = \prod_{i=1}^{n-1} P(l_i \mid l_j) = \prod_{i=1}^{n-1} \frac{P\left(l_i\right) P(l_j \mid l_i)}{P\left(l_j\right)} \tag{16}$$

The term $P(l_j \mid l_i)$ is calculated by geometry transformations. The probability $P(\tilde{l}_i^+)$ forms a Gaussian distribution itself so the resulting ELAs will be an ellipsoid from the geometrical point of the view [20]. Every landmark that remains within its ELA is incorporated into the main map. Otherwise is considered a dynamic landmark. Since the proposed approach is probabilistic, the criterion for deciding whether a landmark is within or outside the ELA uses the Mahalanobis distance instead of the Euclidean distance. The prior from EKF-SLAM $\hat{X}^+$ is necessary to obtain the approximate locations of fixed landmarks in the next time frame.

3.3.2 L-Map Algorithm

Every robot in this implementation builds its L-map while contributes to the global map of the environment. The global map is built by only one of the robot and it is called the main robot. The fact that each robot manages its local map (L-map) distributes the computational load of the SLAM problem making it more feasible to implement on a large-scale environment.

As described previously, in the proposed method, every landmark must pass the ELA criterion in order to be added to the L-map and also be shared with the main robot to add it to the global map. If a moving landmark is detected not only the L-map is updated but also a message is sent to the main robot. This one deletes the landmark from the global map and notifies the other robots about the deleted landmark. In summary, the pseudo-code of the proposed method is as Algorithm.1.

4 Results

The simulations are made using a 6DOF SLAM toolbox for MATLAB developed by Juan Solá *et al.* [21]. The toolbox is modified to include several robots and moving landmarks. All simulations are performed in a 3D environment, which is traversed by a team of three robots. Each robot has a fixed initial position and follows a pre-defined semicircular trajectory. The environment comprises an area of 400 square meters in which 72 landmarks are distributed with fixed initial positions as can be seen in Fig. 4.

Initially, a number of landmarks are picked randomly according to the dynamism percentage. Every dynamic landmark receives a direction of movement (front, back, right, left) assigned randomly. Each landmark moves in its assigned direction at the speed selected for the experiment. Every twenty frames, the direction of movement of each landmark is switched to the opposite.

Thirty different experiments are carried out in order to determine how the speed of dynamic landmarks affects the performance of EKF-SLAM. Then the experiments are repeated using the proposed method. Every experiment runs for 150 frames and was performed 10 times. Five values of dynamism (5%, 20%, 35%, 50%, 65%, and 80%) and six values of speed (0.1m/s, 0.3m/s, 0.5m/s, 0.7m/s, and 0.9m/s) are tested. A common practice among SLAM literature is the use of RMSE as a form of measure the performance of the algorithm [12]. For every time frame, the average of RMSE of all the landmarks $RMSE_k$ is calculated as:

$$RMSE_k = \frac{1}{n}\sum_{i=1}^{n}\sqrt{\left(\hat{x}^{l_i} - x^{l_i}\right)^T \left(\hat{x}^{l_i} - x^{l_i}\right)} \tag{17}$$

where $\hat{x}^{l_i}$ is the estimated landmark position and $\hat{x}^{l_i}$ is the real landmark position. In order to obtain a single measure to evaluate the performance of every experiment $RMSE_k$ is averaged over the entire simulation time (150 frames) as made in [12]:

$$RMSE = \frac{1}{150}\sum_{k=1}^{150} RMSE_k \tag{18}$$

Table 1 shows the RMSE total values for the landmarks location. It can be seen that for every experiments the proposed method exhibit less error than the original EKF-SLAM. Fig. 5 shows the same data as Table 1 in a 3D bar graph. As can be seen, the EKF-SLAM deteriorates as dynamism increases. In the same way, for high levels of dynamism, the EKF-SLAM slowly deteriorates as speed increases. On the contrary, for dynamism under 50%, there is no conclusive evidence about speed effect over the performance. The proposed method proved to be almost insensitive to both

Algorithm 1 Pseudo-code of the proposed method.

```
for each robot in the team
build L-map
update state vector
  for each time frame
  determine FOV
  calculate ELA and $\tilde{l}_i^+$ for the observed landmarks in the L-map
  move and sense the environment
    if no landmarks are detected
    delete from the L-map those landmarks who were expected to be sensed in ELA
    else
      for each observed landmark
        if the landmark previously L-mapped
          if a single landmark is detected
            if in ELA → update L-map and pass the landmark to back-end
            if out ELA → delete the landmark info from L-map and inform main robot
          else if multiple landmarks are detected
            calculate $\tilde{l}_i^+$ for each observed landmark
            if in ELA → update L-map and pass the landmark to back-end
            if out ELA → delete the landmark info from L-map and inform main robot
            end if
           update L-map
         else if the landmark is newly discovered
          enter newly discovered landmark info to the L-map
          update L-map
         end if
      end for
    end if
  end for
end for
```

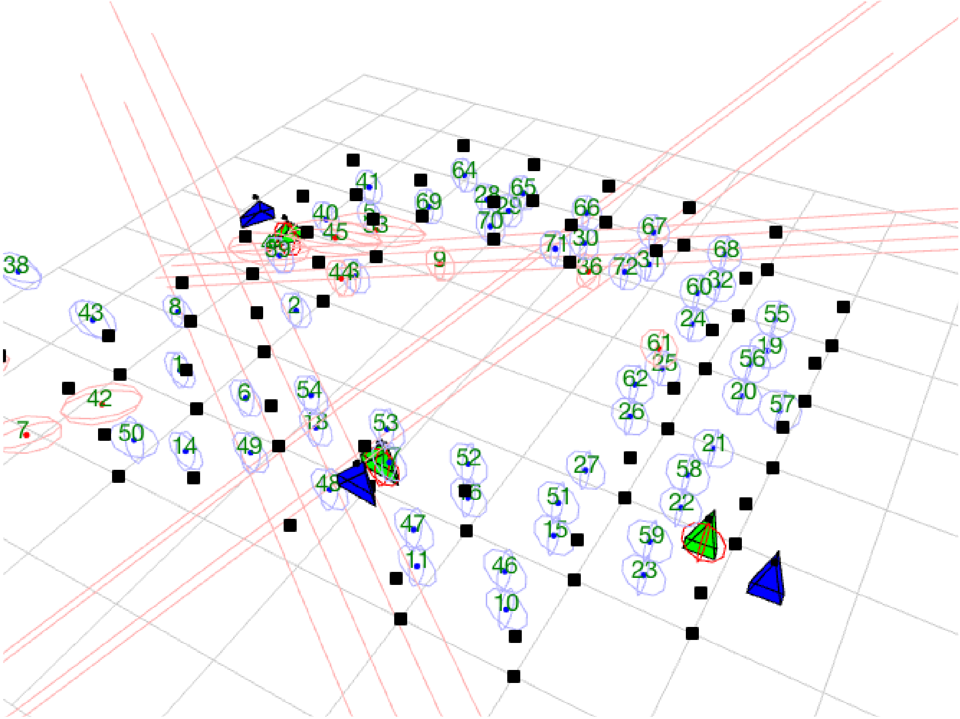
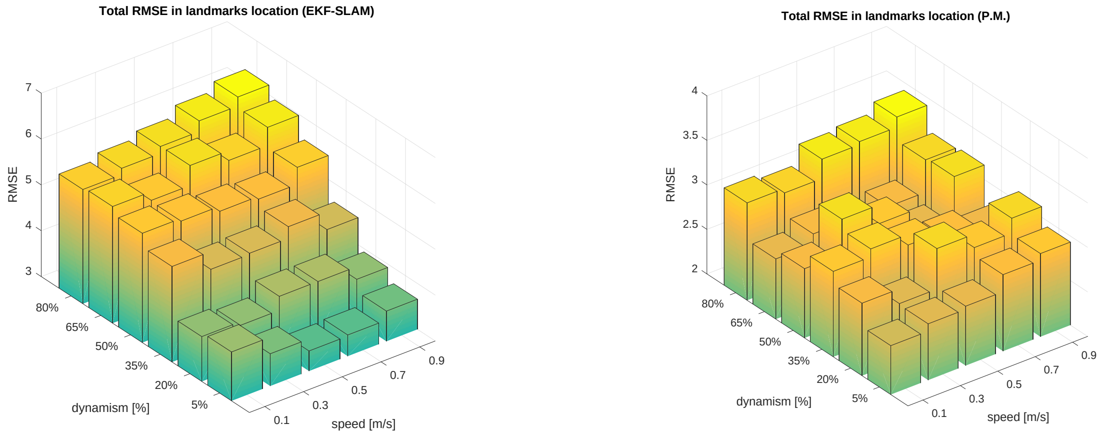


Fig. 4 Simulated environment: three simulated robots (blue), current estimated positions of the robots (green) and uncertainty ellipses (red). Landmarks (black dots) and its uncertainty ellipses (blue and red). The grid is two meters wide.

dynamism and speed except for the most extreme cases. With such a high percentage of dynamism, 80%, even when the method deletes the dynamic landmarks, there are not enough static landmarks to accomplish a better estimation.

Table 1 Total RMSE in Landmarks location for every experiment.

Level of dynamism [%]	Level of speed									
	Very slow [0.1m/s]		Slow [0.3 m/s]		Medium [0.5 m/s]		Fast [0.7 m/s]		Very fast [0.9 m/s]	
	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.
5	4.07	2.55	3.70	2.63	3.44	2.67	3.46	2.86	3.66	2.93
20	3.98	2.81	3.90	2.65	4.21	3.10	4.20	2.95	3.93	3.12
35	5.08	2.96	4.70	3.05	4.72	2.93	4.98	2.86	4.59	2.76
50	5.38	2.78	5.33	3.17	5.21	2.96	5.15	2.71	5.52	3.16
65	5.54	2.68	5.35	2.79	5.79	2.61	5.57	2.78	5.97	3.14
80	5.49	3.09	5.63	3.04	5.78	3.26	6.01	3.29	6.21	3.41



(a) (b)

Fig. 5 Average RMSE for the 30 experiments: a) EKF-SLAM and b) Proposed method.

Table 2 Total RMSE in location of robot 1 for every experiment.

Level of dynamism [%]	Level of speed									
	Very slow [0.1m/s]		Slow [0.3 m/s]		Medium [0.5 m/s]		Fast [0.7 m/s]		Very fast [0.9 m/s]	
	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.
5	0.490	0.225	0.592	0.240	0.407	0.232	0.426	0.278	0.516	0.314
20	0.456	0.250	0.562	0.245	0.503	0.266	0.498	0.391	0.485	0.287
35	0.764	0.324	0.838	0.369	0.648	0.283	0.651	0.294	0.508	0.253
50	0.873	0.265	0.894	0.359	0.699	0.286	0.664	0.248	0.899	0.318
65	1.467	0.212	0.791	0.257	0.910	0.218	0.778	0.269	0.899	0.353
80	1.707	0.217	1.288	0.217	0.987	0.257	1.116	0.230	0.873	0.272

Table 3 Total RMSE in location of robot 2 for every experiment.

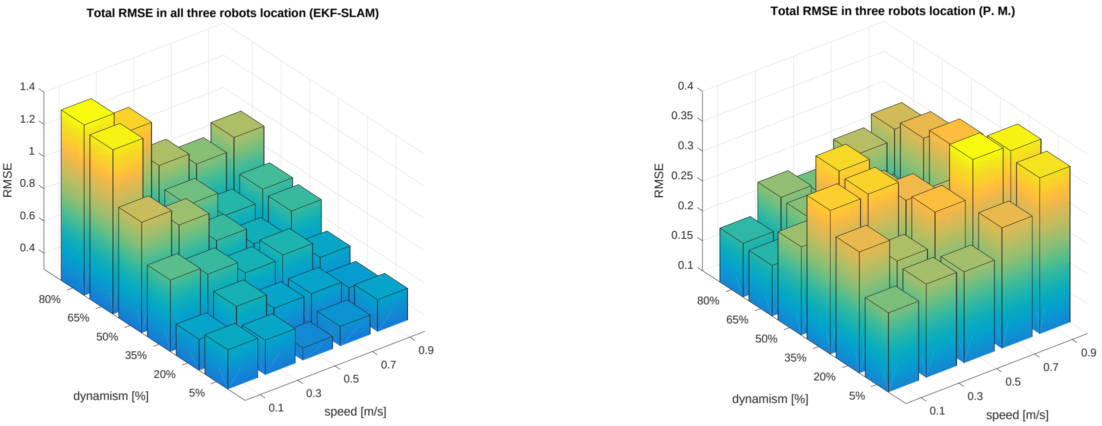
Level of dynamism [%]	Level of speed									
	Very slow [0.1m/s]		Slow [0.3 m/s]		Medium [0.5 m/s]		Fast [0.7 m/s]		Very fast [0.9 m/s]	
	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.
5	0.583	0.252	0.467	0.305	0.360	0.338	0.379	0.339	0.494	0.493
20	0.492	0.363	0.601	0.302	0.475	0.434	0.521	0.436	0.393	0.496
35	0.761	0.399	0.605	0.371	0.644	0.374	0.712	0.287	0.525	0.229
50	0.861	0.277	0.824	0.439	0.792	0.322	0.598	0.189	0.621	0.393
65	1.252	0.138	0.789	0.239	0.865	0.237	0.605	0.191	0.672	0.296
80	1.108	0.157	1.081	0.287	1.053	0.219	0.696	0.325	1.093	0.307

Tables 2-4 show similar results for the three robots positions. As before, the proposed method exhibits better performance in all cases but three that can be treated as outliers. Fig. 6 shows a 3D representation of the averaged RMSE of the three robots. In this case, as with landmarks, the effect of dynamism to degrade the EKF-SLAM is more pronounced than the effect of speed. Using the proposed method the differences are so subtle (less than 0.2m) that it can be said that variation on both, speed and dynamism, has almost no effect on performance.

Fig. 7(a) shows the evolution of RMSE in landmarks over time for several simulations with very slow dynamism (5%). The proposed method (continuous lines) exhibits better performance than the EKF-SLAM (dashed lines) for all the speed cases. In the same way, Fig. 7(b) shows the evolution of RMSE in landmarks over time with very high dynamism (80%). As expected, the performance of the proposed method is better than EKF-SLAM.

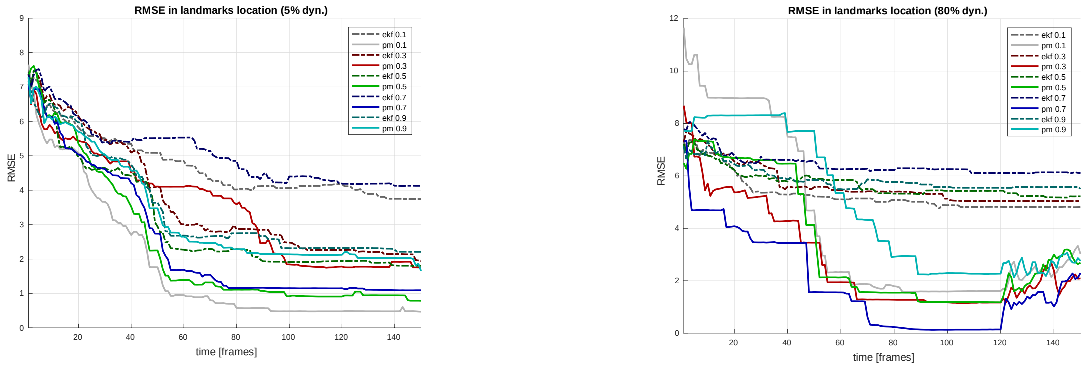
Table 4 Total RMSE in location of robot 3 for every experiment.

Level of dynamism [%]	Level of speed									
	Very slow [0.1m/s]		Slow [0.3 m/s]		Medium [0.5 m/s]		Fast [0.7 m/s]		Very fast [0.9 m/s]	
	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.	EKF.	Prop.
5	0.560	0.252	0.486	0.305	0.366	0.338	0.460	0.339	0.483	0.493
20	0.532	0.363	0.662	0.302	0.546	0.434	0.505	0.436	0.490	0.496
35	0.710	0.399	0.633	0.371	0.539	0.374	0.535	0.287	0.551	0.229
50	1.228	0.277	0.925	0.439	0.594	0.322	0.493	0.189	0.578	0.393
65	1.243	0.138	0.659	0.239	0.627	0.237	0.552	0.191	0.585	0.296
80	1.261	0.157	1.276	0.287	0.743	0.219	0.732	0.325	0.804	0.307



(a) (b)

Fig. 6 Average RMSE for all three robots during the 30 experiments: a) EKF-SLAM and b) Proposed method.



(a) (b)

Fig. 7 RMSE over time: a) 5% of dynamism and b) 80% of dynamism.
*Numbers in the legends represent landmarks speed.

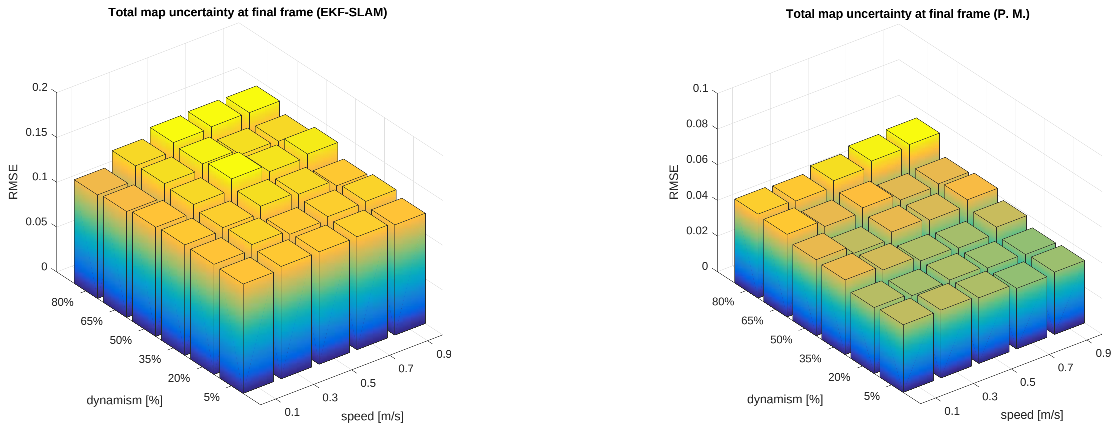
Lastly, for every experiment, uncertainty over the generated map is analyzed using the mean of the diagonal of the covariance matrix diagonal as follows:

$$\text{Ave} = \text{mean}\left(\text{diag}\left(\hat{P}_{k=150}\right)\right) \tag{19}$$

This parameter allows comparing the uncertainty across all the experiments as can be seen in Fig. 8. Once again, original EKF-SLAM exhibits higher uncertainty. It is worth noting that using the original EKF-SLAM, the final uncertainty increases with both increased dynamism and increased speed. However, the proposed method reduces the effect of the former and almost eliminates the effect of the latter.

5 Conclusion

Researches show that the performance of SLAM algorithms deteriorates under dynamic environments. To limit the effect moving landmarks on performance SLAM algorithm, in this paper, a multi-robot simultaneous localization and mapping is implemented within an environment with moving landmarks. The proposed method uses the limited maps around each robot and detects the moving landmarks based on probabilistic constraints. Landmarks positions and the geometric constraints between those are the key aspects to identify and reject erroneous measurements obtained from moving parts of the environment. Thirty different tests are performed for different levels of dynamism (5%, 20%, 35%, 50%, 65%, and 80%) and speed



(a) (b)

Fig. 8 Total map uncertainty at the final frame: a) EKF-SLAM and b) Proposed method.

(0.1m/s, 0.3m/s, 0.5m/s, 0.7m/s, and 0.9m/s) in order to measure the effect of moving landmarks and their speed on SLAM performance. Every experiment runs for 150 frames and is performed 10 times. The RMSE of the landmarks estimated positions, the RMSE of the robot estimated positions, and the final uncertainty of the final map are studied to compare the performance of both algorithms while varying the values of dynamism and speed. In general, the results agree that the effect of increasing dynamism in classic EKF-SLAM is devastating. However, the increase in landmarks speed worsens SLAM performance in some cases but never as devastatingly as the increase on dynamism. However, using the proposed method, the reduction of uncertainty in the final estimated map is above 50% with respect to the uncertainty in the map estimated by the traditional EKF-SLAM. The proposed method not only reduces the devastating effects of both dynamism and speed, but also increases the accuracy, robustness, and reliability of the SLAM in harsh environments. There are still several possibilities for future extensions. Since the proposed method is a probabilistic method that acts on the front-end of the SLAM, the authors suggest implementing the method with other SLAM approaches such as Graph-based SLAM methods to further evaluate effectiveness. In addition, the noise and perturbations at the present study are assumed as uni-modal Gaussian variables. In the future, the authors implement non-Gaussian error models which are more realistic. Data association is one of the main sources of the failure in SLAM algorithms mainstream SLAM methods have been developed with the rigid and static world assumption. The proposed method addresses the robustness against the dynamics of the environments. However, the real world is non-rigid also due to the inherent deformability of objects. Hence, making the SLAM back-end resilient against spurious measurements such as perceptual aliasing and wrong loop-closures is a compelling area for future works. Finally, although computer simulations are useful for evaluating the effectiveness of methods, they might neglect important practical issues that appear in the real world. In this sense, future works can be focused on developing real experiments to validate the applicability of the proposed approach completely.

References

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: Part I," *IEEE Robotics & Automation Magazine*, Vol. 13, No. 2, pp. 99–110, 2006.

[2] T. Bailey and H. Durrant-Whyte, "Simultaneous localisation and mapping (SLAM) Part 2: State of the art," *Robotics and Automation Magazine*, 2006.

[3] M. Di Marco, A. Garulli, A. Giannitrapani, and A. Vicino, "Simultaneous localization and map building for a team of cooperating robots: a set membership approach," *IEEE Transactions on Robotics and Automation*, Vol. 19, no. 2, pp. 238–249, 2003.

[4] Z. Wei, G. Huang, and P. Wang, "The research on multi-robot simultaneous localization mapping algorithm," in *IEEE International Conference on Automation and Logistics*, pp. 1241–1246, 2007.

[5] D. Moratuwage, B. N. Vo, D. Wang, and H. Wang, "Extending bayesian RFS SLAM to multi-vehicle SLAM," in *12th International Conference on Control Automation Robotics & Vision (ICARCV)*, pp. 638–643, 2012.

[6] D. Moratuwage, B. N. Vo, and D. Wang, "Collaborative multi-vehicle SLAM with moving object tracking," in *IEEE International Conference on Robotics and Automation*, pp. 5702–5708, 2013.

[7] L. Xiang, Z. Ren, M. Ni, and O. C. Jenkins, "Robust graph SLAM in dynamic environments with moving landmarks," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2543–2549, 2015.

[8] Y. Sun, M. Liu, and M. Q. H. Meng, "Motion removal for reliable RGB-D SLAM in dynamic environments," *Robotics and Autonomous Systems*, Vol. 108, pp. 115–128, 2018.

[9] H. Wang, C. Zhang, Y. Song, and B. Pang, "Master-followed multiple robots cooperation SLAM adapted to search and rescue environment," *International Journal of Control, Automation and Systems*, Vol. 16, No. 6, pp. 2593–2608, 2018.

[10] M. A. Abdulgalil, M. M. Nasr, M. H. Elalfy, A. Khamis, and F. Karray, "Multi-robot SLAM: An overview and quantitative evaluation of MRGS ROS framework for MR-SLAM," in *International Conference on Robot Intelligence Technology and Applications*, pp. 165–183, 2017.

[11] F. Demim, A. Nemra, A. Boucheloukh, E. Kobzili, M. Hamerlain, and A. Bazoula, "SLAM based on adaptive SVSF for cooperative unmanned vehicles in dynamic environment," *IFAC-PapersOnLine*, Vol. 52, No. 8, pp. 73–80, 2019.

[12] L. Xiao, J. Wang, X. Qiu, Z. Rong, and X. Zou, "Dynamic-SLAM: Semantic monocular visual localization and mapping based on deep learning in dynamic environment," *Robotics and Autonomous Systems*, Vol. 117, pp. 1–16, 2019.

[13] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and RGB-D cameras," *IEEE Transactions on Robotics*, Vol. 33, No. 5, pp. 1255–1262, 2017.

[14] C. Yu, Z. Liu, X. J. Liu, F. Xie, Y. Yang, Q. Wei, and Q. Fei, "DS-SLAM: A semantic visual slam towards dynamic environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1168–1174, 2018.

[15] F. Zhong, S. Wang, Z. Zhang, and Y. Wang, "Detect-SLAM: Making object detection and slam mutually beneficial," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1001–1010, 2018.

[16] B. Bescos, J. M. Fácil, J. Civera, and J. Neira, "DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes," I*EEE Robotics and Automation Letters*, Vol. 3, No. 4, pp. 4076–4083, 2018.

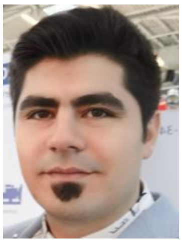
[17] J. Yang and Z. Kang, "A scene-assisted point-line feature based visual slam method for autonomous flight in unknown indoor environments," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 42, No. 2/W13, 2019.

[18] W. Zhang, Q. Chen, W. Zhang, and X. He, "Long-range terrain perception using convolutional neural networks," *Neurocomputing*, Vol. 275, pp. 781–787, 2018.

[19] M. R. U. Saputra, A. Markham, and N. Trigoni, "Visual SLAM and structure from motion in dynamic environments: A survey," *ACM Computing Surveys (CSUR)*, Vol. 51, No. 2, pp. 1–36, 2018.

[20] J. Sola, "Towards visual localization, mapping and moving objects tracking by a mobile robot: A geometric and probabilistic approach," *PhD Dissertation*, École Doctorale Systèmes, Docteur de l'Institut National Politechnique de Toulouse, 2007.

[21] J. Sola, T. Vidal-Calleja, J. Civera, and J. M. M. Montiel, "Impact of landmark parametrization on monocular EKF-SLAM with points and lines," *International Journal of Computer Vision*, Vol. 97, No. 3, pp. 339–368, 2012.



S. Badalkhani received his B.Sc. and M.Sc. degree in Electrical Engineering in Control Engineering from Sahand University of Technology and Tabriz University, Tabriz, in 2009 and 2012, respectively. He joined the University of Birjand, Iran in 2014 as a Ph.D. Candidate. His research interests include simultaneous localization and mapping, path planning and control.

R. Havangi received his M.Sc. and Ph.D. degrees from the K. N. Toosi University of Technology, Tehran, Iran, in 2003 and 2012, respectively. He is currently an Associate Professor of control systems with the Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran. His main research interests are inertial navigation, integrated navigation, estimation and filtering, evolutionary filtering, simultaneous localization and mapping, fuzzy, neural network, and soft computing.