frontiers | Frontiers in Robotics and AI

TYPE Original Research
PUBLISHED 11 January 2024
DOI 10.3389/frobt.2023.1291672

Simultaneous localization and mapping in a multi-robot system in a dynamic environment with unknown initial correspondence

Hadiseh Malakouti-Khah[1],
Nargess Sadeghzadeh-Nokhodberiz[1]* and Allahyar Montazeri[2]*

[1]Department of Control Engineering, Qom University of Technology, Qom, Iran, [2]School of Engineering, Lancaster University, Lancaster, United Kingdom

OPEN ACCESS

EDITED BY
Alessandro Ridolfi,
University of Florence, Italy

REVIEWED BY
Alessandro Bucci,
University of Florence, Italy
Roohollah Barzamini,
Islamic Azad University Central Tehran Branch, Iran
Maral Partovibakhsh,
Islamic Azad University System, Iran

*CORRESPONDENCE
Nargess Sadeghzadeh-Nokhodberiz,
✉ sadeghzadeh@qut.ac.ir
Allahyar Montazeri,
✉ a.montazeri@lancaster.ac.uk

RECEIVED 09 September 2023
ACCEPTED 11 December 2023
PUBLISHED 11 January 2024

CITATION
Malakouti-Khah H, Sadeghzadeh-Nokhodberiz N and Montazeri A (2024), Simultaneous localization and mapping in a multi-robot system in a dynamic environment with unknown initial correspondence.
Front. Robot. AI 10:1291672.
doi: 10.3389/frobt.2023.1291672

A basic assumption in most approaches to simultaneous localization and mapping (SLAM) is the static nature of the environment. In recent years, some research has been devoted to the field of SLAM in dynamic environments. However, most of the studies conducted in this field have implemented SLAM by removing and filtering the moving landmarks. Moreover, the use of several robots in large, complex, and dynamic environments can significantly improve performance on the localization and mapping task, which has attracted many researchers to this problem more recently. In multi-robot SLAM, the robots can cooperate in a decentralized manner without the need for a central processing center to obtain their positions and a more precise map of the environment. In this article, a new decentralized approach is presented for multi-robot SLAM problems in dynamic environments with unknown initial correspondence. The proposed method applies a modified Fast-SLAM method, which implements SLAM in a decentralized manner by considering moving landmarks in the environment. Due to the unknown initial correspondence of the robots, a geographical approach is embedded in the proposed algorithm to align and merge their maps. Data association is also embedded in the algorithm; this is performed using the measurement predictions in the SLAM process of each robot. Finally, simulation results are provided to demonstrate the performance of the proposed method.

KEYWORDS

SLAM, multi-robot SLAM, Fast-SLAM, dynamic environments, moving landmarks, map merging

1 Introduction

Simultaneous localization and mapping (SLAM) is one of the most important developments in the field of robotics, enabling robots operating in GPS-denied environments to perceive their surroundings and localize themselves within the identified map. This is especially useful for industries such as underground mining and nuclear decommissioning, where mobile robots are required to explore and complete various tasks, such as maintenance, inspection, and transportation, in environments that are inaccessible and hazardous for humans. These missions are accomplished either using a single robot in isolation (Montazeri et al., 2021) or using a team of cooperative robots (Burrell et al., 2018). The robot or robots need to cope with the time-varying, restricted, uncertain, and unstructured nature of the environment to achieve planning and execution of the necessary tasks. This in turn requires the design and development of advanced motion control and navigation algorithms, along with strong cognitive capabilities in order for the robot to perceive the surrounding environment effectively. The use of both single- and multi-robot platforms can be advantageous, depending on the specific application and environment.

SLAM-based navigation refers to the techniques by which the robots simultaneously localize themselves in an unknown environment where a map is required. This technique can be applied to single-robot (Debeunne and Vivet, 2020) or multi-robot (Almadhoun et al., 2019) systems, as well as in either static (Kretzschmar et al., 2010) or dynamic (Saputra et al., 2018) environments.

In a static environment, landmarks are fixed, which means that all objects in the environment are stationary and none are moving (Stachniss et al., 2016). A dynamic environment is an environment in which objects are moving, such as an environment that includes humans, robots, and other means of transportation. In a dynamic environment, the SLAM process becomes more complex due to continuous changes in the environment (such as moving objects, changing colors, and combinations of shadows), and this makes the problem more challenging, for example, by creating a mismatch between previous and new data (Li et al., 2021).

Several successful solutions have been presented for the single-robot and multi-robot SLAM problem in static environments (Saeedi et al., 2016; Lluvia et al., 2021). However, the SLAM problem in a dynamic environment is more complicated due to the requirements for simultaneous detection, classification, and tracking of moving landmarks. In recent years, single-robot and multi-robot SLAM problems in dynamic environments have attracted the attention of many researchers. In (Vidal et al., 2015; Badalkhani and Havangi, 2021), solutions for solving the single-robot and multi-robot SLAM problems in dynamic environments are presented. In (Vidal et al., 2015), a method is presented for SLAM in which only static (fixed) landmarks are considered and moving ones are omitted from the map. The paper uses an outlier filter and separates fixed and moving landmarks, which are included in a set of negligible signs; the authors only use fixed landmarks to implement their solution to the SLAM problem and do not consider moving landmarks.

In (Badalkhani and Havangi, 2021), a solution for the multi-robot SLAM problem in a dynamic environment is presented. The proposed SLAM-based method is developed using a Kalman filter (KF); it detects moving landmarks based on possible location constraints and expected landmark area, which enables detection and filtering of moving landmarks. The main goal of (Badalkhani and Havangi, 2021) is the development of a new method to identify the moving parts of the environment and remove them from the map.

There are also some pieces of research implementing the SLAM problem in a dynamic environment through object recognition or using image processing techniques. However, some of these methods, such as those presented in (Liu et al., 2019; Pancham et al., 2020; Chen et al., 2021; Theodorou et al., 2022; Guan et al., 2023), also aim to remove moving landmarks found in the dynamic environment.

As mentioned, researchers who have investigated the problem of SLAM in a dynamic environment using one or more robots have mostly attempted to implement SLAM through the elimination of the moving landmarks, with only fixed ones being retained for the mapping procedure, which leads to the failure of the method when the environment is complex, with many moving objects, or in the absence of fixed objects.

In this paper, in contrast with previous work, a decentralized multi-robot modified Fast-SLAM-based algorithm is designed for use in a dynamic environment where all the landmarks are moving. It is worth mentioning that, in Fast-SLAM (Montemerlo et al., 2002), particle filtering (PF) (Godsill, 2019) is used to handle the non-linear kinematics of the robots via the Monte Carlo technique. Fast-SLAM is a filtering-based approach to SLAM decomposed into a robot localization problem, and a collection of landmark estimation problems conditioned on the robot pose estimate. This can be obtained using Bayes' rule combined with the statistical independence of landmark positions which leads to Rao Blackwellized particle filtering (RBPF) SLAM (Sadeghzadeh-Nokhodberiz et al., 2021).

In this study, we have simplified the multi-robot Fast-SLAM problem by considering two wheel-based robots in a dynamic environment with two moving landmarks with a known kinematic model. The proposed method can easily be extended to cases with more robots and landmarks. Each robot searches the environment and observes it with its onboard lidar sensor. Here, it is assumed that the kinematic models of the robots and landmarks are known, which makes the use of a modified Fast-SLAM method meaningful. This method is based on observer use of a particle filter and extended Kalman filter (EKF). However, several modifications of the normal Fast-SLAM method are required. The first modification is adding a prediction step to the EKF used for mapping using the kinematic model of the landmarks. Adding this step to the algorithm does not obviate the need for initialization, as the initial values of the landmarks' positions are assumed to be unknown. This prediction step is performed using the landmark's kinematic model at every sample time after the first visit to the landmark, even if the landmark has not been visited again by the lidar sensor. The second modification is that, in contrast with the normal approaches to data association with static landmarks, data association is performed based on the predicted measurement obtained from the predicted map. When data association is performed based on the predicted position of a landmark, it is less likely that a previously visited landmark will be wrongly diagnosed as a new one due to its movement. As the third modification, and due to the unknown initial correspondence, coordinate alignment and map-merging are added to the algorithm. In this step, the relative transformation matrix of the robots' inertial frames is computed when the robots meet each other, using a geographical approach as presented in (Zhou and Roumeliotis, 2006; Romero and Costa, 2010) to compute this transformation matrix, and the maps obtained in the coordinate system of each robot are then fused and merged.

In summary, the novelty of this paper can be considered to be the extension of the Fast-SLAM algorithm in several aspects, as follows:

1. Addition of a prediction step to the EKF used for mapping, based on the kinematic model of the landmarks after the first visit to each landmark and at every sample time.
2. Data association based on the predicted measurements obtained from the predicted map.

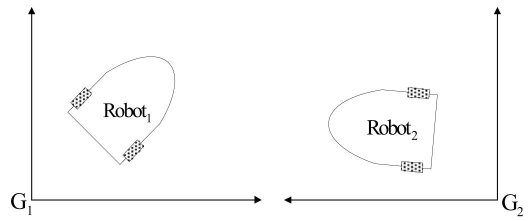

FIGURE 1
The robots and their corresponding coordinates.

3. Addition of coordinate alignment and map-merging to the algorithm when the robots meet each other.

The paper is organized as follows. In Section 2, a system overview is presented, including a kinematic model of the robots and landmarks as well as the lidar measurement model. The modified Fast-SLAM algorithm for the dynamic environment is proposed in Section 3. Section 4 describes the coordinate alignment and map-merging problem. Simulation results are presented in Section 5, and conclusions are provided in Section 6.

2 System overview

In this paper, for the sake of simplicity, two mobile robots and two dynamic landmarks have been considered. As an example, differential drive mobile robots such as the Pioneer P3-DX (Zaman et al., 2011) moving in a dynamic environment were selected for the study. The robots are equipped with lidar and IMU sensors. In this section, the kinematic models of the robots and landmarks and the measurement model of the lidar are assumed to be known.

It is worth mentioning that IMUs can provide information on linear acceleration and angular velocity. However, linear speed can be obtained through the integration of linear acceleration.

2.1 Kinematic model of the robots

The kinematic model for the two differential drive Pioneer P3-DX mobile robots can be written as follows:

$$\begin{bmatrix} x_{r_i}(k+1) \\ y_{r_i}(k+1) \\ \varphi_{r_i}(k+1) \end{bmatrix} = \begin{bmatrix} x_{r_i}(k) + v(k)\cos\left(\varphi_{r_i}(k) + \omega(k)\Delta_t\right) \\ y_{r_i}(k) + v(k)\sin\left(\varphi_{r_i}(k) + \omega(k)\Delta_t\right) \\ \varphi_{r_i}(k) + \omega(k)\Delta_t \end{bmatrix} + \boldsymbol{\upsilon}_{r_i}(k), \quad (1)$$

where x_{r_i} and y_{r_i} are the i^{th} positions of the robot on the x and y axes and φ_{r_i} is its corresponding direction relative to the x-axis for $i = 1, 2$. The coordinate system of the first robot is G_1 and that of the second robot is G_2, as depicted in Figure 1. In (1), $\boldsymbol{\upsilon}_{r_i}(k)$ is zero-mean non-Gaussian process noise with a known probability density function (PDF). Additionally, Δ_t is the sampling time of the process and the variables $v(k)$ and $\omega(k)$ are the linear and the angular velocities of the robot, which are assumed to be known.

Eq. 1 can be rewritten in the general non-linear state space form as follows:

$$\mathbf{x}_{r_i}(k+1) = \mathbf{f}_{r_i}\left(\mathbf{x}_{r_i}(k), \mathbf{u}_{r_i}(k)\right) + \boldsymbol{\upsilon}_{r_i}(k), \quad (2)$$

where $\mathbf{x}_{r_i}(k) = \left[x_{r_i}(k) \quad y_{r_i}(k) \quad \varphi_{r_i}(k)\right]^T$, $\mathbf{u}_{r_i}(k) = \left[\omega(k) \quad v(k)\right]^T$, and $\mathbf{f}_{r_i}\left(\mathbf{x}_{r_i}(k), \mathbf{u}_{r_i}(k)\right)$ is the kinematic model presented in (1).

2.2 Kinematic model of the landmarks

In this study, both landmarks are considered to be moving, with the following kinematic model:

$$\begin{bmatrix} x_{L_j,r_i}(k+1) \\ y_{L_j,r_i}(k+1) \\ \varphi_{L_j,r_i}(k+1) \end{bmatrix} = \begin{bmatrix} x_{L_j,r_i}(k) + v_{L_j}(k)\cos\left(\varphi_{L_j,r_i}(k) + \omega_{L_j}(k)\Delta_t\right) \\ y_{L_j,r_i}(k) + v_{L_j}(k)\sin\left(\varphi_{L_j,r_i}(k) + \omega_{L_j}(k)\Delta_t\right) \\ \varphi_{L_j,r_i}(k) + \omega_{L_j}(k)\Delta_t \end{bmatrix} + \boldsymbol{\upsilon}_{L_j,r_i}(k), \quad (3)$$

where x_{L_j,r_i} and y_{L_j,r_i} are the j^{th} positions of the landmark on the x and y axes, respectively, and φ_{L_j,r_i} is its corresponding direction with respect to the x-axis for $j = 1, 2$, presented in G_i for $i = 1, 2$. Variables $v_{L_j}(k)$ and $\omega_{L_j}(k)$ are the linear velocity and angular velocity of landmark L_j, respectively; Δ_t is the sampling time of the process, and k refers to the sample number. It is assumed that $v_{L_j}(k)$ and $\omega_{L_j}(k)$ are known parameters. Moreover, $\boldsymbol{\upsilon}_{L_j,r_i}(k)$ is a zero-mean Gaussian noise vector: that is, $\boldsymbol{\upsilon}_{L_j,r_i}(k) \sim N\left(0, \mathbf{R}_{L_j,r_i}\right)$, where $N(.,.)$ refers to the Gaussian PDF.

By writing (3) in the general non-linear state space model, we obtain

$$\mathbf{x}_{L_j,r_i}(k+1) = \mathbf{f}_{L_j,r_i}\left(\mathbf{x}_{L_j,r_i}(k), \mathbf{u}_{L_j}(k)\right) + \boldsymbol{\upsilon}_{L_j,r_i}(k), \quad (4)$$

where $\mathbf{x}_{L_j,r_i}(k) = \left[x_{L_j,r_i}(k) \quad y_{L_j,r_i}(k) \quad \varphi_{L_j,r_i}(k)\right]^T$, $\mathbf{u}_{L_j}(k) = \left[\omega_{L_j}(k) \quad v_{L_j}(k)\right]^T$, and $\mathbf{f}_{L_j,r_i}\left(\mathbf{x}_{L_j,r_i}(k), \mathbf{u}_{L_j}(k)\right)$ is the kinematic model presented in (4).

2.3 Lidar measurement model

Lidar sensor output includes the distance and angle of each robot relative to the observed landmarks. Each observation by each robot of the landmarks that are present can be expressed by the following observation model:

$$\mathbf{z}_{L_j,r_i}(k) = \begin{bmatrix} r_{L_j,r_i}(k) \\ \varphi_{L_j,r_i}(k) \end{bmatrix} + \boldsymbol{\upsilon}_{ij},$$

$$r_{L_j,r_i}(k) = \sqrt{\left(x_{r_i}(k) - x_{L_j}(k)\right)^2 + \left(y_{r_i}(k) - y_{L_j}(k)\right)^2},$$

$$\varphi_{L_j,r_i}(k) = \arctan 2\left(\frac{y_{r_i}(k) - y_{L_j}(k)}{\left(x_{r_i}(k) - x_{L_j}(k)\right)}\right), \quad (5)$$

where $\mathbf{z}_{L_j,r_i}$ represents the i^{th} robot observation vector of the j^{th} landmark in the environment, r_{ij} represents the distance of the robot from the observed landmark, and φ_{ij} is the corresponding angle of

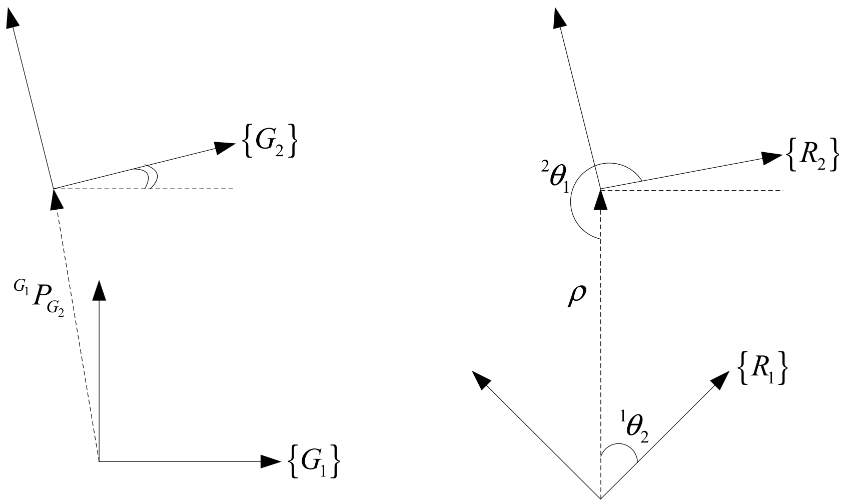

FIGURE 2
The relationships between different coordinates.

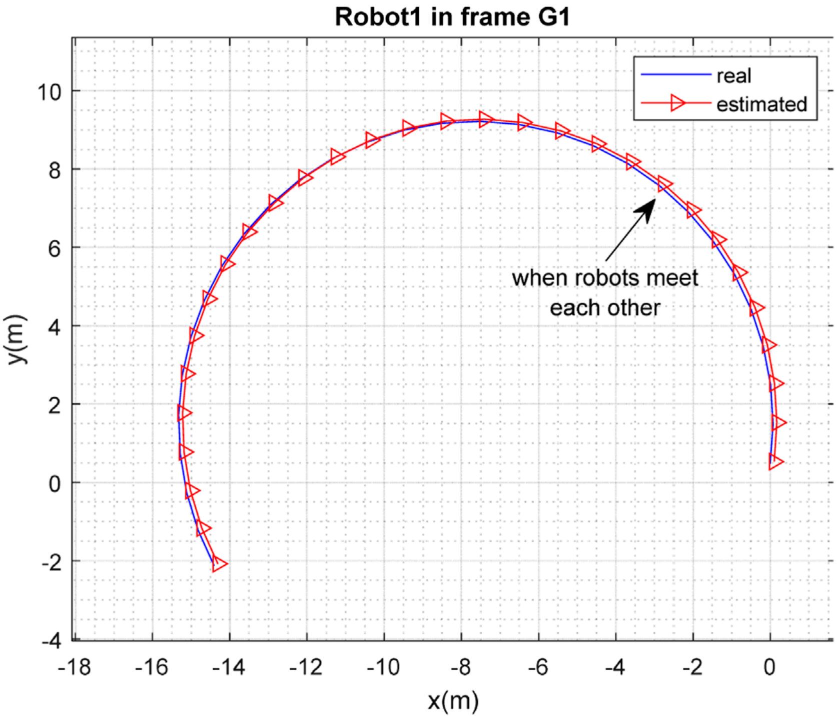

FIGURE 3
Real and estimated position of robot R_1 in the frame G_1.

the robot relative to the landmark. Additionally, $\mathbf{v}_{ij}$ is a zero-mean Gaussian measurement noise vector with the covariance matrix of $\mathbf{R}_{z_{ij}}$. Moreover, (5) can be rewritten in the general non-linear measurement model as follows:

$$\mathbf{z}_{L_j,r_i}(k) = \mathbf{h}\left(\mathbf{x}_{r_i}(k), \mathbf{x}_{L_j}(k)\right) + \mathbf{v}_{ij}. \qquad (6)$$

3 Modified Fast-SLAM with dynamic landmarks

Robots R_1 and R_2 explore the environment, with each of them performing the SLAM process from the corresponding frames of G_1 and G_2, respectively; and an independent map of the environment is thereby created by each of them. Information on

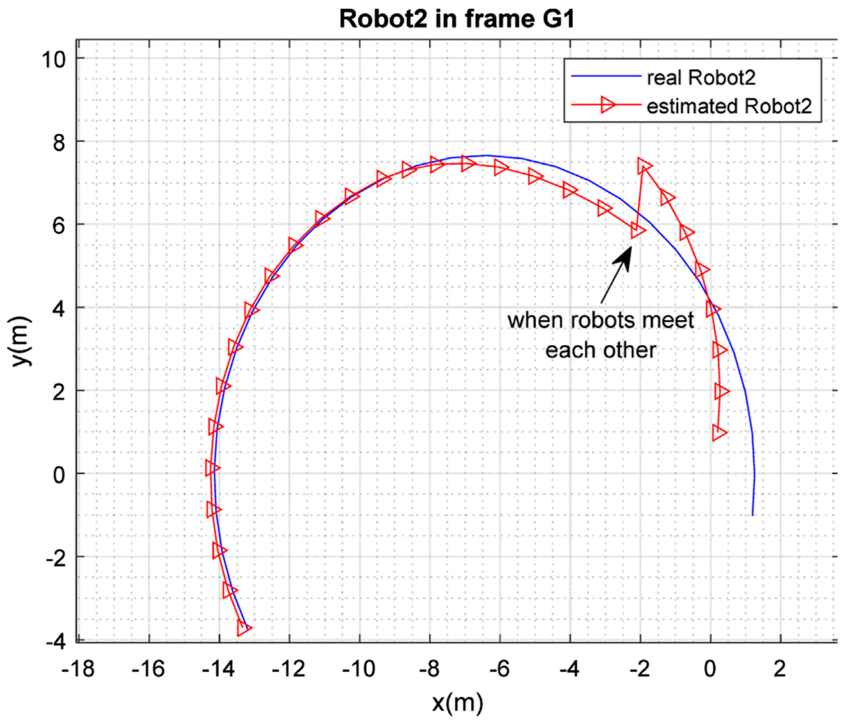

FIGURE 4
Real and estimated position of robot R_2 in the frame G_1.

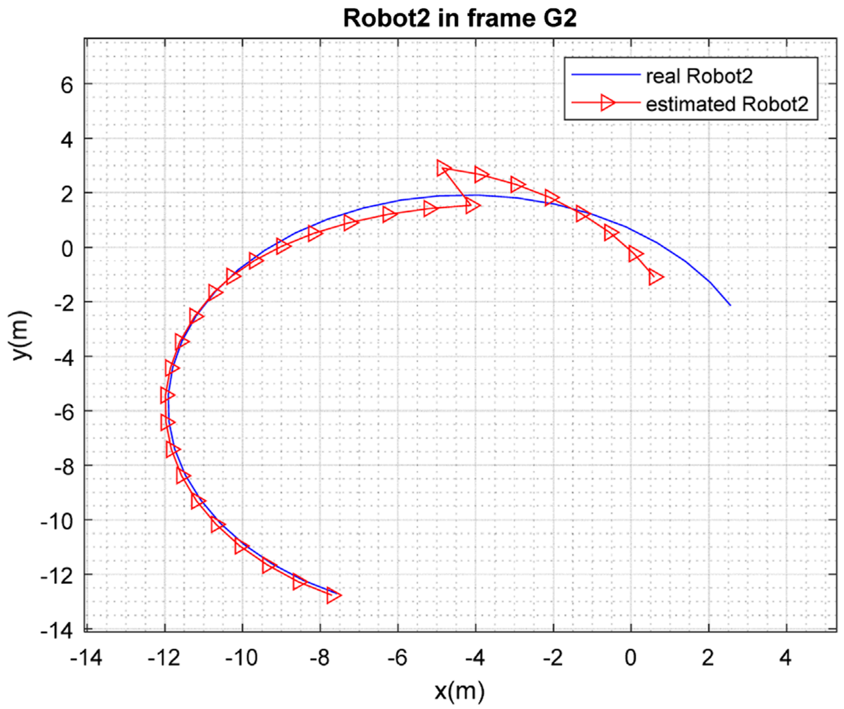

FIGURE 5
Real and estimated position of robot R_2 in the frame G_2.

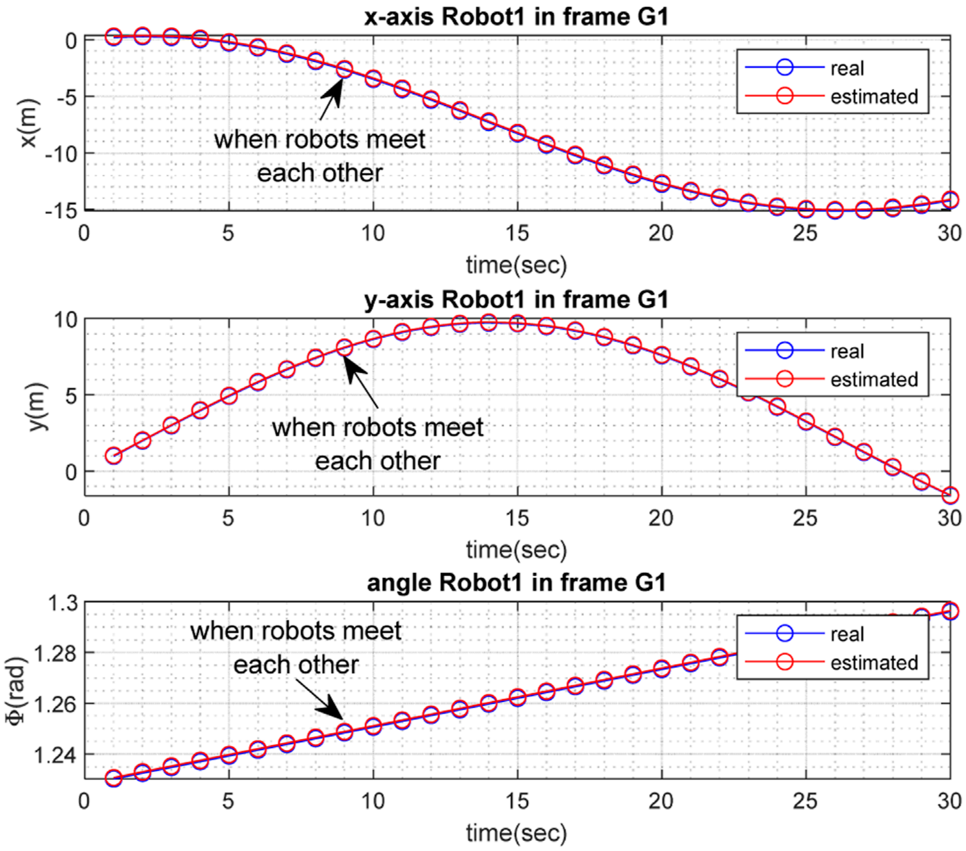

FIGURE 6
Real and estimated values of x_{r_1}, y_{r_1}, and φ_{r_1} in G_1.

the distances and angles of the robots relative to the observed landmarks is provided by the lidar sensor in each robot. Each robot then uses a modified Fast-SLAM algorithm. In the proposed modified Fast-SLAM, Rao Blackwellized particle filtering (RBPF) is employed to estimate both vehicle trajectories and landmark positions, in which each landmark is estimated with EKF and PFs are employed to generate particles only used for trajectory estimation. In the remainder of this section, this process is explained, with the application of a modification in EKF due to the presence of moving landmarks with known kinematics. The general steps of Fast-SLAM consist of particle filtering with embedded extended Kalman filtering for the mapping; the data association process; and the map-merging procedure. These steps, with the necessary modifications, are explained for our problem in the following subsections.

3.1 Particle generation

An important and initial step in the PF is particle generation. To this end, particles are generated using the known PDF of $\boldsymbol{\upsilon}_{r_i}(k)$ through Monte Carlo simulation, where particles of $\boldsymbol{\upsilon}^l_{r_i}(k-1)$ for $l=1,\ldots,N$ are generated and replaced in the motion model of the i^{th} robot (1), and therefore $x^l_{r_i}(k)$, $y^l_{r_i}(k)$, and $\varphi^l_{r_i}(k)$ for $l=1,\ldots,N$ and $i=1,2$ are generated. It is worth mentioning that N refers to the number of particles. This process can be formulated in the following general form according to (2):

$$\mathbf{x}^l_{r_i}(k)=\mathbf{f}_{r_i}\left(\mathbf{x}^l_{r_i}(k-1),\mathbf{u}_{r_i}(k-1)\right)+\boldsymbol{\upsilon}^l_{r_i}(k-1). \tag{7}$$

3.2 Mapping

Each robot separately performs mapping after the particle generation process using EKF, as explained in the below subsections. After the first meeting of the robots and coordinate transformation, the maps are merged, as explained in detail in subsequent subsections.

3.2.1 EKF for mapping

- Initialization

After the particle generation process as explained in the previous subsection, when the j^{th} landmark is visited for the first time, initialization is performed using the inverse sensor model for

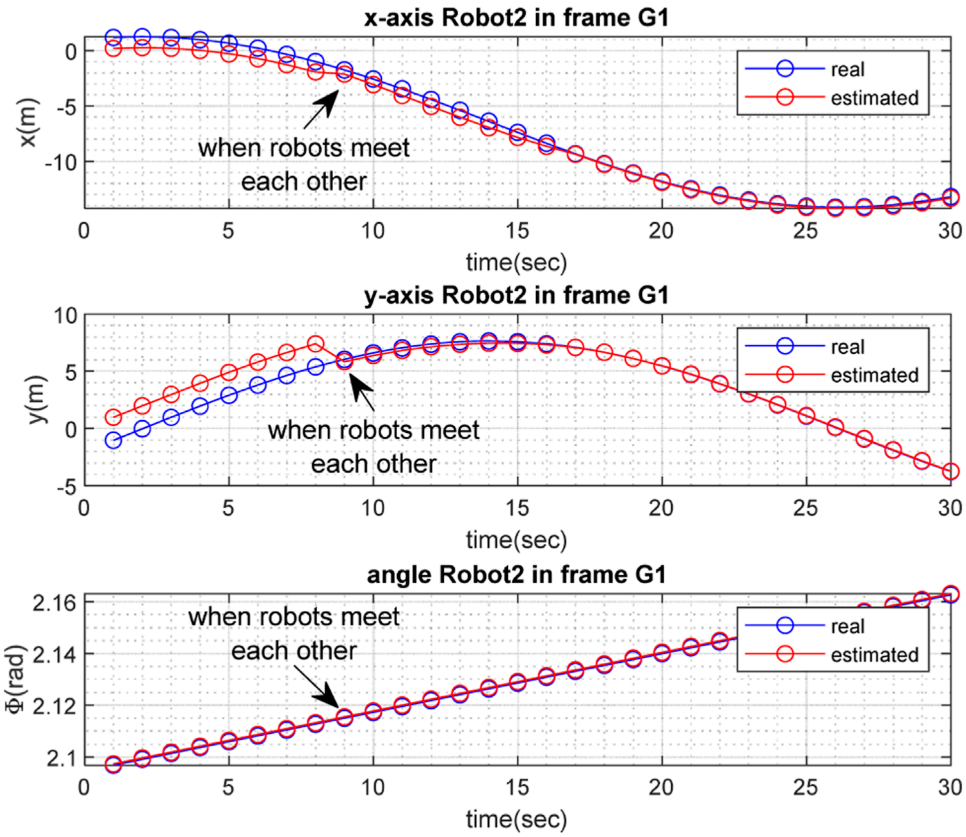

FIGURE 7
Real and estimated values of x_{r_2}, y_{r_2}, and φ_{r_2} in G_1.

each particle, as follows, if it has been visited at sample instant k (neglecting the measurement noise):

$$\begin{aligned}
\hat{x}^{l+}_{L_j,r_i}(k) &= x^{l}_{r_i}(k) + r_{L_j,r_i}(k)\cos\left(\varphi_{L_j,r_i}(k) + \varphi^{l}_{r_i}(k)\right),\\
\hat{y}^{l+}_{L_j,r_i}(k) &= y^{l}_{r_i}(k) + r_{L_j,r_i}(k)\sin\left(\varphi_{L_j,r_i}(k) + \varphi^{l}_{r_i}(k)\right),\\
\hat{\varphi}^{l+}_{L_j,r_i}(k) &= \varphi_{L_j,r_i}(k) + \varphi^{l}_{r_i}(k),
\end{aligned} \tag{8}$$

Where $\hat{x}^{l+}_{L_j,r_i}(k)$, $\hat{y}^{l+}_{L_j,r_i}(k)$, and $\hat{\varphi}^{l+}_{L_j,r_i}(k)$ generally refer to the posterior estimates. Although the posterior estimates are obtained in the update step of the Bayesian filtering procedure, as the initial values will be used for prior estimates in the prediction step, a similar notation has been used. (8) can be rewritten in the following general form using (6):

$$\hat{\mathbf{x}}^{l+}_{L_j,r_i}(k) = \mathbf{h}^{-1}\left(\mathbf{z}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)\right). \tag{9}$$

Moreover, let predicted covariance matrix $\mathbf{P}^{l+}_{L_j,r_i} = K\mathbf{I}_3$, where K is a small real number and $\mathbf{I}_3$ is an identity matrix.

- Prediction step

After the first visit to the j^{th} landmark, prior estimates (predicted estimates) $\hat{x}^{l-}_{L_j,r_i}(k)$, $\hat{y}^{l-}_{L_j,r_i}(k)$, and $\hat{\varphi}^{l-}_{L_j,r_i}(k)$ are computed based on the landmark kinematics (3) at each time step for each particle l. Additionally, it is necessary to compute the predicted covariance matrix. This step is the first modification of our method compared with normal Fast-SLAM with static landmarks, in which no kinematics exist for the landmarks. In other words:

$$\hat{\mathbf{x}}^{l-}_{L_j,r_i}(k) = \mathbf{f}_{L_j,r_i}\left(\hat{\mathbf{x}}^{l+}_{L_j,r_i}(k-1), \mathbf{u}_{L_j}(k-1)\right). \tag{10}$$

$$\mathbf{P}^{l-}_{L_j,r_i}(k) = \mathbf{A}^{l+}_{L_j,r_i}\mathbf{P}^{l+}_{L_j,r_i}(k-1)\left(\mathbf{A}^{l+}_{L_j,r_i}\right)^T + \mathbf{R}_{L_j,r_i}, \tag{11}$$

where in (11) $\mathbf{P}^{l-}_{L_j,r_i}$ is the predicted covariance matrix and $\mathbf{P}^{l+}_{L_j,r_i}$ is the updated one. Moreover:

$$\mathbf{A}^{l+}_{L_j,r_i} = \left.\frac{\partial \mathbf{f}_{L_j,r_i}\left(\mathbf{x}_{L_j,r_i}(k-1), \mathbf{u}_{L_j,r_i}(k-1)\right)}{\partial \mathbf{x}_{L_j,r_i}(k-1)}\right|_{\hat{\mathbf{x}}^{l+}_{L_j,r_i}(k-1), \mathbf{u}_{L_j}(k-1)}. \tag{12}$$

- Data association

Data association is one of the important issues for SLAM in dynamic environments. For this purpose, in this paper, the measurement prediction is simply computed as follows:

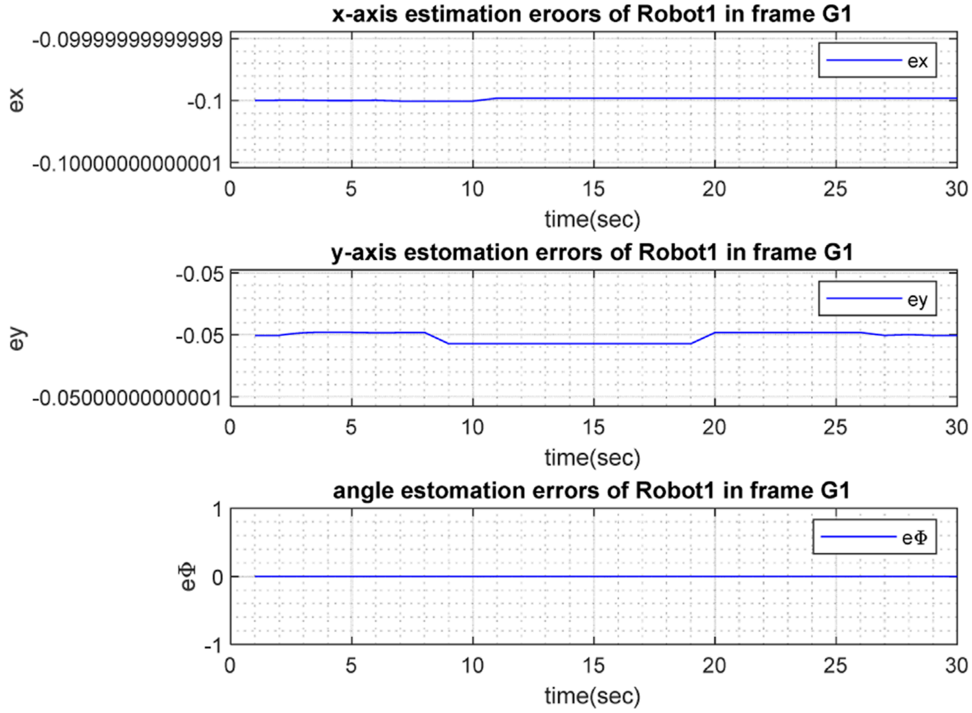

FIGURE 8
Estimation errors $e_{x_{r_1}}$, $e_{y_{r_1}}$, and $e_{\varphi_{r_1}}$ in G_1.

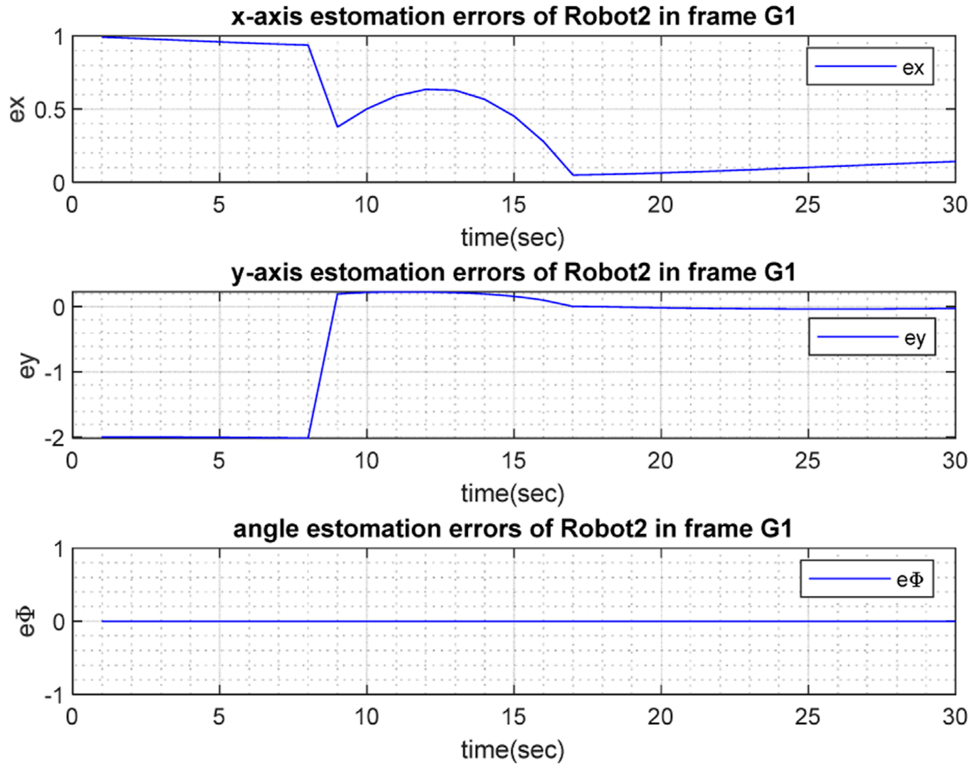

FIGURE 9
Estimation errors $e_{x_{r_2}}$, $e_{y_{r_2}}$, and $e_{\varphi_{r_2}}$ in G_1.

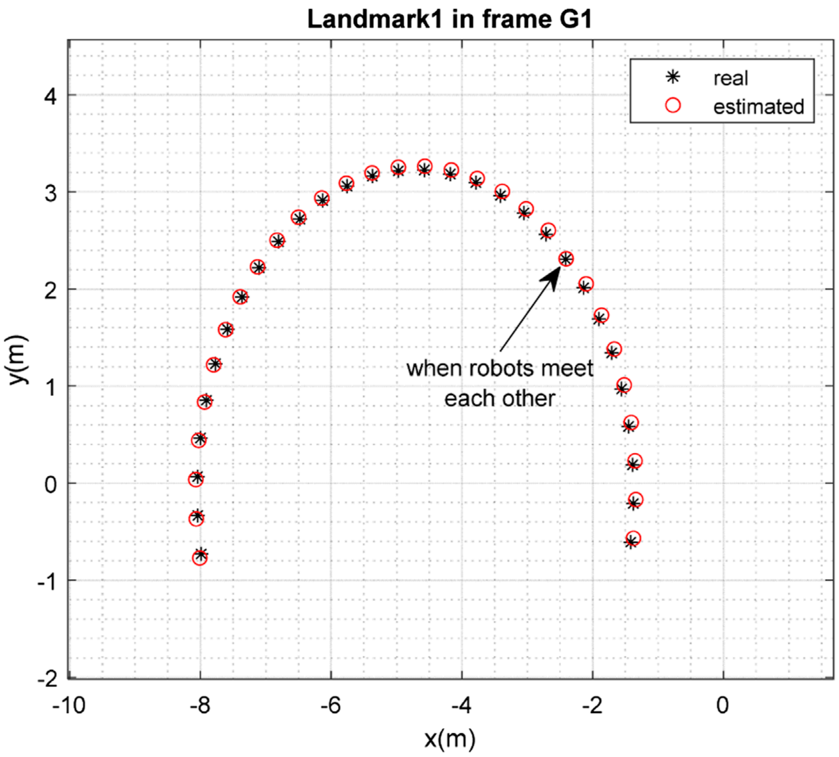

FIGURE 10
Real position of landmark L_1 and position estimated by robot R_1 in the frame G_1.

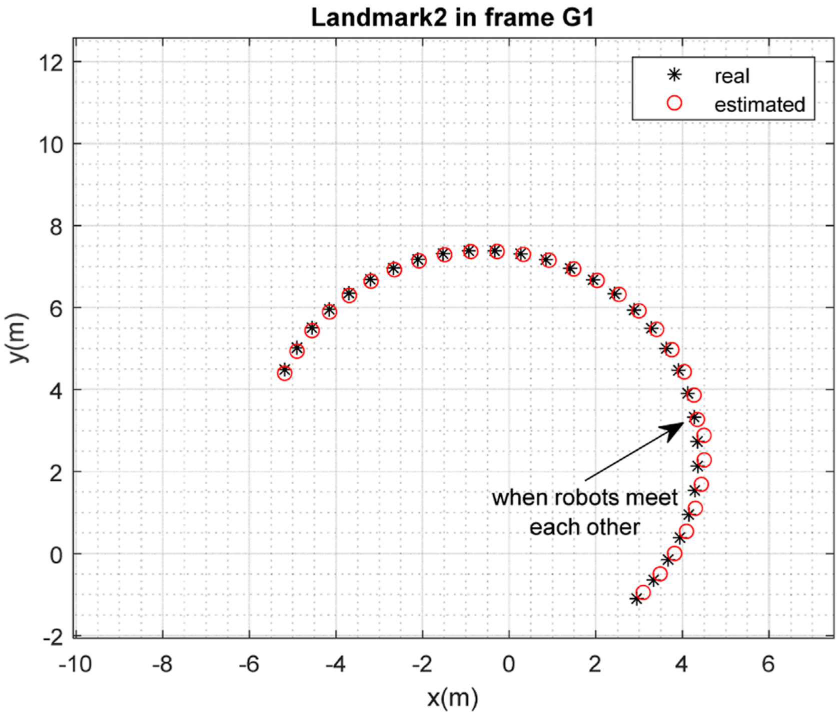

FIGURE 11
Real position of landmark L_2 and position estimated by robot R_1 in the frame G_1.

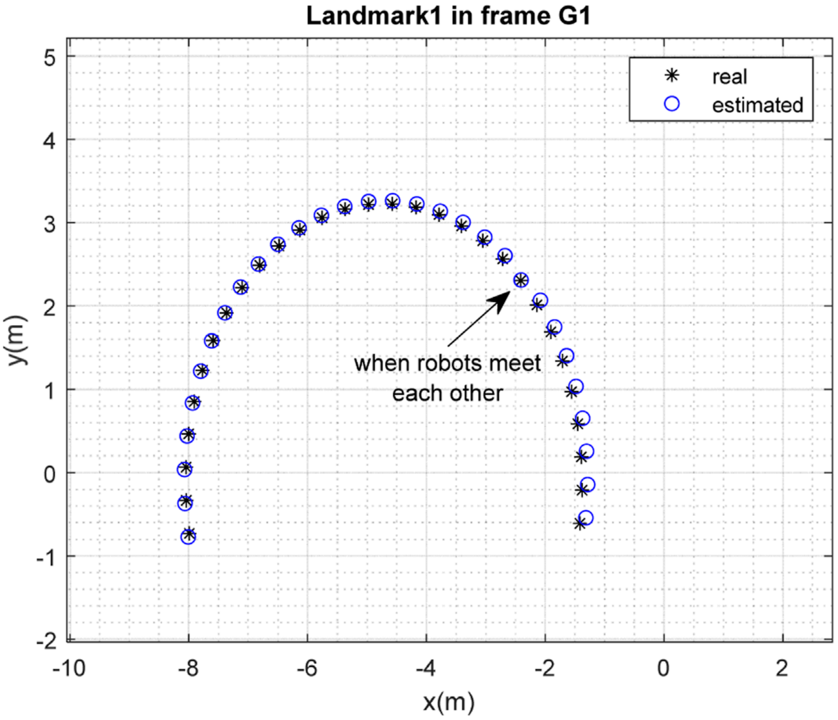

FIGURE 12
Real position of landmark L_1 and position estimated by robot R_2 in the frame G_1.

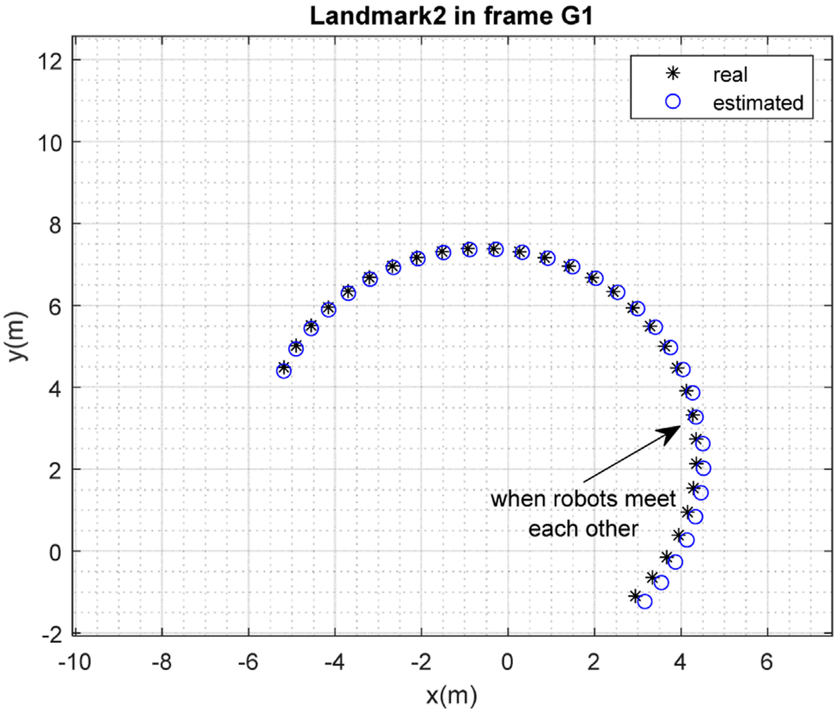

FIGURE 13
Real position of landmark L_2 and position estimated by robot R_2 in the frame G_1.

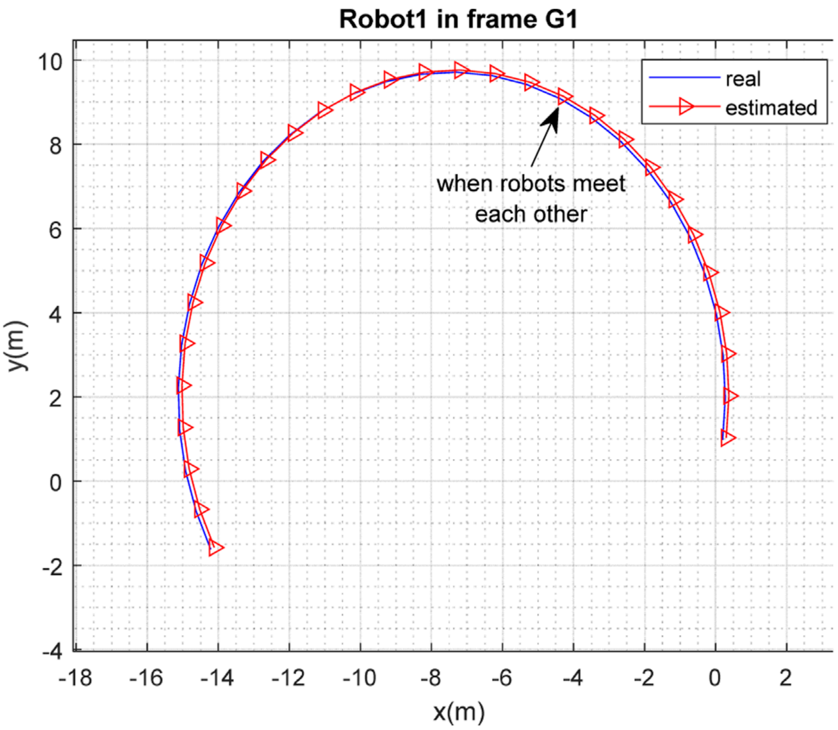

FIGURE 14
Real and estimated position of robot R_1 in the frame G_1 (with different initial conditions).

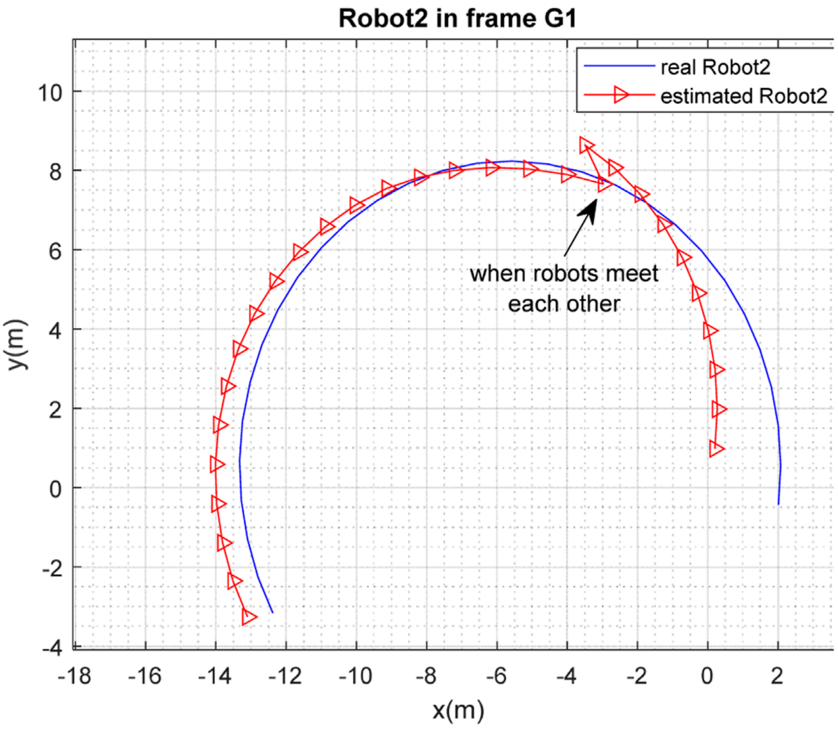

FIGURE 15
Real and estimated position of robot R_2 in the frame G_1 (with different initial conditions).

$$\begin{aligned}
\hat{\mathbf{z}}^{i-}_{L_j,r_i}(k) &= \begin{bmatrix} \hat{r}^{l-}_{L_j,r_i}(k) \\ \hat{\varphi}^{l-}_{L_j,r_i}(k) \end{bmatrix}, j = 1,2 \\
\hat{r}^{l-}_{L_j,r_i}(k) &= \sqrt{\left(x^{l}_{r_i}(k) - \hat{x}^{l-}_{L_j,r_i}(k)\right)^2 + \left(y^{l}_{r_i}(k) - \hat{y}^{l-}_{L_j,r_i}(k)\right)^2}, \\
\hat{\varphi}^{l-}_{L_j,r_i}(k) &= \arctan 2\left(\frac{y^{l}_{r_i}(k) - \hat{y}^{l-}_{L_j,r_i}(k)}{\left(x^{l}_{r_i}(k) - \hat{x}^{l-}_{L_j,r_i}(k)\right)}\right),
\end{aligned} \tag{13}$$

where $\hat{r}^{i-}_{L_j,r_i}(k)$ represents the estimated distance of the i^{th} robot from the j^{th} landmark, and $\hat{\varphi}^{l-}_{L_j,r_i}(k)$ represents the estimated angle of the i^{th} robot relative to j^{th} landmark. Next, the Euclidean distance between the real observations, $\mathbf{z}_{L_j,r_i}(k)$, and the estimated ones is simply calculated:

$$r^{l}_{d_{ij}} = \left\| \mathbf{z}_{L_j,r_i}(k) - \hat{\mathbf{z}}^{l-}_{L_j,r_i}(k) \right\|. \tag{14}$$

For data association, the condition $r^{l}_{d_{ij}} < \mu$ for $j = 1,2$ is first checked, where μ is a predefined threshold. Subsequently, if this condition is established for both observed landmarks, the smaller value is selected.

- Update step

If the landmark j is observed, this step is performed in a similar way to normal Fast-SLAM, which consists of gain computation, state update, and covariance update, as follows:

$$\begin{aligned}
\mathbf{S}^{l}_{L_j,r_i}(k) &= \mathbf{H}^{l-}_{L_j,r_i}(k)\mathbf{P}^{l-}_{L_j,r_i}(k)\left(\mathbf{H}^{l-}_{L_j,r_i}(k)\right)^T + \mathbf{R}_{z_{ij}}, \\
\mathbf{K}^{l}_{L_j,r_i}(k) &= \mathbf{P}^{l-}_{L_j,r_i}(k)\left(\mathbf{H}^{l-}_{L_j,r_i}(k)\right)^T\left(\mathbf{S}^{l}_{L_j,r_i}(k)\right)^{-1}, \\
\hat{\mathbf{x}}^{l+}_{L_j,r_i}(k) &= \hat{\mathbf{x}}^{l-}_{L_j,r_i}(k) + \mathbf{K}^{l}_{L_j,r_i}(k)\left(\mathbf{z}_{L_j,r_i}(k) - \hat{\mathbf{z}}^{l}_{L_j,r_i}(k)\right), \\
\mathbf{P}^{l+}_{L_j,r_i}(k) &= \left(\mathbf{I} - \mathbf{K}^{l}_{L_j,r_i}(k)\mathbf{H}^{l-}_{L_j,r_i}(k)\right)\mathbf{P}^{l-}_{L_j,r_i}(k),
\end{aligned} \tag{15}$$

where $\mathbf{S}^{l}_{L_j,r_i}$ is the residual covariance, $\mathbf{K}^{l}_{L_j,r_i}$ is the Kalman gain, $\hat{\mathbf{x}}^{l+}_{L_j,r_i}$ is the updated map of the j^{th} landmark, and $\mathbf{P}^{l+}_{L_j,r_i}$ is the updated covariance matrix. Additionally, $\mathbf{H}^{l-}_{L_j,r_i}$ is computed as:

$$\mathbf{H}^{l-}_{L_j,r_i} = \left.\frac{\partial \mathbf{h}\left(\mathbf{x}_{r_i}(k), \mathbf{x}_{L_j,r_i}(k)\right)}{\partial \mathbf{x}_{L_j,r_i}(k-1)}\right|_{\hat{\mathbf{x}}^{l-}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)} . \tag{16}$$

For unobserved landmarks, the predicted values are replaced with the posterior ones; in other words, $\mathbf{P}^{l+}_{L_j,r_i}(k) = \mathbf{P}^{l-}_{L_j,r_i}(k)$ and $\hat{\mathbf{x}}^{l+}_{L_j,r_i}(k) = \hat{\mathbf{x}}^{l-}_{L_j,r_i}(k)$.

3.3 Localization (weight computation and resampling)

In this step, the final localization process is performed through computation of the particles' weights and a resampling process. To this end, the normalized weights are computed as follows:

$$w^{l}_{L_j,r_i}(k) = \frac{p\left(\mathbf{z}_{L_j,r_i}(k) | \hat{\mathbf{x}}^{l+}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)\right)}{\sum_{i=1}^{N} p\left(\mathbf{z}_{L_j,r_i}(k) | \hat{\mathbf{x}}^{l+}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)\right)}, \tag{17}$$

where $w^{l}_{L_j,r_i}(k)$ is the computed weight for the l^{th} particle corresponding to the j^{th} landmark and the i^{th} robot, and $p\left(\mathbf{z}_{L_j,r_i}(k) | \hat{\mathbf{x}}^{l+}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)\right)$ refers to the conditional PDF. It is worth a reminder that, in this paper, the measurement noise vector is a Gaussian one, and therefore:

$$p\left(\mathbf{z}_{L_j,r_i}(k) | \hat{\mathbf{x}}^{l+}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)\right) = N\left(\hat{\mathbf{z}}^{l+}_{L_j,r_i}(k), \mathbf{R}_{z_{ij}}\right), \tag{18}$$

where $\hat{\mathbf{z}}^{l+}_{L_j,r_i}(k)$ is computed in a similar way to Eq. 13, but replacing $\hat{x}^{l-}_{L_j,r_i}(k)$ and $\hat{y}^{l-}_{L_j,r_i}(k)$ with $\hat{x}^{l+}_{L_j,r_i}(k)$ and $\hat{y}^{l+}_{L_j,r_i}(k)$, obtained from the update step, respectively. Next, the resampling process is performed using $w^{l}_{L_j,r_i}(k)$, and finally a set of N equally weighted particles is generated; these are averaged to produce the final estimate at each sample time k, $\hat{\mathbf{x}}_{L_j,r_i}(k)$ and $\hat{\mathbf{x}}_{r_i}(k)$.

It is worth mentioning that if no measurement is available, the predicted particles in (7) will be used for particle generation in the next step; additionally, they are weighted equally for the final mapping and localization, $\hat{\mathbf{x}}_{L_j,r_i}(k)$ and $\hat{\mathbf{x}}_{r_i}(k)$.

4 Coordinate alignment and map-merging in multi-robot SLAM

As mentioned earlier, two robots, R_1 and R_2, explore the environment and perform modified Fast-SLAM as explained in the previous section. However, localization and mapping are performed in their corresponding coordinate frames, G_1 and G_2 respectively, and each robot is only aware of its own coordinate frame. In order to merge and fuse the separately provided maps, it is necessary for the robots to meet each other. Although there exist many rendezvous approaches to force the robots to meet each other, such as the one presented in (Roy and Dudek, 2001; De Hoog et al., 2010; Zaman et al., 2011; Meghjani and Dudek, 2012; Meghjani and Dudek, 2011), for the sake of simplicity, in this paper it is assumed that the robots travel in such a way that they can meet each other at least once during the mission. When the robots meet each other, it is possible to transform the coordinates. For this purpose, the proposed method of (Zhou and Roumeliotis, 2006) is employed in this paper.

To this end, when robots R_1 and R_2 meet using the measurements from the installed lidar sensors, the relative angles between the robots (${}^{2}\theta_1; {}^{1}\theta_2$) and the distance between the robots (ρ) are measured, where ${}^{2}\theta_1$ is the angle of the first robot as measured by the second one and ${}^{1}\theta_2$ is the angle of the second robot measured as by the first. Using the geometrical method presented in (Zhou and Roumeliotis, 2006), the rotation matrix, ${}^{G_1}P_{G_2}$, and the translation vector, ${}^{G_1}P_{G_2}$, for transforming a vector in frame G_2 to frame G_1 are computed. Readers are referred to (Zhou and Roumeliotis, 2006) for details of the proposed method. Figure 2 clarifies the geometrical relationships between the coordinates and the measurements. Following this, the map created by the second robot can be presented in G_1 as follows, and the two maps are merged.

$$\left[\hat{\mathbf{x}}^{l}_{L_j,r_2}(k)\right]_{G_1} = {}^{G_1}C_{G_2}\hat{\mathbf{x}}^{l}_{L_j,r_2}(k) + {}^{G_1}P_{G_2}, \tag{19}$$

where $\left[\hat{\mathbf{x}}^{l}_{L_j,r_2}(k)\right]_{G_1}$ refers to the coordinate of a vector in frame G_1.

Finally, all the landmarks are presented in frame G_1. If a landmark L_j has been visited by both robots, its corresponding map is fused with $\hat{\mathbf{x}}^{l}_{L_j,r_1}(k)$ through simple averaging, and it is updated to achieve a more precise map. Moreover, similarly, the trajectory of the second robot can be expressed in G_1 as a global inertial frame.

Similarly, the map created by the first robot is presented in frame G_2 using ${}^{G_2}C_{G_1}$ and ${}^{G_2}P_{G_1}$, where $\left[\hat{\mathbf{x}}^{l}_{L_j,r_1}(k)\right]_{G_2}$ is obtained as follows:

$$\left[\hat{\mathbf{x}}^{l}_{L_j,r_1}(k)\right]_{G_2} = {}^{G_2}C_{G_1}\hat{\mathbf{x}}^{l}_{L_j,r_1}(k) + {}^{G_2}P_{G_1}. \tag{20}$$

Next, $\left[\hat{\mathbf{x}}^{l}_{L_j,r_1}(k)\right]_{G_2}$ is fused with $\hat{\mathbf{x}}^{l}_{L_j,r_2}(k)$ to provide more precise mapping through averaging.

To clarify the proposed method, the general proposed algorithm for multi-robot Fast-SLAM in a dynamic environment is presented and summarized in Algorithm 1.

5 Simulation results

To evaluate the performance of the proposed multi-robot SLAM algorithm in a dynamic environment, a series of simulation studies were conducted using MATLAB m-file codes; these are reported on in this section. Two Pioneer P3-DX mobile robots, each equipped with a lidar sensor, are simulated. As mentioned earlier in the article, two moving landmarks are considered and are modeled via their kinematic models.

As mentioned earlier, robots R_1 and R_2 explore the environment and observe the landmarks within it; each of them implements the SLAM process from its corresponding frame of G_1 or G_2, respectively, and makes an independent map of the environment. The parameters in our simulation are considered as follows:

$$\omega_{L_1} = 0.2rad/\sec, \omega_{L_2} = 0.2rad/\sec, v_{L_1} = 0.4m/\sec,$$
$$v_{L_2} = 0.6m/\sec, \omega = 0.2rad/\sec, v = 1m/\sec$$

All of the sources of noise are considered to be zero-mean Gaussian noise; the covariance matrices of the process (kinematics) noise, for both landmarks and robots, are specified as $0.001\mathbf{I}_3$, and the covariance matrices of the lidar noise are specified as $0.001\mathbf{I}_2$.

Additionally: $${}^{G_2}T_{G_1} = \begin{bmatrix} 0.8142 & -0.5806 & 0 & 1 \\ 0.5806 & 0.8142 & 0 & -2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \Delta_t = 0.65 \sec$$

$$\left[\mathbf{x}_{R_1}(0)\right]_{G_1} = \left[-0.2\ -0.5\ 1.23\right]^T, \left[\mathbf{x}_{R_2}(0)\right]_{G_1} = \left[0\ 0\ 2.09\right]^T$$
$$\left[\mathbf{x}_{L_1}(0)\right]_{G_1} = \left[-1.5\ -1\ 1.22\right]^T, \left[\mathbf{x}_{L_2}(0)\right]_{G_1} = \left[2.5\ -1.5\ 2.09\right]^T$$

Once the robots meet one another, the transformation matrix between the two frames G_1 and G_2 is obtained and the maps can be fused to obtain a more precise map of the environment and achieve more precise localization performance. In the case of our simulation, the robots meet each other at 9.35 s and the estimated transformation matrix is as follows:

```
Step 0: Initialization
at each sample time $k$:
 for each robot $r_i, i = 1,2$:
 for each particle $l = 1,\ldots,N$:
 Receive: $\mathbf{x}^{l}_{r_i}(k-1)$
  Step 1: particle generation: generate particles according to (7).
   Step 2: Mapping:
   Receive: $\mathbf{P}^{l+}_{L_j,r_i}(k-1)$, $\hat{\mathbf{x}}^{l+}_{L_j,r_i}(k-1)$.
   for each $L_j, j = 1,2$:
    • if no measurement is available but the landmark has been visited do:
     1. predict according to (10), (11) and (12)
     2. let $\mathbf{P}^{l+}_{L_j,r_i}(k) = \mathbf{P}^{l-}_{L_j,r_i}(k)$ and $\hat{\mathbf{x}}^{l+}_{L_j,r_i}(k) = \hat{\mathbf{x}}^{l-}_{L_j,r_i}(k)$.
    •endif
    • if measurement $\mathbf{z}_{L_j,r_i}(k)$ is available then:
     • do data association according to (14):
     • if the landmark $L_j$ is visited for the first time do
      1. initialization according to (8) and (9).
     • elseif the landmark has been visited previously do
      1. predict according to (10), (11) and (12).
      2. update according (15) and (16).
     •endif
    •endif
  Step 3: Localization:
   • if measurement $\mathbf{z}_{L_j,r_i}(k)$ is available do:
    1. generate particle weights according to (17) and (18).
    2. perform resampling and generate equally weighted estimates, resulting in:
      $\left\{\frac{1}{N}, \hat{\mathbf{x}}^{l+}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)\right\}^{N}_{l=1}$,
   •endif
   • if no measurement is available do
      $\left\{\frac{1}{N}, \hat{\mathbf{x}}^{l+}_{L_j,r_i}(k), \mathbf{x}^{l}_{r_i}(k)\right\}^{N}_{l=1}$,
   •endif
  Step 4: Coordinate alignment and map merging:
   • if the robot visits another robot do:
    1. compute the rotation matrix, ${}^{G_1}\mathbf{C}_{G_2}$ , and the translation vector, ${}^{G_1}\mathbf{P}_{G_2}$.
    2. transform maps in $G_2$ to $G_1$ according to (19).
    3. transform maps in $G_1$ to $G_2$ according to (20).
    4. fuse maps as explained in Section 4.
   •endif
Return: $\hat{\mathbf{x}}_{L_j,r_i}(k)$ and $\hat{\mathbf{x}}_{r_i}(k)$ through averaging.
```

Algorithm 1. The general multi-robot Fast-SLAM-based algorithm with moving landmarks.

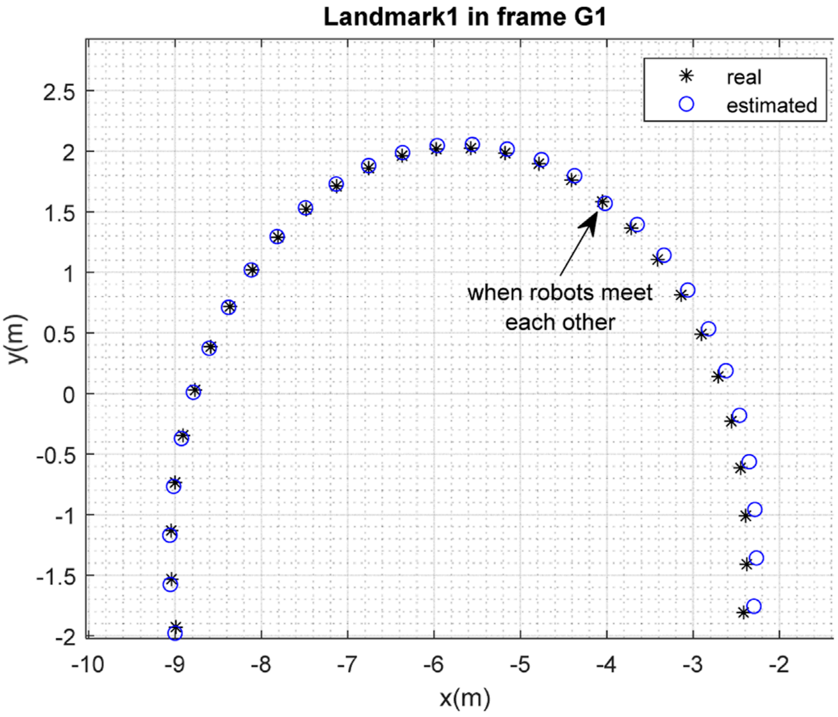

FIGURE 16
Real position of landmark L_1 and position estimated by robot R_2 in the frame G_1 (with different initial conditions).

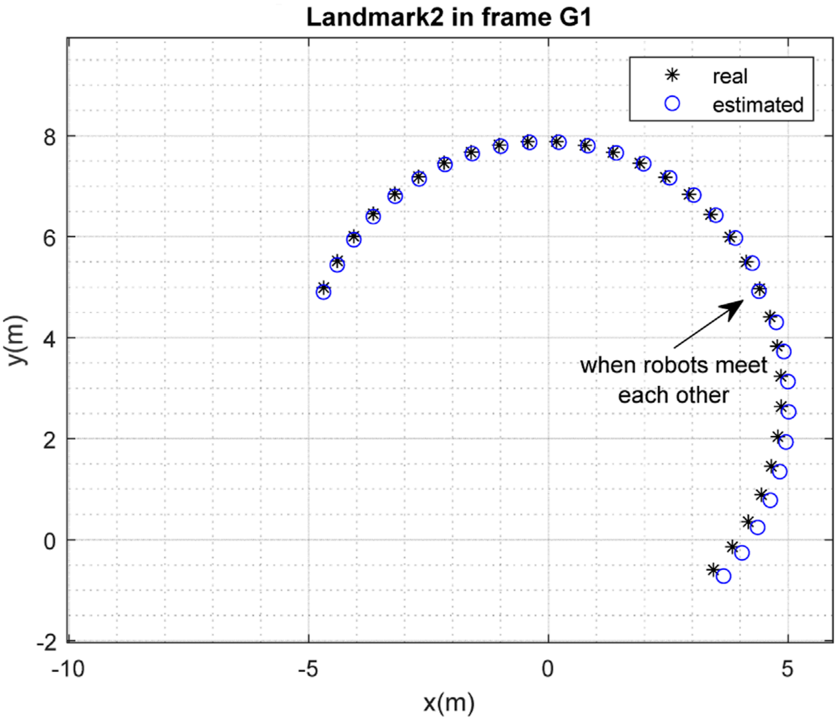

FIGURE 17
Real position of landmark L_2 and position estimated by robot R_2 in the frame G_1 (with different initial conditions).

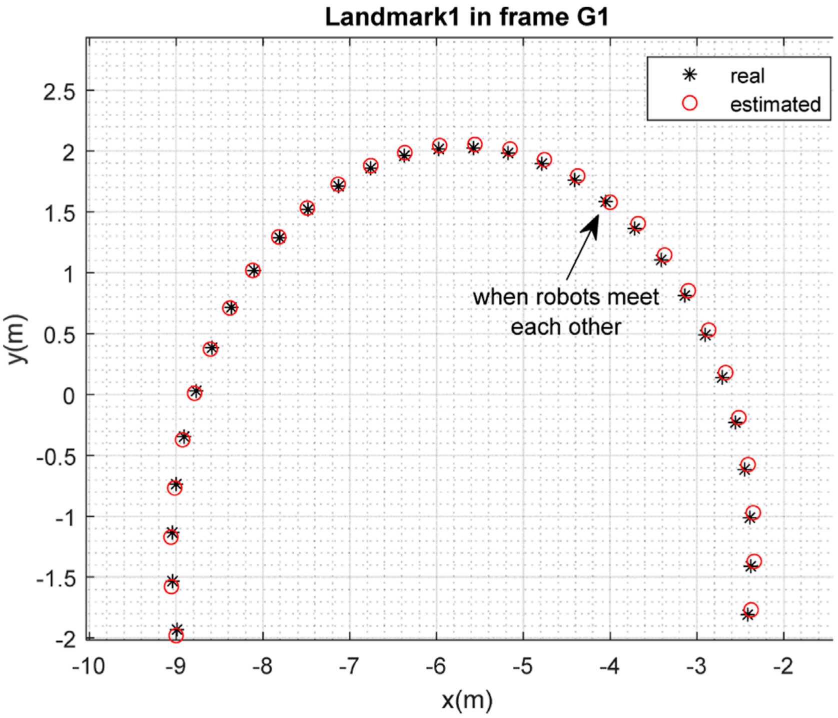

FIGURE 18
Real position of landmark L_1 and position estimated by robot R_1 in the frame G_1 (with different initial conditions).

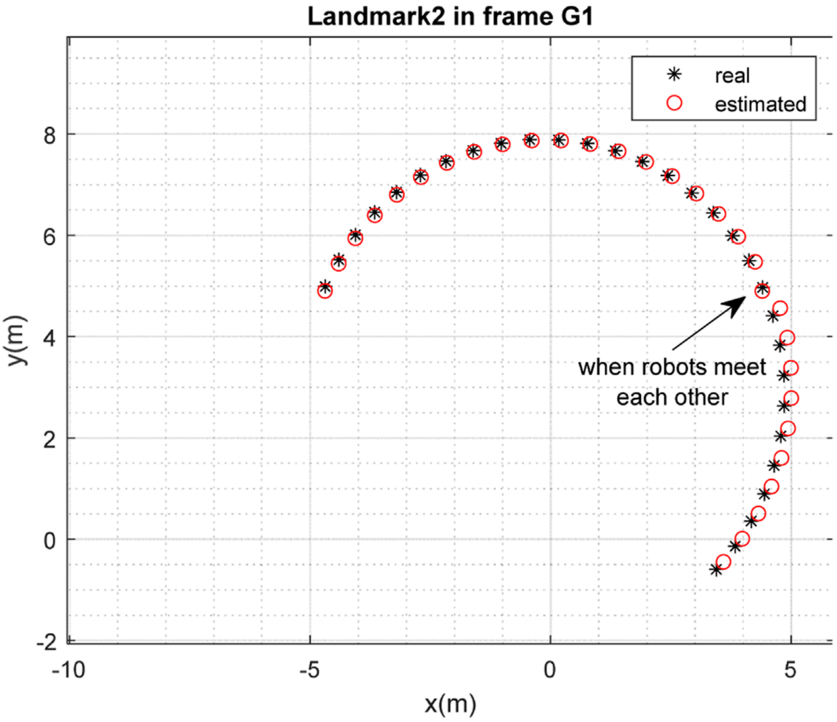

FIGURE 19
Real position of landmark L_2 and position estimated by robot R_1 in the frame G_1 (with different initial conditions).

$$ {}^{G_2}T_{G_1}(estimated) = \begin{bmatrix} 0.8048 & -0.5935 & 0 & 1.0985 \\ 0.5935 & 0.8048 & 0 & -1.7098 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$

Figures 3, 4 depict the results of the localization of robots R_1 and R_2, respectively, in 2D space, expressed in the G_1 coordinate frame, which is selected as the global frame for this paper. Figure 5 depicts the estimated and real paths of robot R_2 in frame G_2 to provide a better illustration of how robot R_2 is performing localization in its own frame G_2.

The real and estimated values of x_{r_i}, y_{r_i}, and φ_{r_i}, expressed in frame G_1 with respect to time, are depicted in Figures 6, 7 for $i = 1$ (robot R_1) and $i = 2$ (robot R_2), respectively. To show the precision of the estimation, Figures 8, 9 depict the corresponding errors: that is, the error between the real and estimated values ($e_{x_{r_i}}$, $e_{y_{r_i}}$, and $e_{\varphi_{r_i}}$) in G_1 for $i = 1, 2$.

The results of the mapping in 2D space are shown in Figures 10, 11 for landmarks L_1 and L_2 obtained from R_1, respectively, expressed in frame G_1; the results obtained from R_2 in frame G_1 are shown in Figures 12, 13. It is worth mentioning again that when the robots meet each other, the maps obtained from R_1 and R_2 are merged. It is clear from the figures that the localization and mapping results are significantly improved after the map-merging process.

To examine the efficiency of the method for the case in which some of the conditions are altered, the initial conditions were selected as follows:

$$ \left[\mathbf{x}_{R_1}(0)\right]_{G_1} = \left[0\ 0\ 1.23\right]^T, \left[\mathbf{x}_{R_2}(0)\right]_{G_1} = \left[1\ 0\ 2.09\right]^T $$
$$ \left[\mathbf{x}_{L_1}(0)\right]_{G_1} = \left[-2.5\ -2.2\ 1.22\right]^T, \left[\mathbf{x}_{L_2}(0)\right]_{G_1} = \left[3\ -1\ 2.09\right]^T $$

The results of localization of the robots in frame G_1 are depicted in Figures 14, 15, and the results of the mapping are depicted in Figures 16–19. It is clear from the figures that the algorithm is successful in simultaneous localization and mapping under changing initial conditions.

6 Conclusion

In this paper, an efficient algorithm has been presented for a multi-robot SLAM problem with unknown initial correspondence in a dynamic environment, using a modified Fast-SLAM method. In our scenario, each robot independently searches the environment, observes the moving landmarks in the environment using a lidar sensor, and implements the SLAM algorithm. In order to distinguish the moving landmarks, kinematic models are considered for the landmarks, which led to a modification of the normal Fast-SLAM method in the form of the addition of a prediction phase to the method; additionally, data association was performed according to the predicted measurements obtained from this prediction step. Although the kinematic models of the landmarks are known within each robot's coordinate system, after the first meeting of the robots, an initialization is embedded in the algorithm to obtain the current positions of the landmarks, as they are unknown to the robots.

Since the initial correspondence of the robots is unknown (or, in other words, each robot performs the mapping from the perspective of its own coordinate frame), a map-merging procedure was embedded in the proposed algorithm to fuse the independent maps of the robots when the robots meet each other. This map-merging is only possible when the relative transformation matrix of the robots' inertial frames is computed, which occurs when the robots meet each other. For this purpose, a geographical approach to compute this transformation matrix was embedded in the proposed algorithm.

The performance of the proposed method was evaluated through simulations in MATLAB. It can be concluded from the simulation results that although each robot was able to solve the SLAM problem with an acceptable level of performance, the accuracy of SLAM was significantly improved when the robots met each other and map-merging was performed.

Although the proposed method showed very good performance in simulations in the MATLAB environment, it will perhaps encounter some difficulties in a real environment, such as model mismatches in relation to both the landmarks and the robots due to noise, and the occurrence of drift, for example, in the inertial sensor measurements. Additionally, designing a rendezvous method for the robots to force them to visit each other can represent another issue in a real environment.

Generalization of the proposed method for the general case of a multi-robot and multi-landmark system, with more than two robots and two moving landmarks, is proposed for future work. Embedding some approach to object detection, instead of using the kinematic models of the landmarks, would also add significant value to this research.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding authors.

Author contributions

HK: Investigation, Methodology, Software, Writing–original draft. NS-N: Conceptualization, Methodology, Project administration, Supervision, Validation, Writing–original draft. AM: Conceptualization, Funding acquisition, Project administration, Resources, Supervision, Writing–review and editing.

Funding

The author(s) declare financial support was received for the research, authorship, and/or publication of this article. This work was supported in part by the Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/R02572X/1.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The author(s) declared that they were an editorial board member of Frontiers, at the time of submission. This had no impact on the peer review process and the final decision.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Almadhoun, R., Taha, T., Seneviratne, L., and Zweiri, Y. (2019). A survey on multi-robot coverage path planning for model reconstruction and mapping. *SN Appl. Sci.* 1, 847–924. doi:10.1007/s42452-019-0872-y

Badalkhani, S., and Havangi, R. (2021). Effects of moving landmark's speed on multi-robot simultaneous localization and mapping in dynamic environments. *Iran. J. Electr. Electron. Eng.* 17 (1), 1740. doi:10.22068/IJEEE.17.1.1740

Burrell, T., West, C., Monk, S. D., Montezeri, A., and Taylor, C. J. (2018). "Towards a cooperative robotic system for autonomous pipe cutting in nuclear decommissioning," in UKACC 12th International Conference on Control (CONTROL), Sheffield, UK, 5-7 Sept. 2018 (IEEE), 283–288.

Chen, B., Peng, G., He, D., Zhou, C., and Hu, B. (2021). "Visual SLAM based on dynamic object detection," in 2021 33rd Chinese Control and Decision Conference (CCDC), Germany, 22-24 May 2021 (IEEE), 5966–5971.

Debeunne, C., and Vivet, D. (2020). A review of visual-LiDAR fusion based simultaneous localization and mapping. *Sensors* 20 (7), 2068. doi:10.3390/s20072068

De Hoog, J., Cameron, S., and Visser, A. (2010). *Selection of rendezvous points for Multi– robot exploration in dynamic environments.*

Godsill, S. (2019). "Particle filtering: the first 25 years and beyond," in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), China, 12-17 May 2019 (IEEE), 7760–7764.

Guan, H., Qian, C., Wu, T., Hu, X., Duan, F., and Ye, X. (2023). A dynamic scene vision SLAM method incorporating object detection and object characterization. *Sustainability* 15 (4), 3048. doi:10.3390/su15043048

Kretzschmar, H., Grisetti, G., and Stachniss, C. (2010). Lifelong map learning for graph-based SLAM in static environments. *KI-Künstliche Intell.* 24, 199–206. doi:10.1007/s13218-010-0034-2

Li, A., Wang, J., Xu, M., and Chen, Z. (2021). DP-SLAM: a visual SLAM with moving probability towards dynamic environments. *Inf. Sci.* 556, 128–142. doi:10.1016/j.ins.2020.12.019

Liu, H., Liu, G., Tian, G., Xin, S., and Ji, Z. (2019). "Visual SLAM based on dynamic object removal," in IEEE International Conference on Robotics and Biomimetics (ROBIO), New York, 5-9 Dec. 2022 (IEEE), 596–601.

Lluvia, I., Lazkano, E., and Ansuategi, A. (2021). Active mapping and robot exploration: a survey. *Sensors* 21 (7), 2445. doi:10.3390/s21072445

Meghjani, M., and Dudek, G. (2011). "Combining multi-robot exploration and rendezvous," in *Canadian conference on computer and robot vision* (USA: IEEE), 80–85.

Meghjani, M., and Dudek, G. (2012). "Multi-robot exploration and rendezvous on graphs," in IEEE/RSJ International Conference on Intelligent Robots and Systems, USA, 23-27 Oct. 2022 (IEEE), 5270–5276.

Montazeri, A., Can, A., and Imran, I. H. (2021). *Unmanned aerial systems: autonomy, cognition, and control. Unmanned aerial systems.* USA: Academic Press, 47–80.

Montemerlo, M., Thrun, S., Koller, D., and Wegbreit, B. (2002). FastSLAM: a factored solution to the simultaneous localization and mapping problem. *Aaai/iaai* 593598. doi:10.5555/777092.777184

Pancham, A., Withey, D., and Bright, G. (2020). Amore: CNN-based moving object detection and removal towards SLAM in dynamic environments. *South Afr. J. Industrial Eng.* 31 (4), 46–58. doi:10.7166/31-4-2180

Romero, V., and Costa, O. (2010). "Map merging strategies for multi-robot fastSLAM: a comparative survey," in *Latin American robotics symposium and intelligent robotics meeting* (America: IEEE), 61–66.

Roy, N., and Dudek, G. (2001). Collaborative robot exploration and rendezvous: algorithms, performance bounds and observations. *Aut. Robots* 11, 117–136. doi:10.1023/a:1011219024159

Sadeghzadeh-Nokhodberiz, N., Can, A., Stolkin, R., and Montazeri, A. (2021). Dynamics-based modified fast simultaneous localization and mapping for unmanned aerial vehicles with joint inertial sensor bias and drift estimation. *IEEE Access* 9, 120247–120260. doi:10.1109/access.2021.3106864

Saeedi, S., Trentini, M., Seto, M., and Li, H. (2016). Multiple-robot simultaneous localization and mapping: a review. *J. Field Robotics* 33 (1), 3–46. doi:10.1002/rob.21620

Saputra, M. R. U., Markham, A., and Trigoni, N. (2018). Visual SLAM and structure from motion in dynamic environments: a survey. *ACM Comput. Surv. (CSUR)* 51 (2), 1–36. doi:10.1145/3177853

Stachniss, C., Leonard, J. J., and Thrun, S. (2016). *Simultaneous localization and mapping*. Germany: Springer Handbook of Robotics, 1153–1176.

Theodorou, C., Velisavljevic, V., and Dyo, V. (2022). Visual SLAM for dynamic environments based on object detection and optical flow for dynamic object removal. *Sensors* 22 (19), 7553. doi:10.3390/s22197553

Vidal, F. S., Barcelos, A. d. O. P., and Rosa, P. F. F. (2015). "SLAM solution based on particle filter with outliers filtering in dynamic environments," in IEEE 24th International Symposium on Industrial Electronics (ISIE, USA, 3-5 June 2015 (IEEE), 644–649.

Zaman, S., Slany, W., and Steinbauer, G. (2011). "ROS-based mapping, localization and autonomous navigation using a Pioneer 3-DX robot and their relevant issues," in *Saudi international electronics, communications and photonics conference* (Germany: SIECPC), 1–5.

Zhou, X. S., and Roumeliotis, S. I. (2006). *Multi-robot SLAM with unknown initial correspondence: the robot rendezvous case* in IEEE/RSJ international conference on intelligent robots and systems, USA, October 23–27, 2022 (IEEE), 1785–1792.