Multi-Robot SLAM using M-Space Feature Representation

Daniele Benedettelli, Andrea Garulli and Antonio Giannitrapani

***Abstract*— This paper presents a SLAM algorithm for a team of mobile robots exploring an indoor environment, described by adopting the M-Space representation of linear features. Each robot solves the SLAM problem independently. When the robots meet, the local maps are fused together using robot-to-robot relative range and bearing measurements.**

A map fusion technique, tailored to the specific feature representation adopted, is proposed. Moreover, the uncertainty affecting the resulting merged map is explicitly derived from the single-robot SLAM maps and the robot-to-robot measurement accuracy. Simulation experiments are presented showing a team composed of two robots performing SLAM in a real-world scenario.

I. INTRODUCTION

To achieve real autonomy, a mobile robot must be able to localize itself in the environment it is exploring. When a map is not available, the robot should build it, while at the same time, localizing itself within it. This problem, known as *Simultaneous Localization and Map building* (SLAM), can be cast as a state estimation problem, and has been extensively studied over the last two decades, for different environment descriptions and estimation techniques (see the survey [1] for a thorough review). In indoor applications, lines and segments are often adopted to map the environment, thanks to the plenty of linear features like walls and furniture (e.g., see [2], [3]). Unfortunately, commonly used line parameterizations suffer from the so called "lever-arm effect", meaning that the parameter uncertainty increases with the distance from the origin of the reference frame. In order to overcome such a drawback, special line representations can be adopted, like the SP-Model [4] or the recently proposed M-Space representation [5].

A crucial issue for a SLAM algorithm is the ability to close loops, i.e. to recognize places that have been already visited. Since the larger the loop, the harder the closure, the difficulty of the SLAM problem increases with the size of the environment. Teams of cooperating robots could be employed to improve the loop closure ability, as well as to perform exploration and mapping tasks more quickly and robustly, and to increase the quality of the map. For this reason, multi-robot SLAM algorithms have been proposed in recent years, adopting different estimation techniques, like Extended Kalman Filters (EKF) [6], [7], Information Filters [8], Particle Filters [9] or Set-Membership estimators [10]. A key issue for the effectiveness of multi-robot SLAM algorithm is the ability to merge in a common reference frame maps built by different robots, in different frames. Such a task is particularly difficult if no information is available on the initial robot poses. This problem is addressed in [11], where a map fusion algorithm has been proposed, based on relative range and bearing measurements among robots, in an environment described by point-wise features.

In this paper, a new multi-robot SLAM algorithm is proposed, which exploits the M-Space representation to describe linear features in the map and adopts a map fusion scheme inspired by the approach proposed in [11]. Each robot runs an EKF for independently localizing and building local maps. When two robots meet, the local maps are merged by using robot-to-robot relative range and bearing measurements, and the uncertainty of the resulting map is properly updated. The main contributions of the paper are: i) a map fusion algorithm for environments described in terms of lines and segments; ii) the explicit computation of the covariance matrix of the resulting global map when the M-Space representation is adopted. The map fusion procedure is described for a team composed of two robots, but can be applied to larger teams by repeating the procedure for each pair of robots.

The paper is organized as follows. In Section II, the M-Space representation of linear features is briefly recalled, while Section III presents an overview of the single-robot SLAM algorithm, based on the EKF. In Section IV the map fusion technique, as well as the update of the resulting map uncertainty, are presented. In Section V the results of simulations and experimental tests in a real-world scenario are reported. Finally, Section VI contains some conclusions and lines of future research.

II. M-SPACE FEATURE REPRESENTATION

In this paper, a line segment in the plane is described by its endpoints coordinates, $\mathbf{x}_f = [x_A\ y_A\ x_B\ y_B]^T$ or, alternatively, by geometric parameters as $\mathbf{x}_p = [\alpha,\ \rho,\ d_A,\ d_B]^T$, where $\alpha \in (-\pi, \pi]$ is the angle between the x-axis and the normal to the line from the origin, $\rho \geq 0$ is the distance from the origin to the line, d_A and d_B are the distances (with proper sign) of the segment endpoints A and B to the point of incidence of the normal to the line (see Figure 1).

The *measurement subspace*, or M-Space [5], is a feature representation that attaches a local frame to each feature element, allowing for a generic treatment of many types of features. The parameters $\mathbf{x}_p$ represent the M-Space coordinates of a feature and are expressed in the corresponding reference frame. Notice that each $\mathbf{x}_p$ lives in a different reference frame.

D. Benedettelli, A. Garulli and A. Giannitrapani are with the Department of Information Engineering, University of Siena, via Roma, 56 - 53100 Siena, Italy. Email: `danielebenedettelli@gmail.com`, `{garulli,giannitrapani}@dii.unisi.it`

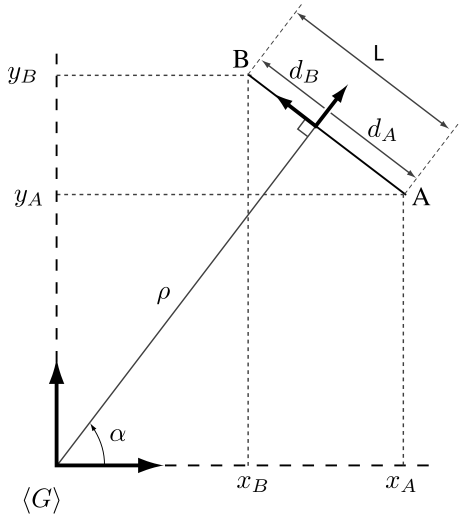


Fig. 1. Line parameters.

Let $\delta\mathbf{x}_p$ denote the change in the M-Space corresponding to a small change in the feature coordinates $\delta\mathbf{x}_f$. Small variations $\delta\mathbf{x}_f$ in the feature space are related to small variations of the corresponding M-Space coordinates $\delta\mathbf{x}_p$ by a projection matrix $\widetilde{B}_f(\mathbf{x}_f)$. The fundamental relationships between $\delta\mathbf{x}_p$ and $\delta\mathbf{x}_f$ are thus

$$\begin{array}{rcl} \delta\mathbf{x}_f &=& \widetilde{B}(\mathbf{x}_f)\delta\mathbf{x}_p \\ \delta\mathbf{x}_p &=& B(\mathbf{x}_f)\delta\mathbf{x}_f \\ I_q &=& B(\mathbf{x}_f)\widetilde{B}(\mathbf{x}_f) \end{array} \tag{1}$$

where I_q denotes the identity matrix of order q, and the projection matrices $\widetilde{B}(\mathbf{x}_f)$ and $B(\mathbf{x}_f)$ depend on the value of $\mathbf{x}_f$ (see [5], [12] for details). When facing the map estimation problem, the relationships (1) are used to project the correction of the estimates from the M-Space ($\delta\mathbf{x}_p$) to the feature space ($\delta\mathbf{x}_f$).

A benefit of the M-Space representation is that the features are parametrized to fully specify their location and extent, but can be initialized in a subspace corresponding to the partial information provided by the robot sensors. In fact, a common problem in feature-based SLAM is that most often the robot cannot initialize a feature in all its dimensions after the first observation. Since the entire line feature is not detected at once, the dimensions q of the M-Space coordinates $\mathbf{x}_p$ can grow from a minimum of 2, when the robot first measures the line position and orientation, up to 4 dimensions, when it has observed both segment endpoints. Then, the projection matrices $B(\mathbf{x}_f)$ and $\widetilde{B}(\mathbf{x}_f)$ have q rows and q columns, respectively. In the following treatment, for ease of notation we will always assume $q = 4$ (the case $q < 4$ requiring straightforward modifications). Another nice property of the M-Space representation is that, being each feature described in a local reference frame, the drawbacks related to the lever-arm effect are mitigated.

III. SINGLE-ROBOT SLAM PROBLEM ALGORITHM

In the following, the notation $\hat{\mathbf{x}}$ denotes the *estimate* of the true quantity $\mathbf{x}$, and $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ is the related estimation error. Let us consider an autonomous robot navigating in a 2D environment, and let $\mathbf{x}_R = [x_R\ y_R\ \theta_R]^T$ be its pose, where $[x_R\ y_R]^T$ is the position and θ_R is the orientation with respect to a global reference frame. Assuming that the robot is navigating indoor, the surrounding environment can be described by using line segments extracted from walls, doors, or furniture. The discrete-time robot motion model based on linear and angular velocity commands $\mathbf{u}(k) = [v(k)\ \ \omega(k)]^T$ can be described as

$$\mathbf{x}_R(k+1) = f(\mathbf{x}_R(k), \mathbf{u}(k), \varepsilon_{\mathbf{u}}(k)) \tag{2}$$

where $\varepsilon_{\mathbf{u}}$ is a white noise affecting the velocities, so that $E[\varepsilon_{\mathbf{u}}(k)] = \mathbf{0}$ and $E[\varepsilon_{\mathbf{u}}(k)\varepsilon_{\mathbf{u}}^T(k)] = Q(k)$.

Let $\mathbf{m}$ denote a measurement of a line segment,

$$\mathbf{m}(k) = [\alpha(k)\ \rho(k)\ d_A(k)\ d_B(k)]^T + \varepsilon_{\mathbf{m}}(k) \tag{3}$$

affected by white noise $\varepsilon_{\mathbf{m}}(k)$, so that $E[\varepsilon_{\mathbf{m}}(k)] = \mathbf{0}$ and $E[\varepsilon_{\mathbf{m}}(k)\ \varepsilon_{\mathbf{m}}^T(k)] = R_{\mathbf{m}}(k)$. It is assumed that the robot is equipped with a laser range finder sensor, and $\mathbf{m}(k)$ and $R_{\mathbf{m}}(k)$ are extracted from the raw readings of the sensor. Notice that $d_A(k)$ and $d_B(k)$ are present in the measurement vector $\mathbf{m}(k)$ only if the endpoints of the identified segment are detected (e.g., due to a corner).

The vector to be estimated is

$$\mathbf{\Xi}(k) = [\mathbf{x}_R^T(k)\ \mathbf{x}_{f_1}^T(k)\ \ \ldots\ \ \mathbf{x}_{f_n}^T(k)]^T \in \mathbb{R}^{3+4n} \tag{4}$$

where $\mathbf{x}_{f_i}$ is the i-th feature ($i = 1 \ldots n$) described by its endpoints in the global frame, and n is the number of features. Notice that features are detected incrementally during navigation and hence n grows with time. Since static features are considered, the time evolution of the vector $\mathbf{\Xi}(k)$ only affects the robot pose $\mathbf{x}_R$ via (2), leaving the features $\mathbf{x}_f$ unchanged. Now, the SLAM problem can be stated as follows.

Let $\hat{\mathbf{\Xi}}(0)$ be an estimate of the initial robot pose and feature coordinates. Given the dynamic model (2) and the measurement model (3), find an estimate $\hat{\mathbf{\Xi}}(k)$ of the robot pose and feature coordinates $\mathbf{\Xi}(k)$ for each time $k \in \mathbb{N}_+$.

When adopting the M-Space representation, the SLAM problem can be tackled as a state estimation problem in which the state vector includes the robot pose $\mathbf{x}_R$ in the global frame and the M-Space coordinates $\mathbf{x}_p$ of each feature. Then, the estimate of the vector $\mathbf{\Xi}(k)$ in (4), and the corresponding uncertainty, are obtained by exploiting the relationships (1) [5].

The state of the filter is defined as

$$\hat{\mathbf{x}}_s \triangleq \left[\hat{\mathbf{x}}_R^T\ \hat{\mathbf{x}}_{p_1}^T\ \ \ldots\ \ \hat{\mathbf{x}}_{p_n}^T\right]^T \tag{5}$$

where $\mathbf{x}_{p_i}$ $i = 1 \ldots n$ are the M-Space coordinates of the i-th feature. The covariance matrix of the state estimation error is $P_{\mathbf{x}_s} = E\left[\tilde{\mathbf{x}}_s\ \tilde{\mathbf{x}}_s^T\right]$. An EKF is used to estimate the state $\hat{\mathbf{x}}_s$ and the covariance matrix $P_{\mathbf{x}_s}$: at each time k, the state estimates $\hat{\mathbf{x}}_s(k|k)$ and the corresponding covariance $P_{\mathbf{x}_s}(k|k)$ (according to the usual EKF notation) are available. Then, the state correction based on the measurement $\mathbf{m}(k)$

$$\delta\hat{\mathbf{x}}_s(k) = \hat{\mathbf{x}}_s(k|k) - \hat{\mathbf{x}}_s(k|k-1)$$

is used to update the estimate of the vector $\hat{\Xi}(k)$. By exploiting (1), (4) and (5) one has $\hat{\Xi}(k) = \hat{\Xi}(k-1) + \mathbf{\Pi}\delta\mathbf{x}_s(k)$, where $\mathbf{\Pi} = \text{blkdiag}\left(I_3,\ \widetilde{B}(\hat{\mathbf{x}}_{f_1}), \ldots, \widetilde{B}(\hat{\mathbf{x}}_{f_n})\right)$ is a block diagonal matrix built using the $\widetilde{B}(\hat{\mathbf{x}}_{f_i})$ matrices evaluated at the current feature estimates $\hat{\mathbf{x}}_{f_i}(k-1)$. Summarizing, the single-robot EKF SLAM algorithm produces an estimate $\hat{\Xi}(k)$ of the robot pose and the segment endpoints in the global frame, as well as the covariance matrix P_s, that expresses robot uncertainty in the global frame and feature uncertainties in the M-Space.

IV. Multi-robot SLAM algorithm

In order to fuse maps created by different robots, whose initial poses are unknown, the transformation between their reference frames needs to be determined. This can be done by using robot-to-robot mutual measurements [11].

First, the two maps are aligned by employing the transformation (rotation and translation) determined by the robot-to-robot measurements (Section IV-A). Then, the estimation error covariances of the maps are updated according to the transformation employed to align the maps (Section IV-B). Finally, possible overlapping between the two maps is analyzed by searching for feature matches. If duplicate features are identified, this information is used to impose constraints that improve the accuracy of the resulting map (Section IV-C).

A. Map alignment

Suppose there are two robots R_1 and R_2 that are mapping independently the area they are exploring, with respect to their initial global frame, $\langle G_1 \rangle$ and $\langle G_2 \rangle$ respectively. In the following, the notation ${}^f X_r$ is adopted, where the left superscript f indicates that the quantity X is expressed in the reference frame $\langle G_f \rangle$, and the right subscript r indicates which robot the quantity X is referred to.

Initially, each robot performs single-robot SLAM, i.e. the SLAM algorithms of the robots R_1 and R_2 estimate the vectors ${}^1\Xi_1 \in \mathbb{R}^{m_1}$ and ${}^2\Xi_2 \in \mathbb{R}^{m_2}$ (see equation (4)), where $m_1 = 3 + 4n_1$ and $m_2 = 3 + 4n_2$, and n_1 and n_2 are the number of features in the map of robot R_1 and R_2, respectively. Let $\mathbf{X} = [{}^1\Xi_1^T\ {}^2\Xi_2^T]^T \in \mathbb{R}^{(m_1+m_2)}$, be the augmented vector, obtained by stacking the two vectors of the single-robot SLAM problems. Denote by $\widetilde{\mathbf{X}}$ the augmented vector estimation error.

The map alignment problem consists in finding the roto-translation between the reference frames $\langle G_1 \rangle$ and $\langle G_2 \rangle$, in order to express the map estimated by a robot in the frame of the other one. For example, this amounts to compute an estimate of the vector ${}^1\Xi_2$, i.e. the map of robot R_2 in the frame $\langle G_1 \rangle$. This problem is tackled by processing mutual *relative range and bearing measurements*, when the two robots are within sensing distance of each other. The transformation between the two maps is computed by exploiting one pair of robot-to-robot measurements.

We assume that each robot can use its range sensor to measure the distance η and bearing ϕ towards the other robot

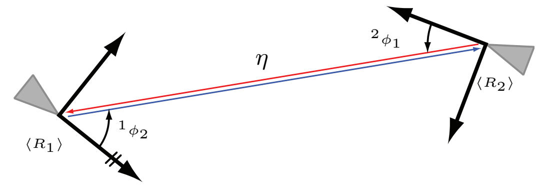


Fig. 2. Robot-to-robot measurements.

(see Figure 2). The measurement of the relative position of robot j with respect to robot i is described by:

$$ {}^i\bar{\mathbf{z}}_j = \begin{bmatrix} {}^i\bar{\eta} \\ {}^i\bar{\phi}_j \end{bmatrix} = \begin{bmatrix} \eta \\ {}^i\phi_j \end{bmatrix} + \begin{bmatrix} \varepsilon_{{}^i\eta} \\ \varepsilon_{{}^i\phi_j} \end{bmatrix} \quad i,j = 1,\ 2 $$

where η is the distance between the two robots, ${}^i\phi_j$ is the direction in which robot R_i sees robot R_j, and $\varepsilon_{{}^i\eta}$, $\varepsilon_{{}^i\phi_j}$ are white zero-mean measurement noise. Since the two distance measurements ${}^1\bar{\eta}$ and ${}^2\bar{\eta}$ are independent, a more accurate estimate of the distance between the two robots can be computed as the weighted average $\bar{\eta}$ of the two measurements. We form the combined measurement vector as

$$ \bar{\mathbf{z}} = \begin{bmatrix} \bar{\eta} \\ {}^1\bar{\phi}_2 \\ {}^2\bar{\phi}_1 \end{bmatrix} = \begin{bmatrix} \eta \\ {}^1\phi_2 \\ {}^2\phi_1 \end{bmatrix} + \begin{bmatrix} \varepsilon_\eta \\ \varepsilon_{{}^1\phi_2} \\ \varepsilon_{{}^2\phi_1} \end{bmatrix} = \mathbf{z} + \varepsilon_{\mathbf{z}} $$

where $\mathbf{z}$ denotes the vector of the real distance and relative bearings, and $\varepsilon_{\mathbf{z}}$ are white measurement noises with covariance

$$ R_{\mathbf{z}} = E[\varepsilon_{\mathbf{z}}\ \varepsilon_{\mathbf{z}}{}^T] = \text{diag}(\sigma_\eta^2, \sigma_{{}^1\phi_2}^2, \sigma_{{}^2\phi_1}^2). \tag{6} $$

From geometrical considerations, the distance and angles η, ${}^1\phi_2$, ${}^2\phi_1$ allow one to compute the exact transformation $\mathbf{t}$ between the reference frame of the two robots (see [11] for details):

$$ {}^1\Xi_2 = \mathbf{t}(\mathbf{X},\ \mathbf{z}). \tag{7} $$

Since the arguments of the function $\mathbf{t}$ are not known exactly, the nominal estimate ${}^1\hat{\Xi}_2$ is determined by replacing them with the corresponding estimates, i.e. ${}^1\hat{\Xi}_2 = \mathbf{t}(\hat{\mathbf{X}},\ \bar{\mathbf{z}})$.

B. Updating map uncertainty

When the robots meet, each of them has an estimate ${}^1\hat{\mathbf{x}}_{s_1}$, ${}^2\hat{\mathbf{x}}_{s_2}$ (in M-Space coordinates) of the region of the environment explored, as well as the covariance of the corresponding estimation errors $P_{\mathbf{x}_{s_1}}$ and $P_{\mathbf{x}_{s_2}}$. Let us stack together the two M-Space state vectors as $\mathbf{X}_s = [{}^1\mathbf{x}_{s_1}^T\ {}^2\mathbf{x}_{s_2}^T]^T$ and let

$$ P_s = E[\widetilde{\mathbf{X}}_s \widetilde{\mathbf{X}}_s^T] = \text{blkdiag}(P_{\mathbf{x}_{s_1}}, P_{\mathbf{x}_{s_2}}) \tag{8} $$

be the covariance matrix of the estimation error. The relationships between the estimation error in the M-Space $\widetilde{\mathbf{X}}_s$ and the estimation error in the feature space $\widetilde{\mathbf{X}}$ can be derived by exploiting the projection equations (1). In fact, let ${}^1\mathbf{x}_{f_i,1}$ be a feature in the map of robot R_1 expressed in the frame $\langle G_1 \rangle$, and let ${}^2\mathbf{x}_{f_j,2}$ be a feature in the map of R_2 expressed in $\langle G_2 \rangle$. The M-Space parameters errors can be projected in feature space, according to (1), as

$$ {}^1\tilde{\mathbf{x}}_{f_i,1} = \widetilde{B}({}^1\hat{\mathbf{x}}_{f_i,1})\ {}^1\tilde{\mathbf{x}}_{p_i,1}, \tag{9} $$

$$ {}^2\tilde{\mathbf{x}}_{f_j,2} = \widetilde{B}({}^2\hat{\mathbf{x}}_{f_j,2})\ {}^2\tilde{\mathbf{x}}_{p_j,2}. \tag{10} $$

Stacking all the features together, equations (9)-(10) provide the sought relationship $\widetilde{\mathbf{X}} = \widetilde{\mathbf{B}}\widetilde{\mathbf{X}}_\mathbf{s}$, where $\widetilde{\mathbf{B}}$ is a block diagonal matrix, defined as

$$\begin{aligned}
\widetilde{\mathbf{B}} &= \text{blkdg}({}^1\widetilde{\mathbf{B}}_1, {}^2\widetilde{\mathbf{B}}_2),\\
{}^1\widetilde{\mathbf{B}}_1 &= \text{blkdg}(I_3, \widetilde{B}({}^1\hat{\mathbf{x}}_{f_1,1}), \ldots, \widetilde{B}({}^1\hat{\mathbf{x}}_{f_{n_1},1})),\\
{}^2\widetilde{\mathbf{B}}_2 &= \text{blkdg}(I_3, \widetilde{B}({}^2\hat{\mathbf{x}}_{f_1,2}), \ldots, \widetilde{B}({}^2\hat{\mathbf{x}}_{f_{n_2},2})).
\end{aligned}$$

As a result, if $P = E[\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T]$ denotes the covariance of the estimation error $\tilde{\mathbf{X}}$, then

$$P = \widetilde{\mathbf{B}} P_s \widetilde{\mathbf{B}}^T, \tag{11}$$

with P_s given by (8). Notice that P still refers to the estimation errors of the feature coordinates of the two maps in two different reference frames. The next step is to compute the covariance matrix of the estimation error of the aligned maps. To this purpose, let us define the new augmented state vector in the reference frame $\langle G_1 \rangle$ as

$$\mathbf{X}^a = \begin{bmatrix} {}^1\mathbf{\Xi}_1 \\ {}^1\mathbf{\Xi}_2 \end{bmatrix} = \begin{bmatrix} {}^1\mathbf{\Xi}_1 \\ \mathbf{t}(\mathbf{X},\ \mathbf{z}) \end{bmatrix}.$$

In order to express the estimation error $\widetilde{\mathbf{X}}^a$ as a function of the estimation error $\widetilde{\mathbf{X}}$, it is necessary to linearize equation (7). Then one gets

$$\widetilde{\mathbf{X}}^a = \begin{bmatrix} \mathbf{I}_{m_1} & \mathbf{0}_{m_1 \times m_2} \\ \mathbf{T}_1 & \mathbf{T}_2 \end{bmatrix} \tilde{\mathbf{X}} + \begin{bmatrix} \mathbf{0}_{m_1 \times 3} \\ \mathbf{\Gamma}_2 \end{bmatrix} \varepsilon_\mathbf{z}, \tag{12}$$

where the matrices $\mathbf{T}_1$, $\mathbf{T}_2$ and $\mathbf{\Gamma}_2$ are the Jacobians of the transformation $\mathbf{t}$ with respect to ${}^1\mathbf{\Xi}_1$, ${}^2\mathbf{\Xi}_2$ and $\mathbf{z}$, respectively, computed at the estimates ${}^1\hat{\mathbf{\Xi}}_1$, ${}^2\hat{\mathbf{\Xi}}_2$ and at the measurement $\bar{\mathbf{z}}$ (see [12] for the analytical expression of matrices $\mathbf{T}_1$, $\mathbf{T}_2$ and $\mathbf{\Gamma}_2$). From (11)-(12), the covariance of the aligned augmented vector $P^a = E[\widetilde{\mathbf{X}}^a(\widetilde{\mathbf{X}}^a)^T]$ is given by

$$\begin{aligned}
P^a = &\begin{bmatrix} \mathbf{I}_{m_1} & \mathbf{0}_{m_1 \times m_2} \\ \mathbf{T}_1 & \mathbf{T}_2 \end{bmatrix} \widetilde{\mathbf{B}} P_s \widetilde{\mathbf{B}}^T \begin{bmatrix} \mathbf{I}_{m_1} & \mathbf{0}_{m_1 \times m_2} \\ \mathbf{T}_1 & \mathbf{T}_2 \end{bmatrix}^T \\
&+ \begin{bmatrix} \mathbf{0}_{m_1 \times 3} \\ \mathbf{\Gamma}_2 \end{bmatrix} R \begin{bmatrix} \mathbf{0}_{m_1 \times 3} \\ \mathbf{\Gamma}_2 \end{bmatrix}^T
\end{aligned} \tag{13}$$

In order for the robots to start again with the single-robot SLAM after the map fusion, the covariance of the aligned map in the M-Space is needed. Let us define the M-Space aligned vector $\mathbf{X}_s^a = [{}^1\mathbf{x}_{s_1}^T \ {}^1\mathbf{x}_{s_2}^T]^T$. Similarly to what has been done before, by exploiting (1) it is possible to relate the estimation error of the aligned vector in the M-Space and the corresponding one in the feature space as $\widetilde{\mathbf{X}}_s^a = \mathbf{B}\widetilde{\mathbf{X}}^a$, where

$$\begin{aligned}
\mathbf{B} &= \text{blkdg}({}^1\mathbf{B}_1, {}^1\mathbf{B}_2),\\
{}^1\mathbf{B}_1 &= \text{blkdg}(I_3, B({}^1\hat{\mathbf{x}}_{f_1,1}), \ldots, B({}^1\hat{\mathbf{x}}_{f_{n_1},1})),\\
{}^1\mathbf{B}_2 &= \text{blkdg}(I_3, B({}^1\hat{\mathbf{x}}_{f_1,2}), \ldots, B({}^1\hat{\mathbf{x}}_{f_{n_2},2})).
\end{aligned}$$

Finally, the covariance $P_s^a = E[\widetilde{\mathbf{X}}_s^a(\widetilde{\mathbf{X}}^a)_s^T]$ can be computed as $P_s^a = \mathbf{B}P^a\mathbf{B}^T$, where P^a is given by (13).

C. Matching and eliminating duplicate features

It is very likely that the areas covered by the two robots before rendezvous have common regions. This means that a number of features may appear as duplicates in the new vector $\mathbf{X}^a$ resulting from the map alignment. By employing this information, one can improve the accuracy of the final map.

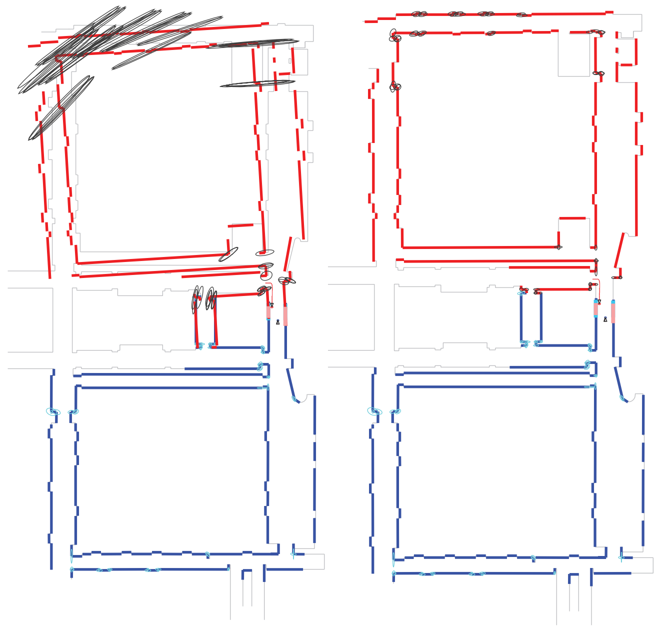
A popular method for searching for matching landmarks is the Nearest Neighbor (NN) algorithm, that simply pairs the two features with the closest Mahalanobis distance [13]. In this work, a modified NN algorithm, similar to that used in [3], has been adopted. Three validation gates on the distance, orientation and overlapping of the features are employed to determine beforehand the candidate pairings. Then, the NN algorithm is run among all the feature associations that have passed the validation gates.

Now, suppose two features $\hat{\mathbf{x}}_{f_i,1}$ and $\hat{\mathbf{x}}_{f_j,2}$ are matched. The one belonging to the aligned map (say $\hat{\mathbf{x}}_{f_j,2}$) is used as a "pseudo-measurement" of the corresponding feature in the map of the other robot ($\hat{\mathbf{x}}_{f_i,1}$), i.e. $\hat{\mathbf{x}}_{f_j,2}$ is treated like a measurement of the state component $\hat{\mathbf{x}}_{f_i,1}$. In order to merge the information contained in both estimates, an update step of the EKF is performed by processing the pseudo-measurement $\hat{\mathbf{x}}_{f_j,2}$. Finally, the feature $\hat{\mathbf{x}}_{f_j,2}$ is removed from the vector $\mathbf{X}^a$. This procedure is repeated for all matching features between the two aligned maps. The resulting new merged map can now be used to perform again single-robot SLAM, until a new rendezvous occurs.

V. Experimental results

The multi-robot SLAM algorithm has been tested both in simulation and in real-world experiments.

A first evaluation of the proposed technique has been done with a custom simulator, developed in the MATLAB environment. The robots are modeled as unicycles and are equipped with a laser range finder. The simulator can handle CAD maps of indoor environments. The SLAM algorithm has been tested on a simplified map of the "S. Niccolò" building hosting the Department of Information Engineering of the University of Siena, a 3000 m^2 scenario. The setting of the experiments is the following. The robot motion noise covariance matrix Q is diagonal, where the standard deviations of the speed errors are set to be proportional to the absolute value of the speeds, i.e. $Q(k) = \text{diag}(\sigma_v^2(k), \sigma_\omega^2(k))$, where $\sigma_v(k) = 0.03|v(k)|\ (m/s)$ and $\sigma_\omega(k) = 0.05|\omega(k)| + 0.0017\ (rad/s)$. The laser measurements are simulated by a raytracing algorithm applied to the CAD map. The covariance matrix of the measurement noise associated to each raw range and bearing laser reading is $R_l = \text{diag}\left(\sigma_r^2,\ \sigma_b^2\right)$, where $\sigma_r = 0.003\ (m)$ and $\sigma_b = 0.003\ (rad)$. The measurement $\mathbf{m}(k)$ in (3) are extracted from the raw laser data, together with the associated error covariance matrix $R_\mathbf{m}(k)$, via segmentation procedure. In order to determine whether a measurement $\mathbf{m}(k)$ refers to a line already present in the map or it is a newly detected feature, a data-association algorithm similar to the one described in IV-C is employed. To avoid including spurious features in the map, new lines are



(a) (b)

Fig. 3. The matching duplicate features are used as constraints to improve the map alignment: (a) the fused map before imposing the constraints (b) the fused map after imposing the constraints.

first inserted in a tentative list until they are deemed reliable enough. The robot-to-robot measurement noise covariance matrix is given by (6), where $\sigma_\eta = 0.02$ (m), $\sigma_{1_{\phi_2}} = \sigma_{2_{\phi_1}} = 0.07$ (rad). Notice that the robot-to-robot observations are much less accurate than the laser raw readings, as it actually occurs in practical experiments (all the noise covariances have been tuned according to the mobile robot Pioneer 3AT and its sensory equipment).

Figures 3-4 report the results of a typical run. The robots first explore the area around two different courtyards, and then meet to fuse the two maps. The robot R_1 closes the loop around one courtyard at $t = 530$ s, and robot R_2 closes its loop around the other courtyard at $t = 590$ s. They meet at $t = 925$ s: if the experiment ended at this time, each robot would have the map of almost the entire floor. Then, the experiment goes on, and each robot explores the area covered before by the other robot, remaining localized quite well, by exploiting the information achieved at the rendezvous. The real-time experiment would take about 26 minutes. In Figure 3, the light lines are the ground-truth CAD map, while the thick lines are the features in the robot map. The 3σ confidence ellipses represent the uncertainty on the detected segment endpoints in the feature space, resulting from matrix P^a in (13) after the elimination of duplicate features.

As described in Section IV-C, the duplicate features that are detected in the merged map after the alignment are used as constraints to improve the map fusion. The effect of this process is shown in Figure 3. Figure 3(a) shows the map of robot R_1 (bottom, blue) and that of robot R_2 (top, red) transformed in the frame $\langle G_1 \rangle$ at $t = 925$ s (rendezvous): the inherited map is notably misaligned because of the high robot-to-robot measurement noise. The 3σ confidence ellipses (drawn only on detected endpoints) are bigger as

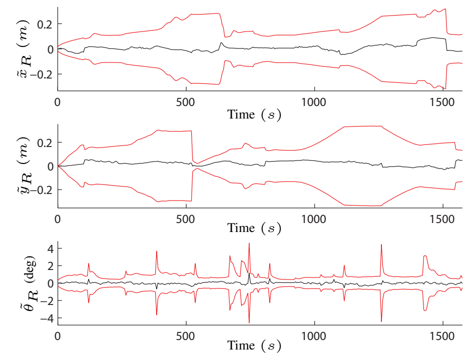


Fig. 4. The robots $\tilde{x}_R$, $\tilde{y}_R$ and $\tilde{\theta}_R$ localization errors, along with the correspondent 3σ confidence intervals (robot R_1).

they get far from the point of the rendezvous, since the angular error is amplified by the distance from the origin of the reference frame (lever-arm effect). After imposing the constraints, the map alignment is improved and the uncertainty is significantly reduced, as shown in Figure 3(b).

At the end of the experiment, the vector Ξ of each robot contains about 200 features. The total number of detected segment endpoints is about 140. The x, y and θ localization errors of robot R_1 are shown in Figure 4, along with the 3σ confidence intervals. The robot R_1 position and orientation error is less than 6 cm and 0.1 degrees, respectively. The corresponding errors for robot R_2 are less than 2 cm and 0.2 degrees.

In the map estimated by robot R_1, the average error on line orientation is less than 0.3 degrees, while that on line distance from the origin is less than 8 cm. For those lines whose endpoints have been detected, the average endpoints position error is less than 7 cm. The corresponding values for the map estimated by robot R_2 are less than 10 cm, 0.35 degrees and 9 cm, respectively. For both maps, the inconsistent features (detected endpoints out of their 3σ confidence ellipses) are about the 1.5% of the total number of features whose endpoints have been estimated.

A set of experiments with real robots has also been performed. To this purpose, two mobile robots Pioneer 3AT have been used to build a map of the second floor of the S. Niccolo' building. The parameters of the filter running on the robots have the same value as the simulations described before. The path followed by each vehicle is similar to that chosen for the simulations, i.e. each robot explores one of the two courtyard and then exchanges its map with the other agent to perform map fusion. Figures 5 and 6 show a comparison of the final map built by robot R_1 in two different setups. The map depicted in Figure 5 results from the processing of the sole measurements taken by robot R_1, whereas in Figure 6 the final map provided by the multi-robot SLAM algorithm is shown. A detailed quantitative comparison is not possible since the ground truth of the robots and an accurate map of the building are not available. However, a look at Figures 5 and 6 suggests that the multi-robot algorithm is actually able to provide an improvement

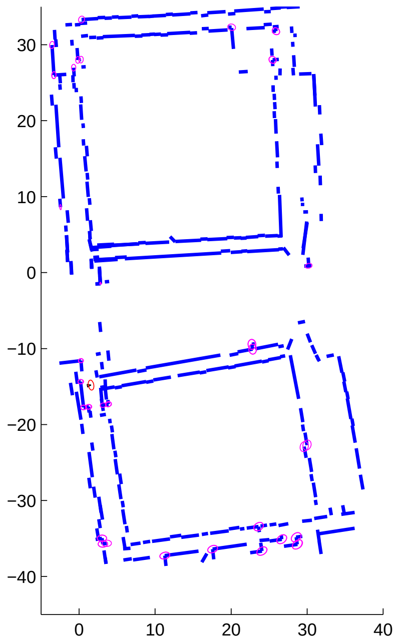


Fig. 5. Final map at the end of a *single-robot* SLAM experiment in the S. Niccolò building (robot R_1).

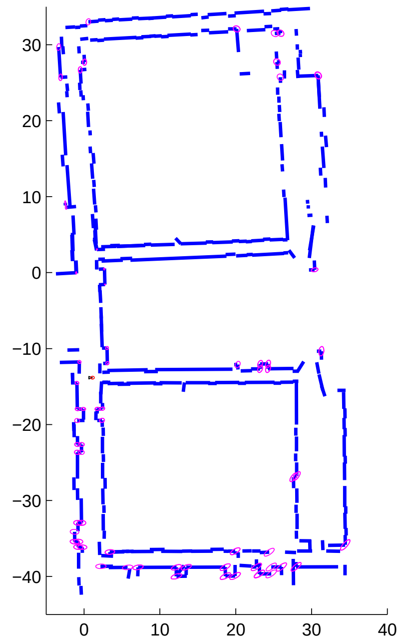


Fig. 6. Final map at the end of a *multi-robot* SLAM experiment in the S. Niccolò building (robot R_1).

to the built map, also in real environments and with real robots. In fact, in spite of a number of uncertainty sources present in reality and often neglected in simulations, the employment of two coordinated robots results in a more reliable map, as is testified by the better alignment of the two courtyards. The experimental results obtained are in good agreement with the simulations previously performed. In the multi-vehicle case, the final map is composed of about 300 features, the average area of the 99% confidence ellipses of the line endpoint estimates being smaller $5\ cm^2$. At the end of the experiment, the uncertainty affecting the estimate of the robot pose is smaller than $10\ cm$ and $0.5\ deg$ for the x, y and θ coordinates, respectively.

VI. Conclusions and Future Work

A multi-robot SLAM algorithm exploiting M-Space feature representation has been proposed. The algorithm relies on a map fusion scheme based on robot-to-robot mutual observations. No information on the initial relative location of the robots is required. Experimental tests performed so far show that the proposed technique can be effectively used for exploration and map building purposes. Besides reducing the time required for environment exploration, a major benefit is the increased ability of closing loops, thanks to the additional information exchanged at the rendezvous.

Several aspects of the proposed approach are currently under investigation. The map quality provided by the single- and multi-robot SLAM algorithms is being analyzed, as a function of the accuracy of both the measurements and the robot-to-robot observations. The map description will be enriched by including new kind of features, such as corners or poles, thanks to the flexibility of the M-Space representation. Simulations and experimental tests with larger teams of robots are also planned.

References

[1] S. Thrun. Robotic mapping: a survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002.

[2] A. T. Pfister, S. I. Roumeliotis, and J. W. Burdick. Weighted line fitting algorithms for mobile robot map building and efficient data representation. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, pages 1304–1311, 2003.

[3] A. Garulli, A. Giannitrapani, A. Rossi, and A. Vicino. Mobile robot SLAM for line-based environment representation. In *Proceedings of the 44th IEEE Conference on Decision and Control and 2005 European Control Conference*, pages 2041–2046, 2005.

[4] J. A. Castellanos and J. D. Tardos. *Mobile Robot Localization and Map Building: A Multisensor Fusion Approach*. Kluwer Academic Publisher, Boston, 1999.

[5] J. Folkesson, P. Jensfelt, and H.I. Christensen. The M-space feature representation for SLAM. *IEEE Transactions on Robotics*, 23(5):1024–1035, 2007.

[6] J.W. Fenwick, P.M. Newman, and J.J. Leonard. Cooperative concurrent mapping and localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1810–1817, 2002.

[7] S.B. Williams, G. Dissanayake, and H. Durrant-Whyte. Towards multi-vehicle simultaneous localisation and mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2743–2748, 2002.

[8] S. Thrun and Y. Liu. Multi-robot SLAM with sparse extended information filters. In *Proceedings of the International Symposium of Robotics Research (ISRR'03)*. Springer, 2003.

[9] A. Howard. Multi-robot simultaneous localization and mapping using particle filters. *The International Journal of Robotics Research*, 25(12):1243–1256, 2006.

[10] 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*, 19(2):238–249, April 2003.

[11] X.S. Zhou and S.I. Roumeliotis. Multi-robot SLAM with unknown initial correspondence: The robot rendezvous case. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1785–1792, 2006.

[12] D. Benedettelli. Multi-Robot SLAM using M-Space feature representation. Master's thesis, Faculty of Engineering, University of Siena, 2009. Available at http://robotics.benedettelli.com.

[13] Y. Bar-Shalom and T.E Fortman. *Tracking and data association*. Academic Press, 1988.