Mutual Localization of a Multi-Robot Team with Anonymous Relative Position Measures

Antonio Franchi Giuseppe Oriolo Paolo Stegagno
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Ariosto 25, 00185 Roma, Italy
{franchi,oriolo,stegagno}@dis.uniroma1.it

Abstract

In this paper we formulate and solve the mutual localization problem for a multi-robot system under the assumption of anonymous relative position measures. The anonymity hypothesis can cause a combinatorial ambiguity in the inversion of the measure equation giving more than one possible solution to the problem. We propose MultiReg, an innovative algorithm aimed at obtaining sets of possible relative pose hypotheses, whose output is processed by a data associator and a multiple EKF to select the best hypothesis. We study the performance of the developed localization system using both simulations and real robot experiments.

Keywords: multi-robot systems, mutual localization, multiple registration.

1 Introduction

In this paper we formulate and solve *mutual localization* (ML) problems for a multi-robot system. ML problems assume great importance in decentralized tasks that use data fusion, such as decentralized cooperative map-building, formation control and cooperative tracking. Accuracy of the estimates of the change of coordinates can significantly affect the quality of the execution of the task.

We refer to *relative mutual localization* (RML) as the problem of estimating the pose of a robot in the frame attached to another robot. Assuming that each robot has its own fixed frame in which it expresses its configuration and measures, we can also define *absolute mutual localization* (AML), i.e., the problem of estimating the change of coordinates between fixed frames of different robots. If each robot is self-localized in its own fixed frame, the solution of a problem can be obtained from the solution of the other by simple changes of coordinates.

Previous works usually address the problem of cooperative localization (CL) of robots in a common[1] fixed frame or the RML problem, while, to the best of our knowledge, no researchers have directly investigated the AML problem. In these works, two different classes of approaches are used: *i) filter-based* approaches, that use Extended Kalman Filters (EKF) or particle filters to dynamically estimate changes of coordinates from measures; *ii) geometry-based* approaches, that use geometric relationships to attempt an instantaneous inversion of the mapping between changes of coordinates and measures.

In most of the early filter-based approaches, like in [?, ?, ?, ?], the RML problem is solved by filtering out the noise from the output of a vision-based sensor that measures directly the relative poses between robots; at the same time, the filter is used to solve the CL problem. In other works, the filter was also used to reconstruct a part of the change of coordinates, like in [?], where relative range-only measures, obtained by the use of a combined RF/ultrasonic sensor, is used; in [?], where an extension of [?] was presented for various sensing equipments; and in [?], where a more detailed analysis is performed for range-only measurement.

As for geometry-based approaches, some papers investigate the solvability of the problem of estimating the relative positions of robots in a formation by range-only measures [?, ?], or bearing-only measures [?]. For example, in case of position (bearing plus range) relative measurements, it is possible to obtain the relative pose of a robot respect to another by simply processing two bearing and one range measure [?].

An important limitation of all the above methods is their assumption

[1] The idea of a common fixed frame presumes a certain degree of centralization, because it is necessary that robots share some information at the beginning of the task.

that the relative measures come with the ID of the measured robot, i.e., the identity of the measured robots is known. In fact, interesting situations that may occur in practice are: *i)* the identities of the measured robots is not known (anonymous measures), *ii)* false positives and *iii)* false negatives occur in the relative position measurement process. The first and the second situation fit, for example, robot measurement systems based on a feature extraction module that looks for characteristics that are common to all robots and may also be found in other objects: for example, this happens when the robots and some obstacle in the environment have the same size, color, or shape, either by chance or by hostile camouflage. The third situation accounts for the fact that robots in sensor range may not be sensed, e.g., due to occlusions.

A pioneering work that addresses the anonymous measure RML problem with a geometry-based approach is [**?**], in which an algorithm that uses geometrical properties of triangles is presented to obtain relative pose estimates from anonymous bearing measurements. The omnidirectional sensor hypothesis, however, can prevent the application to real world environments because it prevents the use of a wide class of sensors that cannot get measurements behind obstacles, such as range finders.

In this paper we address RML and AML problems with anonymous measures affected by false positives and negatives, as formalized in Sect. **??**. The localization system architecture is explained in Sect. **??**. Our approach solves the RML problem with a geometry-based algorithm (Sect. **??**) whose output is used to solve the AML problem with a multiple EKF (Sect. **??**). Experimental results are presented in Sect. **??**.

2 Problem Setting

We take the following assumptions:

A1. The multi-robot system $\mathcal{A}$ includes n robots $\mathcal{A}_1, \ldots, \mathcal{A}_n$, where $n > 2$ is unknown[2]. The robots move in $\mathbb{R}^2$. We denote with $\mathcal{I}$ the index set $\{1, \ldots, n\}$.

A2. The configuration of robot $\mathcal{A}_i$ is $x_i = (p_i^T, \theta_i)^T$, where $p_i \in \mathbb{R}^2$ and $\theta \in S^1$ are respectively the cartesian coordinates of a representative point of the robot and the robot orientation, expressed in a certain frame. In particular each $\mathcal{A}_i$ has two associated frames: a *fixed* frame $\mathcal{F}_I$ and a *moving* frame $\mathcal{F}_i$. The latter is attached to the robot: its origin is at p_i and its orientation coincides with θ_i. Throughout the paper, we denote by ${}^a t_b$, $a = i, I$, $b = j, J$ the 3-vector describing the position and orientation of $\mathcal{F}_b$ with respect to $\mathcal{F}_a$. From ${}^a t_b$, it

[2]Case $n = 2$ is trivial because anonymity disappears.

is immediate to build the change of coordinates from $\mathcal{F}_b$ to $\mathcal{F}_a$. Note that ${}^a t_j = {}^a x_j$, i.e., the configuration of $\mathcal{A}_j$ expressed in $\mathcal{F}_a$.

A3. Each $\mathcal{A}_i$ comes with an independent *self-localization module* that provides an estimate ${}^I\hat{x}_i$ of ${}^I x_i$.

A4. Each $\mathcal{A}_i$ is equipped with a *robot detector*, a sensor device that measures the *relative position* ${}^i p_j$ of other robots provided that they belong to a perception set D_p which depends on x_i (typically, it is 'centered' at p_i and 'oriented' as θ_i).

A5. Each $\mathcal{A}_i$ has a *communication module* that can send/receive data to/from any other robot $\mathcal{A}_j$ that belongs to a communication set D_c, that is a ball centered at p_i. Each message sent by $\mathcal{A}_i$ contains: *i)* the robot ID; *ii)* the estimate ${}^I\hat{x}_i$ as provided by the self-localization module; *iii)* the relative position of other robots as measured by the robot detector. We assume that $D_p \subseteq D_c$, so that if $\mathcal{A}_i$ can measure the relative position of $\mathcal{A}_j$ it can also communicate with it.

The relative position measures provided by the robot detector are *anonymous*, in the sense that they do not include the ID of the detected robot. This is true, for example, when the robot detection process relies on features that are identical in the robots of the team. The robot detector is also prone to *false positives*, in the sense that it can be deceived by objects that look like robots. Moreover, *false negatives* may also occur: this is the case of a robot belonging to D_p which is not detected, e.g., due to a line-of-sight occlusion. For all these reasons, the measures coming from the robot detector will be generically referred to as *features*.

In the above framework, the *absolute mutual localization* problem for the i-th robot is the estimation of ${}^I t_J$ for $j \in \mathcal{I}$, $j \neq i$. Note that anonymity of the relative position measure is a major problem.

3 System Architecture

The mutual localization system running on $\mathcal{A}_i$ is shown in Fig. **??**. We denote with $C_i[k] \subset \mathcal{A}$ the set of robots from which $\mathcal{A}_i$ receives data in the time interval $[t_{k-1}, t_k)$. The system is composed by a cascade of two subsystems. The first is a memoryless registration algorithm called *MultiReg*. At each update step k it receives in input a certain number of feature sets: one set is provided directly by the robot detector, while others come from the robots in $C_i[k]$ through the communication module. The output of MultiReg is a sets of hypotheses on the relative pose ${}^i x_j$ for each $\mathcal{A}_j \in C_i[k]$. The second subsystem (DAEKF) is a variable-size array of components, one for each $\mathcal{A}_j \in \cup_{h=1}^{k} C[h]$, consisting of a *data associator* and a *multi-EKF*

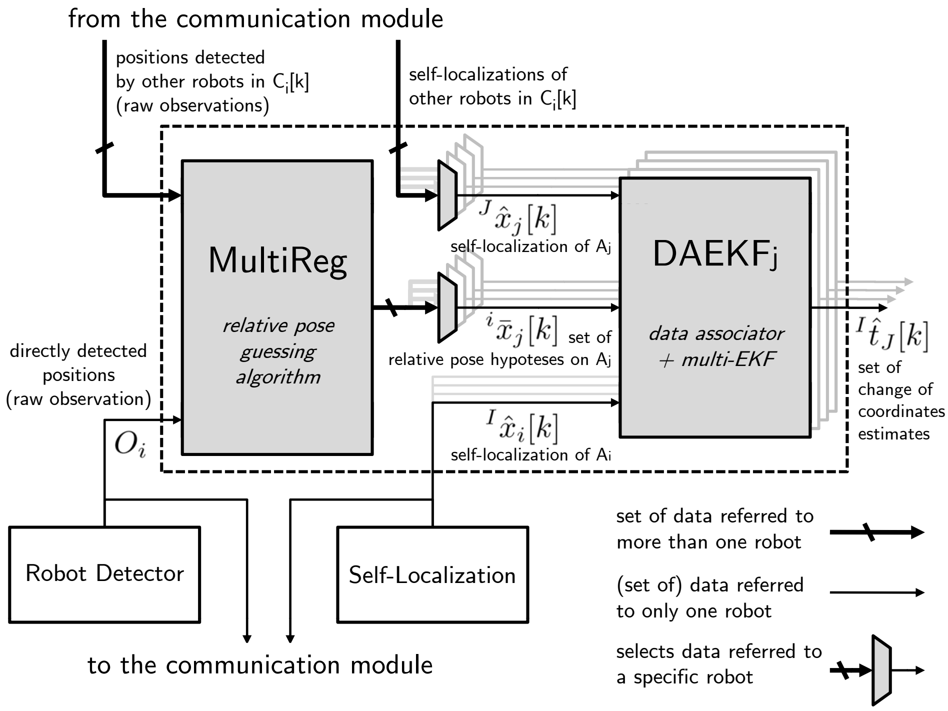


Figure 1: A scheme of the mutual localization system that runs on $\mathcal{A}_i$.

(DAEKF$_j$). The input of DAEKF$_j$ at the step k is the set of current hypotheses on $^i x_j$. At the same time each DAEKF$_j$ also receives the estimates $^I\hat{x}_i$ and $^J\hat{x}_j$, respectively from the self-localization and the communication module. At each step k the output of the overall localization system is a set of of hypotheses for $^I t_J$ for each $\mathcal{A}_j \in \cup_{h=1}^{k} C_i[h]$. The number of hypotheses in each set changes as the localization go on.

4 Relative Pose Guessing with MultiReg

MultiReg is restarted at each step k by the generic robot $\mathcal{A}_i$ to perform a multiple registration among the set of features measured, at the same step k, by $\mathcal{A}_i$ and by each robot $\mathcal{A}_j \in C_i[k]$, in order to estimate the changes of coordinates between their attached frames. It uses a binary registration subroutine based on RANSAC paradigm [**?**], however it works with every binary registration algorithm.

We call *observation* a pair $O := (F, A)$, where F is a finite set of features and $A \subset F \times \mathcal{I}$ is a functional relation on $F \times \mathcal{I}$. Given $f \in F$ we denote with $A(f) := \{i \in \mathcal{I} | \ (f, i) \in A\}$ and $\forall i \in \mathcal{I}$ we denote with $A(i) := \{f \in F | \ (f, i) \in A\}$. Functional means that $|A(f)| \leq 1$, where $| \cdot |$ denotes the cardinality of a set. We denote by $A(F) := \cup_{f \in F} A(f)$ the set of indexes of O and by $A(\mathcal{I}) := \cup_{i\in\mathcal{I}} A(i)$ the set of features of O associated to any index. This abuse of notation will be used throughout the paper. A feature f is called *anonymous* when $A(f) = \emptyset$, i.e., $f \notin A(\mathcal{I})$. Two observations $O_1 = (F_1, A_1)$ and $O_2 = (F_2, A_2)$ are *independent* if they relate at least one common feature to two different indexes, that is $\exists f \in A_1(\mathcal{I}) \cap A_2(\mathcal{I})$ such that $A_1(f) \neq A_2(f)$, or if two different features are related to the same index, that is $\exists f_1 \in A_1(\mathcal{I}), f_2 \in A_2(\mathcal{I})$, with $f_1 \neq f_2$, such that $A_1(f_1) = A_2(f_2)$. Given a change of coordinates T and a set of features F we can write $T(F) = \{q \in \mathbb{R}^2 | \exists f \in F : T(f) = q\}$.

Each robot detector provides an observation in which all features are anonymous, except for the feature $(0,0)$ associated to the ID of the measuring robot; this is called *raw observation*.

4.1 RANSAC-based binary registration

Given two observations $O_1 = (F_1, A_1)$ and $O_2 = (F_2, A_2)$, such that[3] $A_1(F_1) \cap A_2(F_2) = \emptyset$, consider a candidate change of coordinates T between the two associated frames. A binary relation $B \subset F_1 \times F_2$ is associated to T as follows: $(f_1, f_2) \in B \Leftrightarrow \|f_1 - T(f_2)\| \leq \delta$, where δ is a certain *fitting threshold*. The elements of $B(F_1)$ and $B(F_2)$ are the *inliers* of F_2 and F_1 respectively. The cardinality of B, denoted by $|B|$, is the *number of inliers*.

Given $\delta > 0$ and $\mu > 0$, a *binary registration* of O_1, O_2 is a complete algorithm for finding a change of coordinates T such that the associated B is left- and right-unique, and satisfies: *i)* $|B| \geq \mu$ and *ii)* $|A(f_1) \cup A_2(f_2)| \in \{0, 1\}$ $\forall (f_1, f_2) \in B$. The first condition is a constraint on the *minimum number of inliers* (note that $|B| = |B(F_1)| = |B(F_2)|$). The second states that, for any pair of features f_1, f_2 that are related by B, either f_1 or f_2 (or both) must be anonymous. In fact, being $A_1(F_1) \cap A_2(F_2) = \emptyset$, a 'double' assignment would certainly represent a conflict.

In view of the fact our multiple registration algorithm uses a binary registration as an essential tool, it is convenient to define a new observation $O_{12} = (F_{12}, A_{12})$ where $F_{12} = F_1 \cup T(F_2)$ and $A_{12} \subset F_{12} \times \mathcal{I}$ is such that

$$A_{12}(f) = \begin{cases} A_1(f) & \text{if } f \in A_1(\mathcal{I}) \\ A_2(f) & \text{if } f \in A_2(\mathcal{I}) \\ A_1(f) \cup A_2(B(f)) & \text{if } f \in B(F_2) \\ A_1(B(f)) \cup A_2(f) & \text{if } f \in B(F_1) \\ \emptyset & \text{otherwise} \end{cases}$$

for any $f \in F_{12}$. For our purposes, the output of the algorithm (called *solution* in the following) is the triple $(T, O_{12}, |B|)$.

Clearly, for a given pair of observation there could exist more than a solution. Two solution are said *independent* if their corresponding observations

[3]As it will be clear in the following, this assumption is always satisfied in our multiple registration algorithm.

inputs

$O_1 = (F_1, A_1), O_2 = (F_2, A_2)$ — 2 observations

parameters		**variables**	
I	max. number of iterations	T	change of coordinates
δ	fitting threshold	O_{12}	observation
μ	min. number of inliers	B	relation of inliers
		D	equidistant pairs of features
		$(f_1^a, f_1^b, f_2^a, f_2^b)$	pairs of features

output

$\mathcal{R}(O_1, O_2) = \{\ldots, (O_{12j}, T_j, v_j), \ldots\}$ — set of solutions

algorithm

1. $\mathcal{R}(O_1, O_2) = \emptyset$
2. $D = \{(q, r, s, t) \in F_1 \times F_1 \times F_2 \times F_2 : \; \|q - r\| - \|s - t\| \leq 2\delta\}$
3. **while** h $\leq I$ **and** $D \neq \emptyset$
   a. extract randomly and without repetitions $(f_1^a, f_1^b, f_2^a, f_2^b)$ from D
   b. compute the change of coordinates T that aligns the segment $f_2^a f_2^b$ with the segment $f_1^a f_1^b$ and overlaps their middle points
   c. compute relation B from T and δ
   d. **if** $|B| \geq \mu$ **then**
      i. compute observation O_{12} from B and T
      ii. add $(T, O_{12}, |B|)$ to $\mathcal{R}(O_1, O_2)$
4. **return** $\mathcal{R}(O_1, O_2)$

Table 1: RANSAC-based binary registration algorithm

are independent. In the following is assumed that the binary registration algorithm returns a set of independent solutions.

The combinatorial essence of the problem suggests the use of probabilistic techniques, while the presence of outliers (i.e., features observed by only one robot) calls for a robust estimation paradigm. We chose RANSAC because is has both this properties. Our implementation (see Table **??**) follows from algorithm presented in [**?**] for a binary lidar scan registration.

4.2 Multiple registration

At each perception step k, each $\mathcal{A}_i$ executes MultiReg on the set Ω composed by its own raw observation O_i and the raw observations of the robots belonging to C_i[k]. In view of the fact that MultiReg is a memoryless algorithm, for compactness we omit in its description the symbol $[k]$. Denote with $\mathcal{I}_i^C$ the set of indexes of C_i, therefore $\Omega = \{\ldots, O_j, \ldots\}_{j \in \mathcal{I}_i^C \cup \{i\}}$, where

inputs

$\Omega = \{\ldots, O_j, \ldots\}$, with $O_j = (F_j, A_j)$ — $|C_i| + 1$ raw observations

variables

${}^i\bar{x}_{jl} = (\ldots, {}^i\bar{x}_{jhl}, \ldots)$ — partial solution at the l-th iteration

$\tilde{O}_l$ — partial registered observation at the l-th iteration

$U_l \subseteq \mathcal{I}_i^C$ — indexes of the unregistered observations at the l-th iteration

output

$X(\Omega) = \{\ldots, {}^i\bar{x}_{js}, \ldots\}$ — set of solutions (in shared memory)

algorithm

1. ${}^i\bar{x}_{j0} = (\mathbf{0}_3, \ldots, \mathbf{0}_3)$, $U_0 = \{1\}$, $X(\Omega) = \emptyset$
2. **for** l=1 **to** $|C|$
   a. $\Gamma = \bigcup_{O \in \Omega_{U_{l-1}}} \mathcal{R}(\tilde{O}_l, O)$
   b. $\Gamma^* = \{\gamma = (x_\gamma, o_\gamma, v_\gamma) \in \Gamma \mid v_\gamma = \max_v \Gamma\} \subseteq \Gamma$; this is the subset of Γ of elements that maximize the number of inliers
   c. compute the maximal subset of independent solutions $\tilde{\Gamma} \subseteq \Gamma^*$
   d. perform a least square estimation for every $\gamma \in \tilde{\Gamma}$, substituting x_γ with that minimizing the mean square error among the inliers and recomputing o_γ accordingly
   e. **fork** $\forall \gamma \in \tilde{\Gamma}$ with
      i. $\tilde{O}_l = o_\gamma$;
      ii. ${}^i\bar{x}_{jh} = ({}^i\bar{x}_{2(l-1)}, \ldots, {}^i\bar{x}_{|C|(l-1)})$
      iii. ${}^i\bar{x}_{hl} = x_\gamma$
      iv. $U_l = U_{l-1} \cup \{s\}$ where s is the index of the raw observation added in γ
3. $X(\Omega) = X(\Omega) \cup \sigma_n$

Table 2: MultiReg algorithm

$O_j = (F_j, A_j)$. As stated before it results $|A_j(F_j)| = 1$, $\forall j \in \mathcal{I}_i^C \cup \{i\}$, and $\bigcap_{j \in \mathcal{I}_i^C \cup \{i\}} A_j(F_j) = \emptyset$. We set $\Omega_j = \Omega \backslash O_j$ and, for any $\bar{C} \subseteq C_i \cup \{\mathcal{A}_i\}$ with indexes $\bar{\mathcal{I}}$, we set $\Omega_{\bar{C}} = \bigcap_{j \in \bar{\mathcal{I}}} \Omega_j$.

MultiReg output is a finite set $X(\Omega) = \{\ldots, {}^i\bar{x}_j, \ldots\}$ where ${}^i\bar{x}_j = \{\ldots, {}^i\bar{x}_{jh}, \ldots\}_{j \in \mathcal{I}_i^C}$, whose generic element ${}^i\bar{x}_{jh}$ is a hypothesis on ${}^it_j$. MultiReg executes $|C_i|$ iterations and at the end of each iteration, if necessary, it forks (possibly more than one time) copying itself and its own memory, and executes the next iteration with some different variables.

Step **??** initialize the first iteration (see Table **??**). At step **??** , during the l-th iteration, $|C_i| - l$ binary registrations $\mathcal{R}(o_l, O)$ are performed between the observation so far obtained, o_l, and the raw observations $O \in \Omega_{U_{l-1}}$.

Their solutions γ are stored in Γ. At step **??** the solutions with the maximum number of inliers are stored in Γ^*, and at step **??** $\tilde{\Gamma}$ is computed, that is the maximal subset of Γ^* of independent solutions. This is a key step because it guarantees the algorithm completeness. In fact two independent solution are always found in two different branches of the algorithm tree. At step **??** the estimated change of coordinates $x_\gamma \in \gamma$ is tuned, $\forall \gamma \in \tilde{\Gamma}$, minimizing the mean square error of the inliers pairs. This is done by the use of the algorithm in [**?**]. At step **??** MultiReg forks in $|\tilde{\Gamma}|$ branches, one for each $\gamma \in \tilde{\Gamma}$. Variables are consistently updated. Each branch of the algorithm ends at its n-th iteration queuing its solution to $X(\Omega)$, which is the only variable shared by all branches.

5 Data Association and Extended Kalman Filter

For ${}^b t_a = ({}^b p_a^T, {}^b\theta_a)^T$ define the matrix

$$S({}^b t_a) := \begin{pmatrix} R({}^b\theta_a) & \mathbf{0}_2 & {}^b p_a \\ \mathbf{0}_2^T & 1 & {}^b\theta_a \\ \mathbf{0}_2^T & 0 & 1 \end{pmatrix},$$

where $R({}^b\theta_a)$ is the rotation matrix associated to ${}^b\theta_a$. Since ${}^b t_i = {}^b x_i$, ${}^I t_J$ can be expressed as

$$ {}^I t_J = N S({}^I x_i[k]) S({}^i x_j[k]) S^{-1}({}^J x_j[k]) (\mathbf{0}_3^T \ 1)^T \tag{1}$$

where $N = (I_3 \ \mathbf{0}_3)$ is a selection matrix.

At the step k, the DAEKF subsystem running on robot $\mathcal{A}_i$ is composed by an array of $|\cup_{h=1}^k C_i[h]|$ components (see Fig. **??**). Each component is associated to a robot $\mathcal{A}_j \in \cup_{h=1}^k C_i[h]$ and it estimates ${}^I t_J$. At step k its inputs are:

1. The estimates provided by the self-localization modules

$$\begin{aligned} {}^J\hat{x}_j[k] &= {}^J x_j[k] + {}^J w_j[k] \\ {}^I\hat{x}_i[k] &= {}^I x_i[k] + {}^I w_i[k], \end{aligned}$$

   where ${}^J w_j[k]$ and ${}^I w_i[k]$ are gaussian noises with zero mean and covariances ${}^J\widehat{Q}_j[k]$ and ${}^I\widehat{Q}_i[k]$. Note that ${}^J\hat{x}_j[k]$ is available provided that $\mathcal{A}_j \in C_i[k]$;

2. The set of hypotheses about the relative pose ${}^i x_j[k]$

$${}^i\bar{x}_j[k] = \{\ldots, {}^i\bar{x}_{jh}[k], \ldots\},$$

   provided by MultiReg. Here, ${}^i\bar{x}_{jh}[k]$ is a gaussian random variable with unknown mean and covariance ${}^i\bar{Q}_{jh}[k]$. The probability ${}^i s_j[k]$ that at least one hypothesis of the set ${}^i\bar{x}_j[k]$ has mean ${}^i x_j[k]$ is known also from MultiReg.

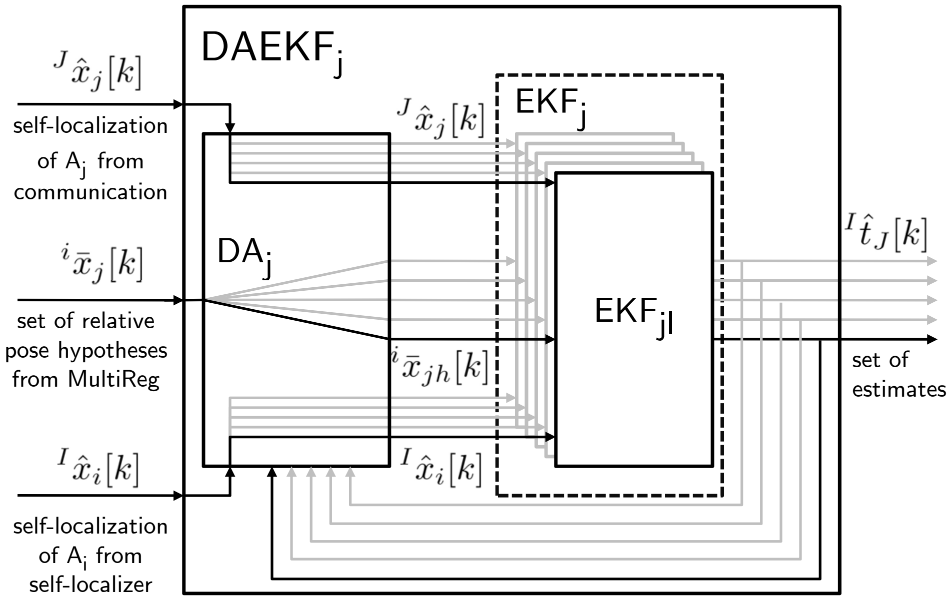


Figure 2: A scheme of the DAEKF$_j$ which estimates ^It_J.

Each DAEKF$_j$ (see Fig. **??**) is composed by a *data associator* (DA$_j$) and a variable-size *multi-EKF* EKF$_j$ = $\{\ldots, \text{EKF}_{jl}, \ldots\}$. The data associator [**?**] is a 'nearest neighbor like' memoryless algorithm in charge of dispatching each relative pose hypothesis $^i\bar{x}_{jh}[k]$ produced by MultiReg to the appropriate EKF$_{jl}$ of the array. This EKF$_{jl}$, taking as input $^J\hat{x}_j[k]$, $^I\hat{x}_i[k]$ and $^i\bar{x}_{jh}[k]$, produces an estimate $^I\hat{t}_{Jl}[k]$ of ^It_J with its covariance matrix. The pseudocode of DAEKF$_j$ is shown in Table **??**.

5.1 Data Association

At each step k, DA$_j$ also receives, as feedback, the sets

$$^I\hat{t}_J[k] = \{\ldots, {}^I\hat{t}_{Jl}[k], \ldots\}$$

of estimates of ^It_J produced by the corresponding multi-EKF. For each $^i\bar{x}_{jh}[k]$, using $^J\hat{x}_j[k]$ and $^I\hat{x}_i[k]$, DA$_j$ computes the following hypothesis on ^Jt_I based on (**??**):

$$^I\bar{t}_{Jh}[k] = NS({}^I\hat{x}_i[k])S({}^i\bar{x}_{jh}[k])S^{-1}({}^J\hat{x}_j[k])(\mathbf{0}_3^T\ 1)^T. \tag{2}$$

In addition, for each pair $(h,l) \in \{1,\ldots,|^i\bar{x}_j[k]|\} \times \{1,\ldots,|^I\hat{t}_J[k]|\} =: {}^i\mathcal{X}_j[k] \times {}^I\mathcal{T}_J[k]$, DA$_j$ computes the covariance-weighted distance

$$^id_{jhl}[k] = \sqrt{({}^I\bar{t}_{Jh} - {}^I\hat{t}_{Jl})\,{}^iD_{jh}[k]({}^I\bar{t}_{Jh} - {}^I\hat{t}_{Jl})^T} \tag{3}$$

where ${}^iD_{jh}[k] = {}^iV_{jh}[k]{}^iR_{jh}[k]{}^iV_{jh}[k]^T$, ${}^iV_{jh}[k]$ is the jacobian matrix of ${}^It_J$ with respect to $({}^Ix_i^T\ {}^ix_j^T\ {}^Jx_j^T)^T$, computed at $({}^I\hat{x}_i^T[k]\ {}^i\bar{x}_j^T[k]\ {}^J\hat{x}_j^T[k])^T$ whose expression will be showed in (**??**), and

$$R_{kh}[k] = \begin{pmatrix} {}^I\widehat{Q}_i[k] & 0 & 0 \\ 0 & {}^J\widehat{Q}_j[k] & 0 \\ 0 & 0 & {}^i\bar{Q}_{jh}[k] \end{pmatrix}.$$

Given a *maximum distance* $d_{\max}$ the DA_j searches for the largest left- and right-unique relation ${}^i\mathcal{R}_j^*[k] \subset {}^i\mathcal{X}_j[k] \times {}^I\mathcal{T}_J[k]$ minimizing $\sum_{(h,l)\in {}^i\mathcal{R}_j^*[k]} {}^id_{jhl}[k]$ and such that ${}^id_{jhl}[k] \leq d_{\max}$, $\forall (h,l) \in {}^i\mathcal{R}_j^*[k]$. This relation defines a rule by which, for each $(h,l) \in {}^i\mathcal{R}_j^*[k]$, the hypothesis ${}^i\bar{x}_{jh}[k]$ is dispatched to EKF_l. For each hypothesis ${}^i\bar{x}_{jm}[k]$ which is not associated through ${}^i\mathcal{R}_j^*[k]$, a new filter is added to EKF_j, initialized with the corresponding triple $\{{}^i\bar{x}_{jm}[k], {}^I\hat{x}_i[k], {}^J\hat{x}_j^T[k]\}$. At each step, a *mark* is associated to each EKF_{jl}, i.e., the number of *hits* (steps in which ${}^i\bar{x}_j[k] \neq \emptyset$ and $l \in {}^i\mathcal{R}_j^*[k]({}^i\mathcal{T}_j[k])$) in the last $L[k]$ steps. The *backward horizon* $L[k]$ is chosen so as to guarantee that $1 - \prod_{h=k-L[k]}^{k}(1 - {}^is_j[k]) \geq s_{\min}$, i.e., the probability that in $[k - L[k], k]$ there is at least a measure with the 'right' mean is larger than $s_{\min}$. An EKF_{jl} whose mark goes below a certain threshold $\mu_{\min}$ is removed from the array EKF_j. The EKF_{jl} with the highest mark provides the best current estimate of ${}^It_J$.

5.2 Extended Kalman Filter

The generic EKF (omitting for compactness the subscripts jl) is used to estimate the constant parameter ${}^It_J$, with model equation ${}^It_J[k] = {}^It_J[k-1]$ and measurement equation given by **??**. Denoting by $K[k]$ the gain and by $P[k]$ the covariance estimate , the filter equations are:

$$\begin{aligned} K[k] &= P[k-1](P[k-1] + V[k]R_{kh}[k]V[k]^T)^{-1} \\ {}^I\hat{t}_J[k] &= {}^I\hat{t}_J[k-1] + K[k]({}^I\bar{t}_{Jh}[k] - {}^I\hat{t}_J[k-1]) \\ P[k] &= (I_3 - K[k])P[k-1] \end{aligned}$$

where the jacobian matrix $V[k]$ has the following expression

$$\begin{pmatrix} I_2 & R({}^I\bar{\theta}_j[k]){}^i\bar{p}_j[k] - R^*(\Theta[k]){}^J\hat{p}_j[k] & -R(\Theta[k]) \\ \mathbf{0}_2^T & 1 & \mathbf{0}_2^T \end{pmatrix.$$

$$\begin{matrix} R^*(\Theta[k]){}^J\hat{p}_j[k] & R({}^I\theta_i[k]) & -R^*(\Theta[k]){}^J\hat{p}_j[k] \\ -1 & \mathbf{0}_2^T & 1 \end{matrix}\Bigg) \tag{4}$$

where $\Theta[k] := {}^I\hat{\theta}_i[k] + {}^i\bar{\theta}_j[k] - {}^J\hat{\theta}_j[k]$ and

$$R^*(\phi) := \begin{pmatrix} -\sin\phi & -\cos\phi \\ \cos\phi & -\sin\phi \end{pmatrix}.$$

inputs

${}^{i}\bar{x}_j[k]$	set of relative pose hypotheses as provided by MultiReg
${}^{I}\hat{x}_i[k]$	estimate of ${}^{I}t_i[k]$ as provided by the self-localization
${}^{J}\hat{x}_j[k]$	estimate of ${}^{J}t_j[k]$ estimate of ${}^{I}t_i[k]$ as provided by the communication

parameters

$d_{\max}$	max. hypo. distance
$s_{\min}$	backward horizon prob.
$\mu_{\min}$	survival mark threshold

output

$\{\ldots, {}^{I}\hat{t}_{Jjl}[k], \ldots\}$	set of estimates of ${}^{I}t_J$

algorithm (k-th step)

1. **for** $h = 1$ **to** $|{}^{i}\bar{x}_j[k]|$
   a. compute ${}^{i}\bar{t}_{jh}[k]$ from equation **??**
   b. **for** $l = 1$ **to** $|\mathrm{EKF}_j[k]|$ compute ${}^{i}d_{jhl}[k]$ from equation **??**
2. compute the routing relation ${}^{i}\mathcal{R}^{*}_{j}[k]$
3. **for** $h = 1$ **to** $|{}^{i}\bar{x}_j[k]|$
   a. **if** $h \in {}^{i}\mathcal{R}^{*}_{j}[k]({}^{i}\mathcal{X}_j[k])$ **then**
      i. dispatch ${}^{i}\bar{x}_{jh}[k]$ to $\mathrm{EKF}_{j,{}^{i}\mathcal{R}^{*}_{j}[k](h)}$
      ii. update the mark μ_l of $\mathrm{EKF}_{j,{}^{i}\mathcal{R}^{*}_{j}[k](h)}$
   b. **else** add an EKF and initialize it with ${}^{i}\bar{x}_{jh}[k]$
4. **for** $l = 1$ **to** $|\mathrm{EKF}_l[k]|$
   i. **if** $\mu_l > \bar{\mu}$ **then** remove EKF_{jl} from EKF_j
   ii. **else** make a step of the EKF_{jl} producing ${}^{I}\hat{t}_{Jjl}[k]$

Table 3: DAEKF_j algorithm

6 Experiments

We have validated our approach using the software platform described in [**?**], both simulating robots with Player/Stage and using a team of 5 Khepera III real robots. In the following, we refer to real robot experiments. Each robot is equipped with a Hukuyo URG-04LX laser scan with a 240° angular range, 0.33° resolution and range artificially limited up to 2 meters. Due to its resolution, at this maximum distance the URG can detect only obstacles of size greater than 1.2 cm. The robot detector is a simple feature extraction algorithm that looks for a cardboard ‘hat’ placed on top of each robot around the sensor. In particular, the detector identifies $1 \div 12$ cm wide protrusions

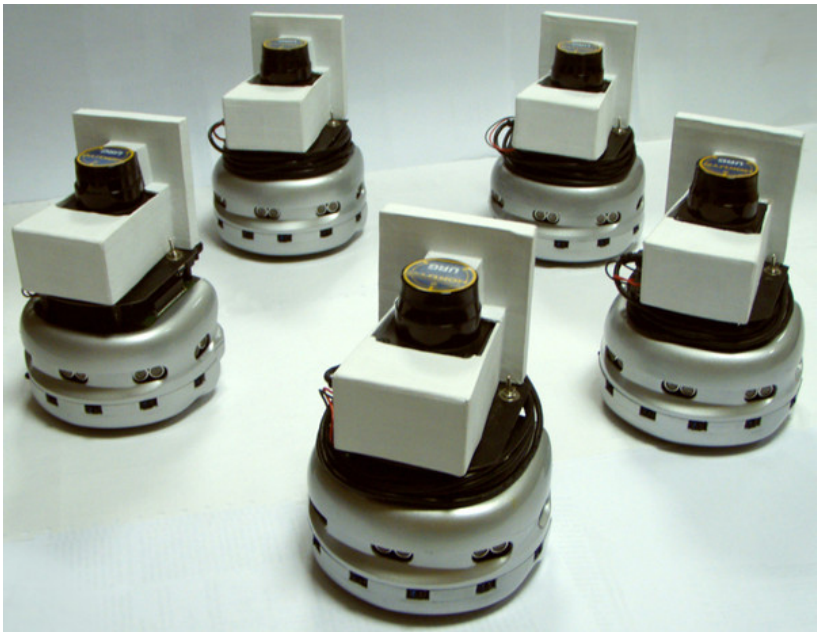


Figure 3: The team of 5 Khepera III used in our experiments. Around the URG-04LX a cardboard 'hat' is mounted to allow the feature extraction.

and it is thus unable to distinguish among different robots, or robots and obstacles whose size is in the same range. This variation range accounts for the fact the 'hat' gives a protrusion whose width depends on the robot orientation.

Accurate measures of the ${}^{I}t_{J}$ to be used as ground truth are taken before the experiments by a human operator. The self-localization is a simple dead reckoning.

In Fig. **??** we show the early steps of an experiment conducted with 5 robots and 4 deceiving obstacles, starting from a highly symmetric arrangement (that is very ambiguous for MultiReg) moving for 5 minutes. The mutual localization system runs on robot $\mathcal{A}_4$ at 10Hz, which is the frequency of the URG. While at the start the best estimates of $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$ are wrong, due to occlusions and symmetry, in a few steps the correct estimates prevail.

Figure **??** summarizes the experiments in terms of the errors (cartesian an angular) and marks of the estimates generated by all the EKF$_j$, for $j = 0, \ldots, 3$. The timescale is 150 sec. The estimates whose normalized mark goes below 0.15 are removed. Figures **??** and **??** refer to another experiment, whose peculiarity is that 2 robots act as deceiving movable obstacles, since they move but do not broadcast their observations. As we expect, the estimation system works well also in this case. Videos of the experiments and more material could be found at `http://www.dis.uniroma1.it/~labrob/research/mutLoc.html`

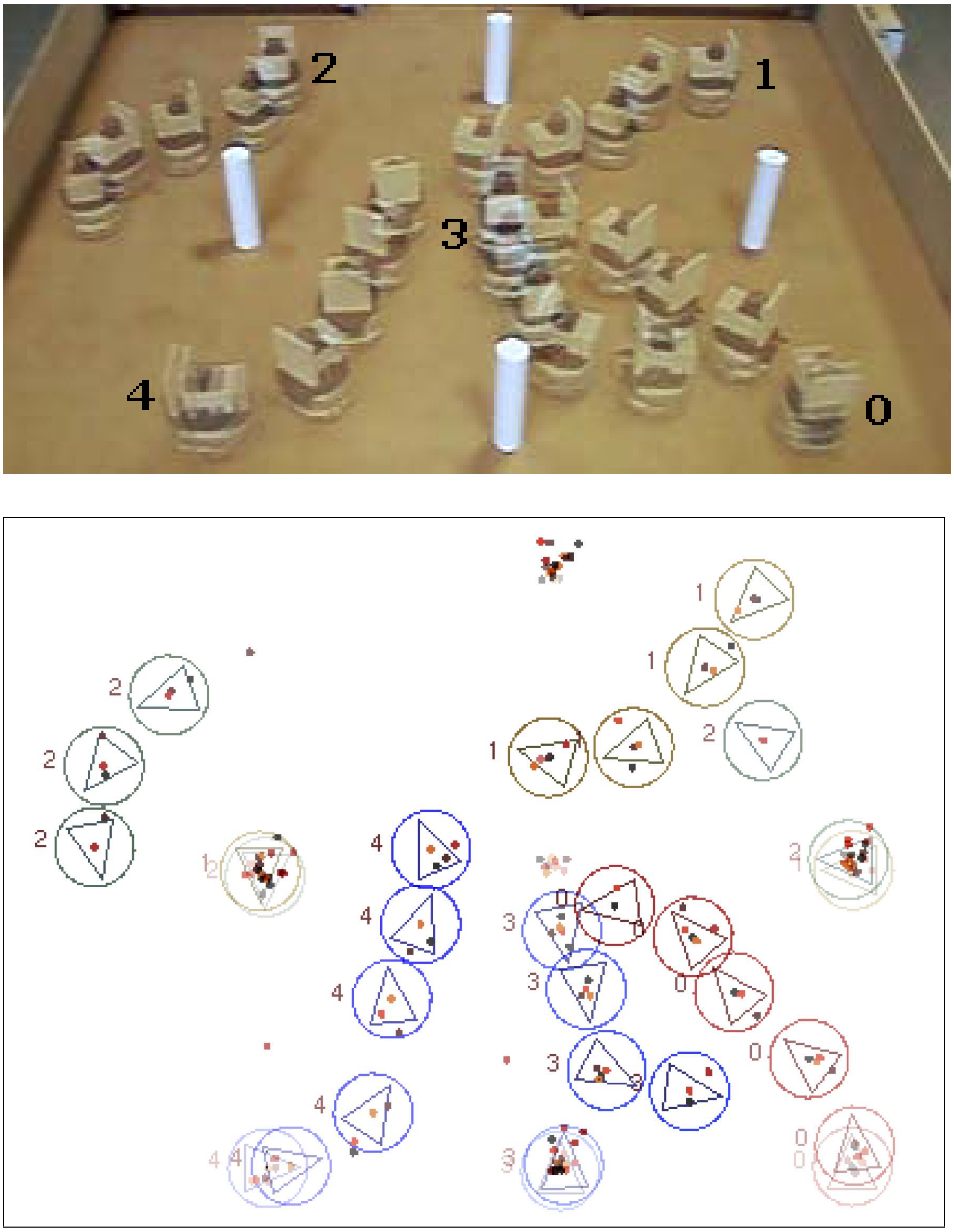
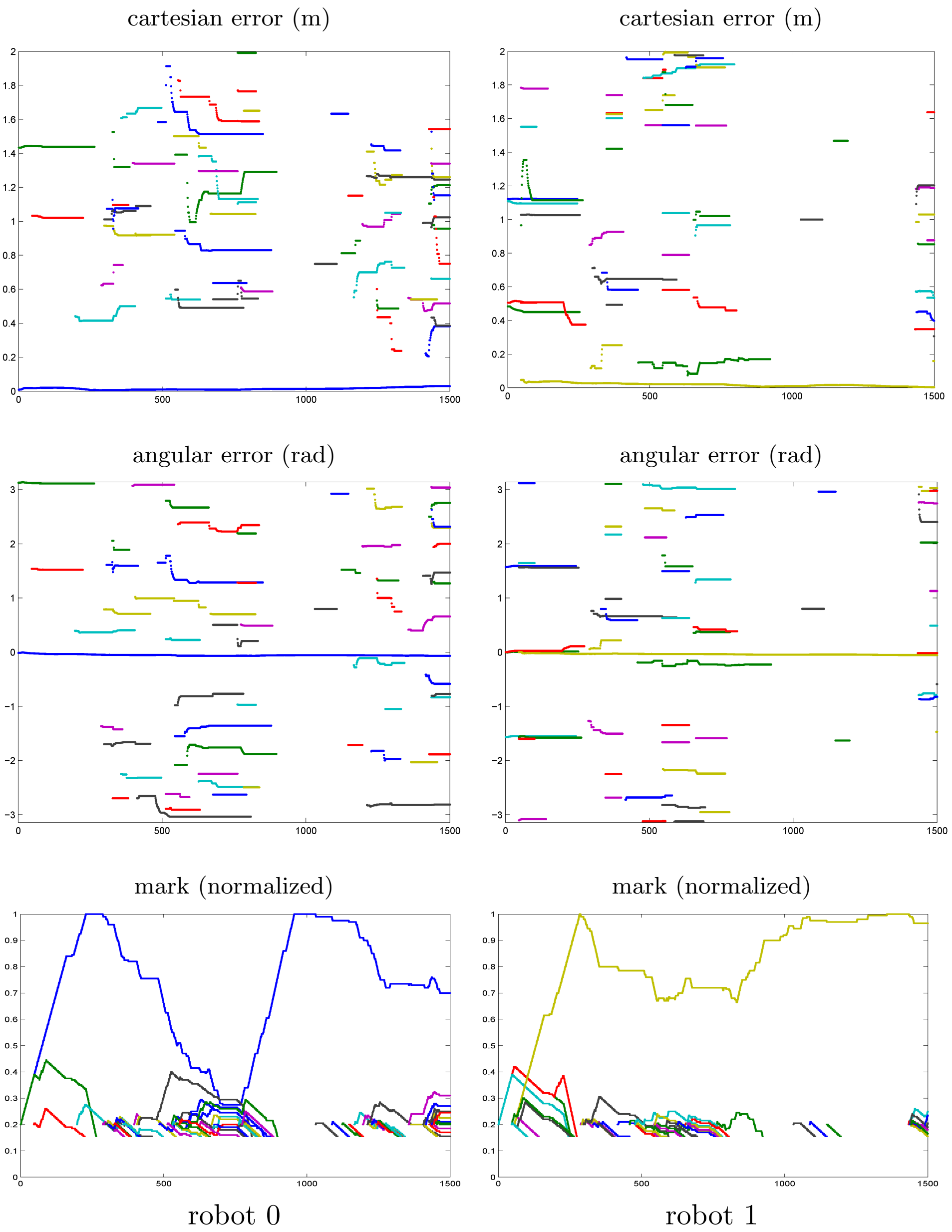


Figure 4: Above: stroboscopic motion of the early step of the first experiment. Below: the best estimate for the same steps (lighter robots indicates older estimates). Small dots are the features generated by the robot detectors.


cartesian error (m)
cartesian error (m)
angular error (rad)
angular error (rad)
mark (normalized)
mark (normalized)
robot 0
robot 1

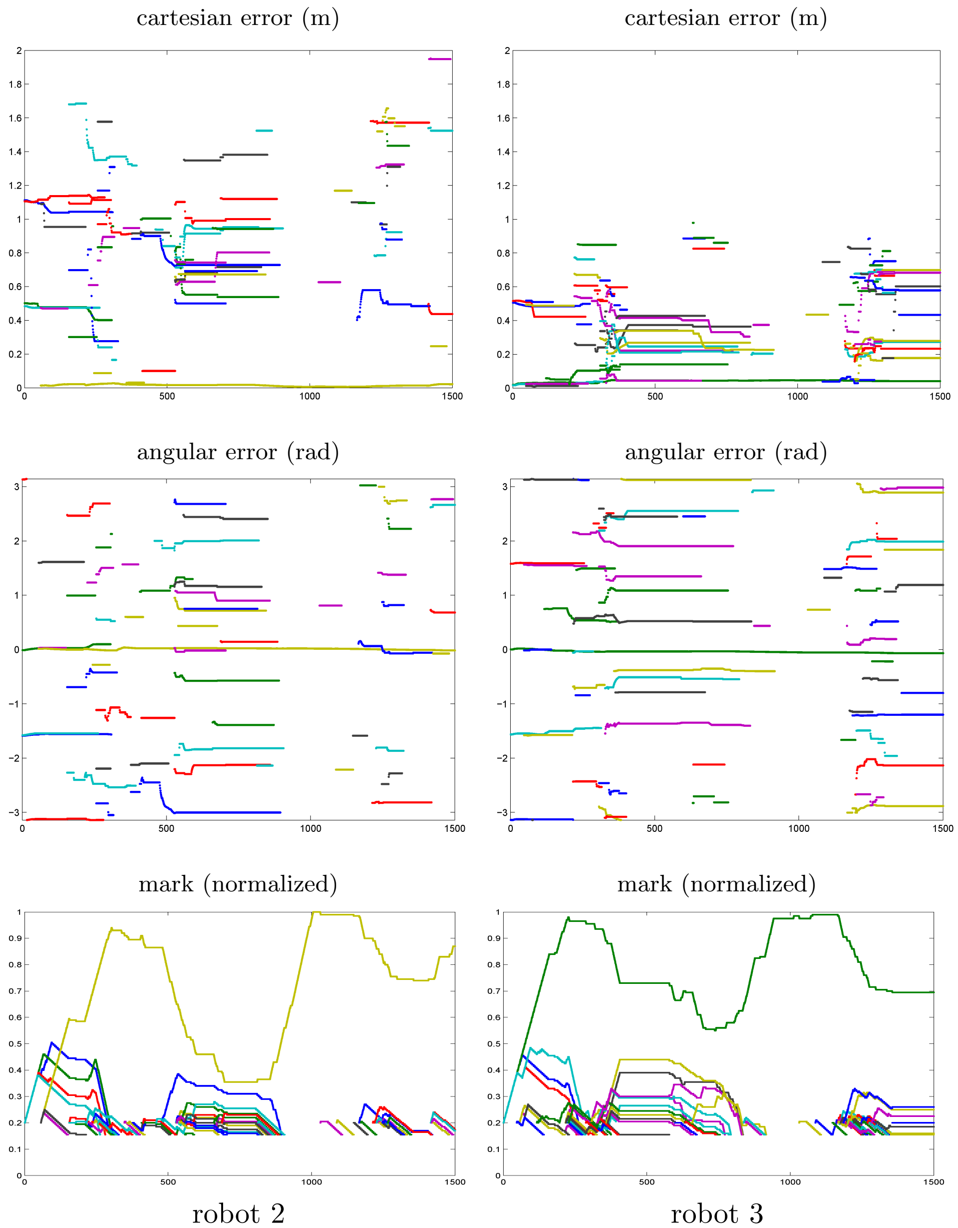


Figure 5: Errors and marks of the estimates generated by the 4 multi-EKFs in the first experiment. Time step is 100 ms.

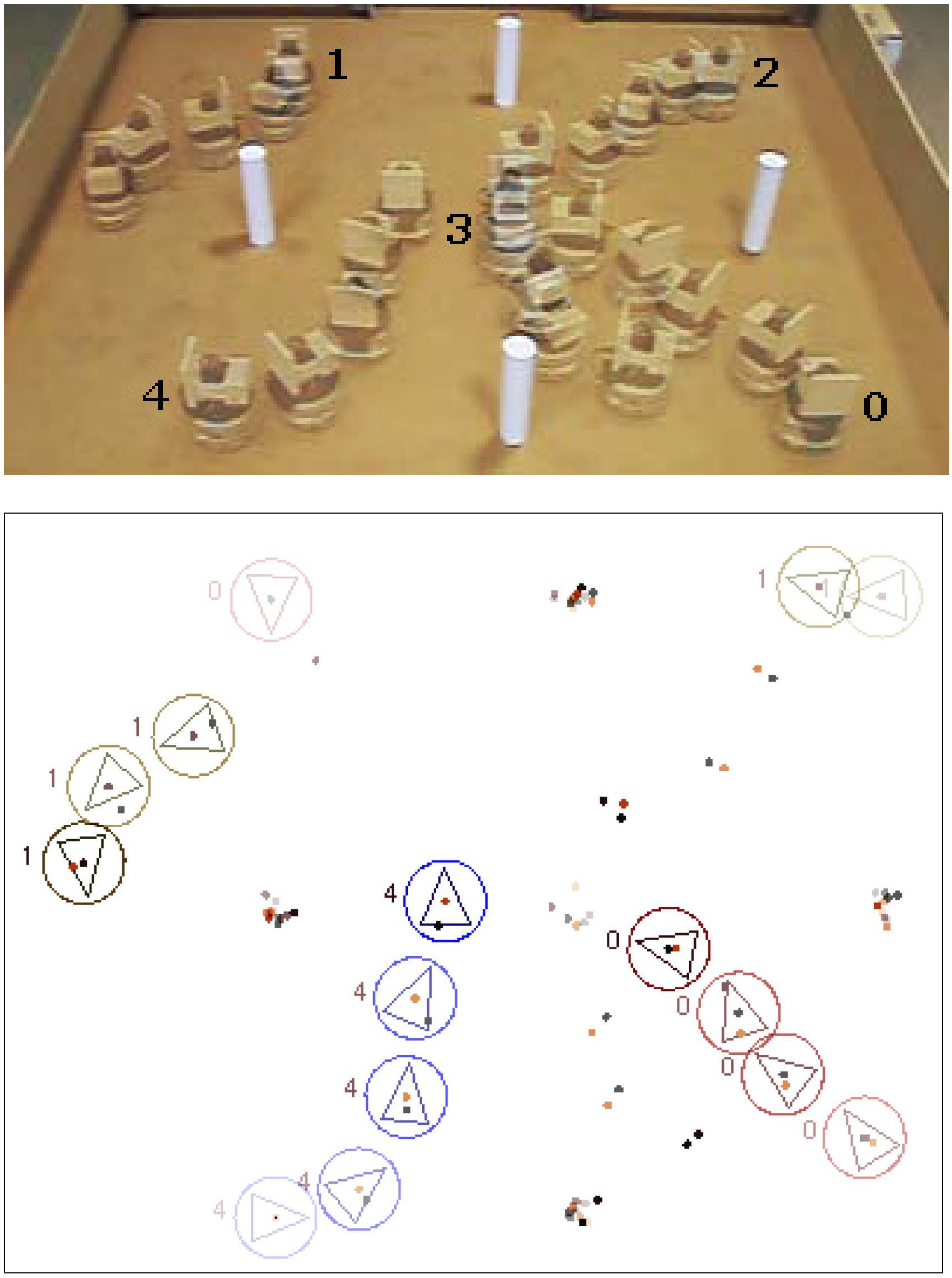


Figure 6: Above: stroboscopic motion of the early step of the second experiment. Below: the best estimate for the same steps (lighter robots indicates older estimates). Robots $\mathcal{A}_2$ and $\mathcal{A}_3$ acts as deceiving movable obstacles.

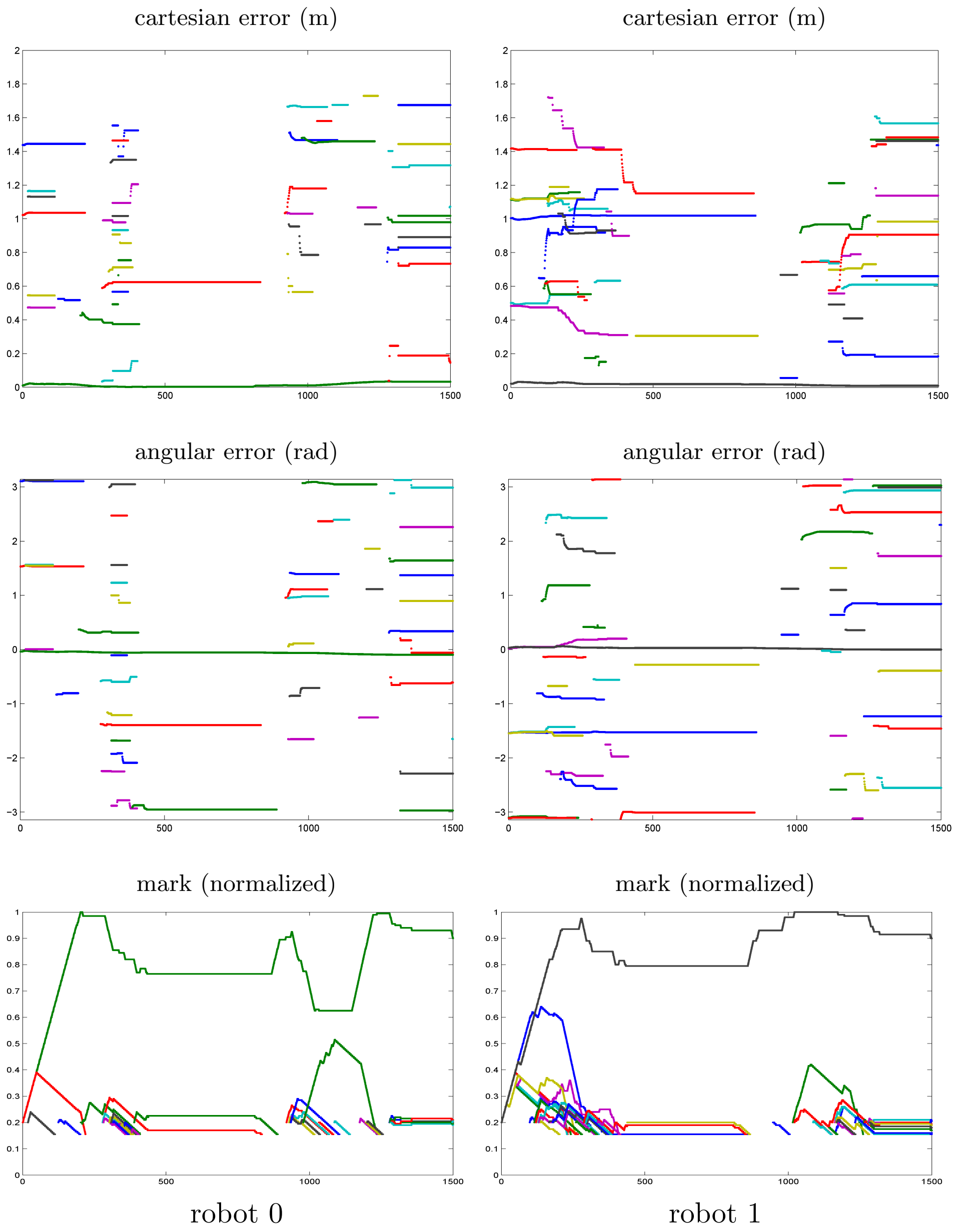


Figure 7: Errors and marks of the estimates generated by the 2 multi-EKFs in the second experiment. Time step is 100 ms.

6.1 Execution times

The MultiReg execution time, which constitutes the major part of the estimation system step time, depends on the number of raw observations in input, say C, and on the ambiguity of the robots arrangement. With unambiguous formations $\frac{C(C-1)}{2}$ binary registrations are needed to reach the solution. Since each binary registration is performed at most in constant time, in this case the MultiReg complexity is $o(n^2)$. In ambiguous formations, more than $(n-1)!$ configurations can be equivalent because of anonymity. So, if a complete algorithm pretend to generate all of them, time $o(n!)$ is needed, unless it is accepted not to generate all of them.

In Fig.**??** are shown the experimental times of MultiReg execution respect to C (left) and to the number of solutions found by itself (right) (that in some way is a quantification of the ambiguousness of the configuration). All times are reported in milliseconds. In particular lower-bound, upper-bound and mean are plotted. All these statistical data are extracted from a pool of 38 experiments of about 4 minutes of duration, which corresponds to 63898 MultiReg calls. In the plot on the left the upper-bound increases exponentially, the lower-bound is quite constant and the mean time has a slightly over-linear increasing rate. Therefore this result matches with the theoretical prediction. In the plot on the right upper-bounds are greater for a small number of solutions, even though the mean has a linear increasing rate. This happens because these cases are more frequent, about 25000 samples against a few dozens in cases with more solutions.

A strategy to reduce execution time could be to generate a maximum number of relative pose hypotheses at each step, relying on the subsequent filter to isolate the right estimate even if the right hypothesis is not generated at each step. In the practice, the assumption of a finite communication range allows to assume an a priori known upper bound for C at each step.

The experiments suggests that the mutual localization module can run easily at 10 Hz on each robot for a team of 5 robots. However, it is not necessary a so high frequency of execution.

7 Conclusions

In this paper, we have presented an innovative system to estimate changes of coordinates between components of a fully decentralized multi-robot system under anonymous measure hypothesis and sensor prone to false positive and false negative measures. The raw data collected from each robot are processed by the MultiReg algorithm to obtain a set of possible relative pose for each robot of the team. The anonymity hypothesis can cause an ambiguity in the inversion of the measure equation, that is solved using a multi-hypotheses filter. Good performance has been obtained both in simulations and in real robot experiments, showing that the proposed localization

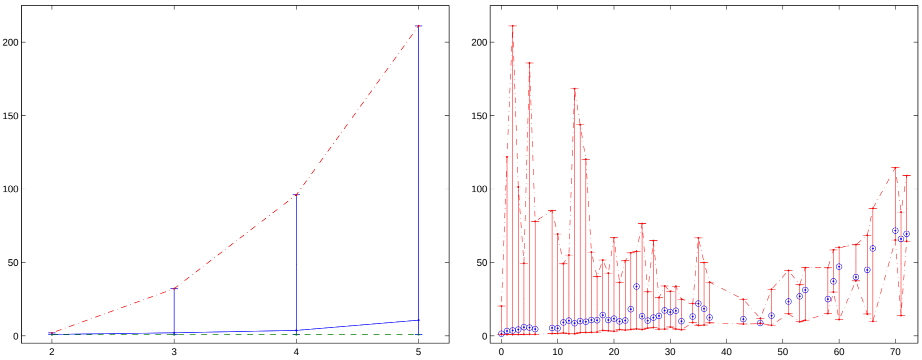


Figure 8: Upper-, mean and lower-bound of the time of execution of MultiReg with respect to the number of raw observations (left) and the number of solution of MultiReg itself (right). Clustering is based on 63898 executions of MultiReg during 38 real robot experiments.

system is applicable in practice.

One problem with the proposed approach is that the execution time of MultiReg may increase considerably if the number of its solutions grows. In future work, we plan to introduce some modifications to improve its performance. For example, a theoretical study of the ambiguity introduced by the anonymity hypothesis can allow to reduce the number of MultiReg solutions, by generating only one representative for each class of equivalent solutions. Another improvement can be obtained by considering only solutions that are close to the estimates generated from the filter, so as to introduce a feedback mechanism from the filters to MultiReg. On the other hand, the 'nearest neighbor' policy of the data association can be avoided by implementing a particle filter that samples also on data association, such as that developed in [**?**]. Another objective is to apply the developed system to a real world task as formation control and cooperative exploration. To this end, a better self-localization module than dead reckoning should be used, for example based on scan matching.

References

[1] S. Roumeliotis and G. Bekey, "Distributed multirobot localization," *Robotics and Automation, IEEE Transactions on*, vol. 18, no. 5, pp. 781–795, Oct 2002.

[2] D. Fox, W. Burgard, H. Kruppa, and S. Thrun, "Collaborative multi-robot localization," in *23rd Annual German Conference on Artificial Intelligence*, 1999, pp. 255–266.

[3] D. Fox, W. Burgard, H. Kruppa, and T. S., "A probabilistic approach to collaborative multi-robot localization," *Autonomous Robots*, vol. 8, no. 3, pp. 325–344, 2000.

[4] A. Howard, M. Mataric, and G. Sukhatme, "Cooperative relative localization for mobile robot teams: an ego-centric approach," in *Naval Research Lab. Workshop on Multi-Robot Systems*, 2003, pp. 65–76.

[5] R. Grabowski, L. Navarro-Serment, C. Paredis, and P. Khosla, "Heterogeneous teams of modular robots for mapping and exploration," *Autonomous Robots*, vol. 8, no. 3, pp. 43–52, 2000.

[6] A. Martinelli, F. Pont, and R. Siegwart, "Multi-robot localization using relative observations," in *2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 2797–2802.

[7] X. Zhou and S. Roumeliotis, "Determining the robot-to-robot relative pose using range-only measurements," in *2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 4025–4031.

[8] T. Eren, P. Belhumeur, and A. Morse, "Closing ranks in vehicle formations based on rigidity," in *41st Conference on Decision and Control*, vol. 3, 2002, pp. 2959–2964.

[9] T. Eren, W. Whiteley, A. Morse, P. Belhumeur, and B. Anderson, "Sensor and network topologies of formations with direction, bearing, and angle information between agents," in *42nd Conference on Decision and Control*, vol. 3, 2003, pp. 3064–3069 Vol.3.

[10] J. Aspnes, T. Eren, D. Goldenberg, A. Morse, W. Whiteley, Y. Yang, B. Anderson, and P. Belhumeur, "A theory of network localization," *Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663–1678, 2006.

[11] X. Zhou and S. Roumeliotis, "Multi-robot slam with unknown initial correspondence: The robot rendezvous case," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[12] H. Kato, K. Ishiguro and M. Barth, "Identifying and localizing robots in a multi-robot system environment," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, 1999, pp. 966–971.

[13] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *CACM*, vol. 24, no. 6, pp. 381–395, 1981.

[14] D. Fontanelli, L. Ricciato, and S. Soatto, "A fast ransac-based registration algorithm for accurate localization in unknown environments using lidar measurements," in *IEEE 2007 International Conference on Automation Science and Engineering*, 2007, pp. 597–602.

[15] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *Pattern Anal. Mach. Intell., IEEE Transaction on*, vol. 13, no. 4, pp. 376–380, 1991.

[16] I. Cox, "A review of statistical data association techniques for motion correspondence," *International Journal of Computer Vision*, vol. 10, no. 1, pp. 53–66, 1993.

[17] A. Franchi, L. Freda, L. Marchionni, G. Oriolo, and M. Vendittelli, "Decentralized cooperative exploration: Implementation and experiments," in *Intelligent Autonomous Systems 10*, 2008, pp. 348 – 355.

[18] M. Montemerlo and S. Thrun, "Simultaneous localization and mapping with unknown data association using fastslam," in *2003 IEEE International Conference on Robotics and Automation*, vol. 2, 2003, pp. 1985–1991.