Towards Cooperative Localization in Robotic Swarms

Anderson G. Pires, Douglas G. Macharet and Luiz Chaimowicz

Abstract Cooperative localization allows groups of robots to improve their overall localization by sharing position estimates within the team. In spite of being a well studied problem, very few works deal with the increased complexity when a large number of robots is used, as is the case in robotic swarms. In this paper, we present a characterization and analysis of the cooperative localization problem for robotic swarms. We use a decentralized cooperative mechanism in which robots take turns as dynamic landmarks providing information to their teammates. We perform several simulations and analyze the influence of these dynamic landmarks in the localization. More specifically, we study the impact of the number of robots in the localization and how the choice of landmarks affects the results.

Keywords Cooperative localization · Cooperative mobile robots · Swarm robotics

1 Introduction

The localization problem is one of the most fundamental in mobile robotics. It generally consists in estimating the robot pose relative to a reference frame in the environment. When robots are equipped with exteroceptive sensors (such as laser range finders) and a set of known landmarks or a map of the environment is available,

A.G. Pires (✉) · D.G. Macharet · L. Chaimowicz
Computer Vision and Robotics Laboratory (VeRLab), Computer Science Department – Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, MG, Brazil
e-mail: anderson@leopoldina.cefetmg.br

D.G. Macharet
e-mail: doug@dcc.ufmg.br

L. Chaimowicz
e-mail: chaimo@dcc.ufmg.br

A.G. Pires
Computer and Mechanics Department – Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG), Leopoldina, MG, Brazil

N.-Y. Chong and Y.-J. Cho (eds.), *Distributed Autonomous Robotic Systems*,
Springer Tracts in Advanced Robotics 112, DOI 10.1007/978-4-431-55879-8_8

localization is relatively simple. This is also true for outdoor robots equipped with a good GPS, which can provide position estimates in a global reference frame. But in more general settings, in which GPS is not available and the robot has no knowledge about the environment, robots have to rely on dead reckoning methods that compute new pose estimates from previous ones. Unfortunately, these methods are subjected to accumulated errors when traveling long distances, which lead to uncertainties that may compromise the quality of the localization.

In multi-robot teams, individual localization estimates can be corrected based on the teammates' positions instead of landmarks in the environment. This is one of the benefits of cooperative robotics, which allows robots to share responsibilities and exchange information to better accomplish tasks. The pose belief adjustment occurs by means of information exchange, which generally happens when the robots meet each other (i.e. there are robots within the communication range). Multi-robot systems employing this technique, commonly denoted Cooperative Localization (CL), have less dependence on the availability of global localization information. Consequently, this kind of system can be used to explore unknown areas or scenarios where global localization is not always available.

These advantages can be leveraged with the use of large groups of robots, which usually present increased flexibility and robustness. Generally called *Robotic Swarms*, these systems employ a large number of simpler agents to perform different types of tasks, acting in a completely decentralized fashion. As will be discussed in the next section, most of the CL methods have focused on the use of a small number of robots since the complexity in terms of coordination and information exchange increases with the number of robots. Thus, the problem of CL in robotic swarms has not been fully explored and presents relevant questions to be investigated.

In this paper, we present a characterization and analysis of the cooperative localization problem considering a swarm of robots. We use a decentralized method, in which the main idea is to have some robots in the swarm acting as dynamic landmarks and providing a localization structure to the group. More specifically, we have the members of the group alternately working as localization providers. These members, acting as beacons, publish localization information in their vicinities to allow their neighbors to adjust their localization beliefs. In this context, we perform an extensive series of simulations and analyze how the number of dynamic landmarks and their choice may impact the localization in a robotic swarm.

The remainder of this paper is structured as follows. A review on the CL literature is presented in Sect. 2. The methodology is presented in Sect. 3, which initially describes the theoretical formalizations (Sect. 3.1), followed by the cooperative swarm localization method (Sect. 3.2), and the swarm motion strategy (Sect. 3.3) used to move the group as a unit. Experiments and statistical analysis are presented in Sect. 4. Finally, Sect. 5 brings the conclusions and directions for future work.

2 Related Work

One of the first works that use robots as landmarks to perform cooperative localization is [6], in which a group of robots, with awareness of its initial localization, is divided in two subgroups with alternating motion and roles. At each time-step, one group is in motion while the other remains static to serve as landmark. After the motion, the robots update their localization beliefs by using relative observations and then remain stationary acting as landmarks to the other group. Despite the good results shown in real applications [5], the need of a centralized entity that controls the actions of all robots and estimates their beliefs compromises the robustness and scalability of the method.

Another seminal work is [7], in which the concept of cooperative localization is employed in a task related to mapping. Two robots are equipped with sensors that allow them to track each other. The coordination mechanism permits them to divide the environment by using spatial decomposition, such that at any single time one robot acting as landmark is positioned in a corner of the environment, while the other spans the perimeter maintaining visual contact. Therefore, the regions of the entire free space are covered and a dual graph is generated, which can be used to posterior exploration of the area.

A more general approach to the CL problem is presented in [10]. The method is based on the generation of a *joint* estimation of the robots' pose in a group, which is computed using an Extended Kalman Filter (EKF) [4]. Both centralized and decentralized methods were applied to generate the joint estimation. In the decentralized approach, each robot performs the prediction step of the filter individually while the update step is performed by exchanging information with others via communication and exteroceptive sensors. The localization interdependence is considered and its representation (cross correlation terms) are stored by all robots and explicitly propagated to the teammates. Using these terms, a robot can estimate its pose by considering the shared knowledge associated with previous meetings. In spite of having the best estimate as a consequence of the use of localization interdependence, this strategy has the disadvantage of requiring a previous knowledge about the group size and presents a complexity that increases quadratically with the number of robots, precluding its use for large groups of robots, such as swarms.

To deal with this drawback, other works have proposed approximate strategies to perform the belief update. These approaches use only part of the group to calculate the robot's estimate. In a recent approximate approach [1], the belief update is performed by using the Covariance Intersection Algorithm (CI) [3], which is a consistent method to fuse estimates of a same quantity with unknown cross-correlation terms. This approach allows each robot to maintain only its own state-covariance estimate with a cost to generate a new estimate of $O(n)$.

Some works have investigated different strategies to increase the quality of localization. For instance, [2, 14, 15] explore the motion mechanism of the group. Tully et al. [14], for example, presents a leap-frog motion technique that has been designed to aid localization for a team of three robots that move alternatively. The results show that this motion strategy outperforms the optimal formation-based path. A method based on leader-following is presented in [16], in which they explore the formation to generate optimal motion strategies. Two robots are employed and the simulation results present better localization accuracy in comparison with the other formation methods used.

However, just a few works have studied the influence of the group size in relation to the quality of the group localization. In [11], a theoretical analysis relates the effect of the number of robots and the error accumulation. In this analysis, the continuous exchange of localization information among the robots is considered and each robot uses sensors of limited accuracy to provide absolute orientation. It is shown that the uncertainty growth is inversely proportional to the number of robots and the rate of growth depends only on this number and the uncertainty of proprioceptive sensors. In [8], this aspect is evaluated together with the type of measurement used. Although the results have shown that the increase in the number of robots contributes to the quality of localization, relevant details have not become explicit, such as the number of robots used as landmarks, which prevents a more detailed analysis. In a recent work [12], the influence of the group size is studied in simulated experiments using groups from two to five robots. A centralized EKF sequentially performs the localization estimate of the group using the data sent by all robots. Because of the restricted scope of the experiments, the results cannot be generalized.

Thus, in spite of the different studies in cooperative localization, its use in robotic swarms is still incipient. In special, the study of scalability issues in this context is a relevant problem that we consider suitable for investigation.

3 Methodology

In this work, we consider a large group of robots (swarm) that moves in a cohesive manner. As in [6, 7], the swarm is divided into two subgroups that move in a mutually exclusive way. One group remains stationary broadcasting their pose information while the robots of the other group move using proprioceptive sensors to estimate their pose. After moving for a certain amount of time, each of these robots updates its belief by using some of the stationary robots in its neighborhood as landmarks. The process is completely decentralized: each robot estimates its distance and orientation to the landmarks, and use the pose information disseminated by them to correct its pose. After this, robots exchange roles: the stationary group starts moving while the robots of the other group become stationary landmarks.

Similarly to [1], we consider that robots are able to identify and measure relative ranges and bearings to their neighbors and exchange information with them. Also, robots are equipped with proprioceptive sensors that allow them to measure their own motion. Since we are using holonomic robots, we do not consider the robot orientation. We assume that all sensor measurements are subjected to white Gaussian noise, but communication is performed without errors.

3.1 Theoretical Formalization

Let us consider the scenario where a swarm $\boldsymbol{\mathcal{R}} = \{\mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_\eta\}$ of η holonomic robots must navigate in a static 2D environment. Let $\mathbf{p}_k^i = [x_k^i \; y_k^i]^\top$ be the vector at time-step k that represents the true position of the ith robot ($\mathcal{R}_i$) in a common global frame W, and $\mathbf{u}_k^i = [vx_k^i \; vy_k^i]^\top$ the vector that represents its control action in the same time-step, in which vx_k^i and vy_k^i stand for the input velocities in x and y directions, respectively.

The state $\mathbf{x}_k^i$ of the robot $\mathcal{R}_i$ at time-step k is equal to its position, *i.e.* $\mathbf{x}_k^i = \mathbf{p}_k^i = [x_k^i \; y_k^i]^\top$, and its discrete-time motion model is expressed by:

$$\begin{aligned} \mathbf{x}_{k+1}^i &= f(\mathbf{x}_k^i, \mathbf{u}_k^i), \qquad\qquad i = 1, \ldots, \eta \\ &= \begin{bmatrix} x_k^i + vx_k^i \Delta k \\ y_k^i + vy_k^i \Delta k \end{bmatrix}. \end{aligned} \tag{1}$$

A neighborhood $\mathcal{N}$ consists of a circular region of radius $\boldsymbol{\tau}$ around the current position of a robot. Thus, we can define $\boldsymbol{\mathcal{N}} = \{\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_\eta\}$ as the set of calculated neighborhoods, all with the same radius. As mentioned, we assume that $\mathcal{R}_i$ can exchange information and measure relative range ρ and bearing ϕ of all robots inside its neighborhood $\mathcal{N}_i$. Moreover, it is assumed that robots inside a neighborhood $\mathcal{N}_i$ can be uniquely identified by the exteroceptive sensor of robot $\mathcal{R}_i$.

The true range and bearing taken by robot i of a robot j at time-step k are respectively denoted by $\rho_k^{i,j}$ and $\phi_k^{i,j}$.[1] Thus, the true range $\rho_k^{i,j}$ and bearing $\phi_k^{i,j}$ taken at time-step k by robot i of robot j, is given by $h(\mathbf{x}_k^i, \mathbf{x}_k^j)$, where

$$h(\mathbf{x}_k^i, \mathbf{x}_k^j) = \begin{bmatrix} \rho_k^{i,j} \\ \phi_k^{i,j} \end{bmatrix} = \begin{bmatrix} \sqrt{(x_k^j - x_k^i)^2 + (y_k^j - y_k^i)^2} \\ \mathrm{atan2}(y_k^j - y_k^i, x_k^j - x_k^i) \end{bmatrix}. \tag{2}$$

The measurement model at time-step $k+1$, when $\mathcal{R}_i$ gets a relative position measurement of $\mathcal{R}_j$, $\mathbf{z}_{k+1}^{i,j} = [\hat{\rho}_{k+1}^{i,j} \; \hat{\phi}_{k+1}^{i,j}]$, $i, j = 1, \ldots, \eta, i \neq j, j \in \mathcal{N}_i$, is given by

$$\mathbf{z}_{k+1}^{i,j} = h(\mathbf{x}_{k+1}^i, \mathbf{x}_{k+1}^j) + \mathbf{n}_{k+1}^{i,j}, \tag{3}$$

where $\mathbf{n}_{k+1}^{i,j}$ is the zero-mean white Gaussian measurement noise with covariance $\mathbf{R}_{k+1}^{i,j}$ added to the true relative measurements given by $h(\mathbf{x}_{k+1}^i, \mathbf{x}_{k+1}^j)$.

[1] We use the notation $*_k^{i,j}$ to express that a certain value associated with robot i was obtained using information and/or measurements from robot j at time-step k.

3.2 Cooperative Swarm Localization

As described earlier, this work uses a swarm $\boldsymbol{\mathcal{R}}$ of η holonomic robots. The swarm is divided in two subgroups, and their motions are coordinated such that the subgroups move as units in a mutually exclusive way (see Sect. 3.3). Individually, each robot i maintains only its own state estimate $\hat{\mathbf{x}}_k^i$ and the respective covariance $\mathbf{P}_k^i$, due to the costs of processing and communication when cooperatively localizing a large group of robots. In this work we do not address the cross-correlation terms [10], and propose an approximate decentralized algorithm for CL.

A robot $\mathcal{R}_j$ acting as landmark continually broadcasts messages with its state and covariance to its neighbors. After its motion, a robot $\mathcal{R}_i$ trying to localize itself, processes the relative range and bearing measurement $\mathbf{z}_{k+1}^{i,j}$ together with the information received from robot $\mathcal{R}_j$. Using these data and its own predicted state $\hat{\mathbf{x}}_{k+1|k}^i$ [2] and covariance $\mathbf{P}_{k+1|k}^i$ estimates, the robot processes the new state $\hat{\mathbf{x}}_{k+1|k+1}^i$ and covariance $\mathbf{P}_{k+1|k+1}^i$ estimates. This procedure is repeated incrementally for each landmark in order to improve the localization estimates. The mathematical details of this procedure is presented as follows.

The discrete-time motion model described in (1) is used to propagate the state of the robot $\mathcal{R}_i$ as:

$$\hat{\mathbf{x}}_{k+1|k}^i = f(\hat{\mathbf{x}}_{k|k}^i, \hat{\mathbf{u}}_k^i), \qquad i = 1, \ldots, \eta. \tag{4}$$

The robot's state is updated according to a linear function f that considers the previous state $\hat{\mathbf{x}}_k^i$ and an input $\hat{\mathbf{u}}_k^i = \mathbf{u}_k^i + \mathbf{w}_k^i = [\widehat{vx}_k^i \; \widehat{vy}_k^i]^\top$, which is basically the commanded velocities $\mathbf{u}_k^i$ augmented with additive zero-mean white Gaussian noise $\mathbf{w}_k^i$, with covariance $\mathbf{Q}_k^i$. During the motion, each robot individually evolves this model with time-steps of length Δk.

Using an EKF [4], the respective covariance propagation for $\mathcal{R}_i$ is given by:

$$\mathbf{P}_{k+1|k}^i = \Phi_k^i \mathbf{P}_{k|k}^i (\Phi_k^i)^\top + \mathbf{G}_k^i \mathbf{Q}_k^i (\mathbf{G}_k^i)^\top, \tag{5}$$

where Φ_k^i is a 2×2 identity matrix ($\mathbf{I}_2$) and $\mathbf{G}_k^i$ is this same matrix multiplied by the time-step Δk.

Thus, when $\mathcal{R}_i$ receives the message with the state and covariance estimates from robot j and obtains a relative measurement $\mathbf{z}_{k+1}^{i,j}$ of it, $\mathcal{R}_i$ can generate an estimate of its state as if such estimate had been calculated by the robot $\mathcal{R}_j$. The following equation illustrates this process:

$$\hat{\mathbf{x}}_{k+1}^{i,j} = \hat{\mathbf{x}}_{k+1|k+1}^j - g(\mathbf{z}_{k+1}^{i,j}), \tag{6}$$

[2] Notation is similar to [1], where $\hat{\mathbf{y}}_{l|m}$ denotes the estimate of the random variable $\mathbf{y}$ at time-step l, given the measurements up to time-step m.

where

$$g(\mathbf{z}_{k+1}^{i,j}) = \begin{bmatrix} \hat{\rho}_{k+1}^{i,j}\cos(\hat{\phi}_{k+1}^{i,j}) \\ \hat{\rho}_{k+1}^{i,j}\sin(\hat{\phi}_{k+1}^{i,j}) \end{bmatrix}.$$

As described in [13], the uncertainty $\mathbf{R}_{k+1}^{i,j}$ tied to the measurement $\mathbf{z}_{k+1}^{i,j}$ can be converted to the common global frame by the means of the jacobian $\mathbf{J}_{k+1}^{i,j}$, as follows:

$$\begin{aligned} \mathbf{J}_{k+1}^{i,j} &= \nabla_{\mathbf{x}_k} g(\mathbf{z}_{k+1}^{i,j})\big|_{\mathbf{x}_k^i=\hat{\mathbf{x}}_{k+1|k}^i,\mathbf{x}_k^{i,j}=\hat{\mathbf{x}}_{k+1}^{i,j}} \\ &= \begin{bmatrix} \cos(\hat{\phi}_{k+1}^{i,j}) & -\hat{\rho}_{k+1}^{i,j}\sin(\hat{\phi}_{k+1}^{i,j}) \\ \sin(\hat{\phi}_{k+1}^{i,j}) & \hat{\rho}_{k+1}^{i,j}\cos(\hat{\phi}_{k+1}^{i,j}) \end{bmatrix}. \end{aligned} \tag{7}$$

The jacobian $\mathbf{J}$ relates the deviation of the original $[\Delta\hat{\rho}\ \Delta\hat{\phi}]^\top$ and the transformed $[\Delta\hat{x}\ \ \Delta\hat{y}]^\top$ variables, which represent the distance from robots i and j in x and y coordinates, respectively. Calculated as:

$$\begin{bmatrix} \Delta\hat{x} \\ \Delta\hat{y} \end{bmatrix} = \mathbf{J}\begin{bmatrix} \Delta\hat{\rho} \\ \Delta\hat{\phi} \end{bmatrix}. \tag{8}$$

The covariance is defined by the expectation of the squared deviates. So, the covariance of the measurement $\mathbf{z}$ in the common frame is defined by multiplying both sides of Eq. (8) by their respective transposes and taking the expectation of the result. This transformation represents an adequate linear approximation when the variables are represented by Gaussians with small variances, as stated in [13]. The uncertainty $\mathbf{P}_{k+1}^{i,j}$ related to the $\hat{\mathbf{x}}_{k+1}^{i,j}$ estimate is generated by the combination of the covariance matrices:

$$\mathbf{P}_{k+1}^{i,j} = \mathbf{P}_{k+1|k}^{i} + \mathbf{J}_{k+1}^{i,j}\mathbf{R}_{k+1}^{i,j}(\mathbf{J}_{k+1}^{i,j})^\top. \tag{9}$$

The estimates $\hat{\mathbf{x}}_{k+1|k}^i$ and $\hat{\mathbf{x}}_{k+1}^{i,j}$ are combined by the EKF update step. This generates a new state $\hat{\mathbf{x}}_{k+1|k+1}^{i,j}$ and covariance $\mathbf{P}_{k+1|k+1}^{i,j}$, which represent the actual belief of the robot. New landmark information and relative measurements are combined with this belief using the procedure described above in an incremental way. The final state $\hat{\mathbf{x}}_{k+1|k+1}^i$ and covariance $\mathbf{P}_{k+1|k+1}^i$ estimate is used for the next motion step.

An important point of the methodology is the choice of the neighbors used as landmarks. We use two different methods: the first one considers the k closest neighbors while the second choses the k neighbors with lowest uncertainties (covariance trace). The performances of these different methods are compared in Sect. 4.

3.3 Swarm Motion Strategy

As mentioned, the swarm $\mathcal{R}$ is randomly divided into two subgroups: one that moves while the other maintains its position. After a pre-specified number of time-steps, the two subgroups exchange roles. Lets call these subgroups $\mathcal{R}_m$ and $\mathcal{R}_s$, for *moving* and *stationary* ones, respectively.

Robots motion is governed by decentralized flocking rules, which allow them to move in a cohesive way, while avoiding collisions. The motion strategy is based on some of the basic rules of Reynolds' flocking algorithm [9]. Each rule establishes a vector that determines a direction to be followed.

The first rule, separation, aims to maintain a safe distance among the robots. The separation behavior is calculated by:

$$\mathbf{v}_i^{sep} = \sum_{j \in \mathcal{R}, j \neq i} \left(\frac{r_s}{\|\mathbf{d}_{ij}\|} - 1 \right) \mathbf{d}_{ij}, \ \ \|\mathbf{d}_{ij}\| \leq r_s, \tag{10}$$

which computes a separation vector $\mathbf{v}^{sep}$ based on the displacement $\mathbf{d}_{ij}$ between a robot and its teammates located inside a specific separation range (r_s).

The second rule aims to maintain the robots together acting as a unit. This cohesion rule computes the average position of the k moving robots that are located inside a cohesion range (r_c), and generates a vector $\mathbf{v}_i^{coh}$ pointing in this direction as:

$$\mathbf{v}_i^{coh} = \frac{1}{k} \sum_{j \in \mathcal{R}_m, j \neq i} \mathbf{d}_{ij}, \ \ \|\mathbf{d}_{ij}\| \leq r_c. \tag{11}$$

Finally, we consider that the robots have a series of common targets to be reached during their motion. These targets are used to compute a direction vector ($\mathbf{v}^{dir}$) to be followed. The displacement vector between a robot i and its next target t is represented by $\mathbf{d}_{it}$. As shown in Eq. (12), the direction vector consists of the unit vector in the direction of $\mathbf{d}_{it}$:

$$\mathbf{v}_i^{dir} = \frac{\mathbf{d}_{it}}{\|\mathbf{d}_{it}\|} \ . \tag{12}$$

The control action $\mathbf{u}_i$ is given by:

$$\mathbf{u}_i = k_c \ \mathbf{v}_i^{coh} + k_s \ \mathbf{v}_i^{sep} + k_d \ \mathbf{v}_i^{dir} \ , \tag{13}$$

which is composed by the weighted sum of the three vectors. The control action is then decomposed in velocities vx_i and vy_i, that will be used by robot i.

We assume all robots have a synchronized clock, and the decision to switch robots from $\mathcal{R}_s$ to $\mathcal{R}_m$ (i.e. from stationary landmarks to the moving robots group), and vice versa, is made periodically on a completely decentralized manner. At first, the swarm is divided in two predefined subgroups, and a common timer is initialized.

As soon as the timer reaches a certain value, the robots change their roles, and the timer is reinitialized. This loop is repeated until the mission is fulfilled.

3.4 Complexity Analysis

In this section, we present a brief analysis of the computational complexity of each stage of the proposed methodology, as well as bounds regarding the number of messages used by each robot.

In most systems dealing with the CL problem, the cost to estimate the position is usually $O(n^2)$, where n is the number of robots, since $n(n-1)$ measurements are needed to calculate a new estimation.

In this work, the swarm is divided into two subgroups, and all the robots on a subgroup should measure the relative range and bearing of all robots from the other subgroup. Therefore, it takes $(n/2)(n/2)$ measurements, still leading to an $O(n^2)$ complexity, where n is the number of robots. However, as will be shown in Sect. 4, the use of just part of the robots from the other group improves significantly the position estimation, which takes $k(n/2)$ measurements. Therefore, we have an $O(kn)$ complexity, since we consider $k \ll n/2$ as the number of robots that will be used as landmarks, and n is the total number of robots.

4 Experiments

Several simulations were performed to analyze how the number of dynamic landmarks and the way they are chosen impact the localization. We have also varied parameters related to the quality of the information in order to study its influence on the localization error and the number of necessary landmarks.

The experiments presented here were executed considering a swarm with 30 holonomic robots. The group navigates in an obstacle-free static environment of approximately one hundred square meters (10 m $\times$ 10 m). Robot motion is directed by a series of waypoints, which define targets to be reached by the group. These targets dictates the preferred direction ($\mathbf{v}_i^{dir}$) for each robot and it is used in computing the commanded velocity $\mathbf{u}_i$. These velocities are limited to 0.1 m/s, and are subjected to additive zero-mean Gaussian noise of 10 % of the true velocity. Each simulation takes approximately 6500 time-steps, where $\Delta k = 0.1$ s is the duration of each time-step, and the total length of the traveled distance is ≈30 m.

Following the methodology, the robots are initially assembled together and divided into two subgroups at random. The motion of the groups is coordinated so that they move in turns for a specific number of time-steps. One subgroup is selected to start moving while the other remains stationary. After motion, the group stops and each member computes range and bearing to some of the static robots located inside their

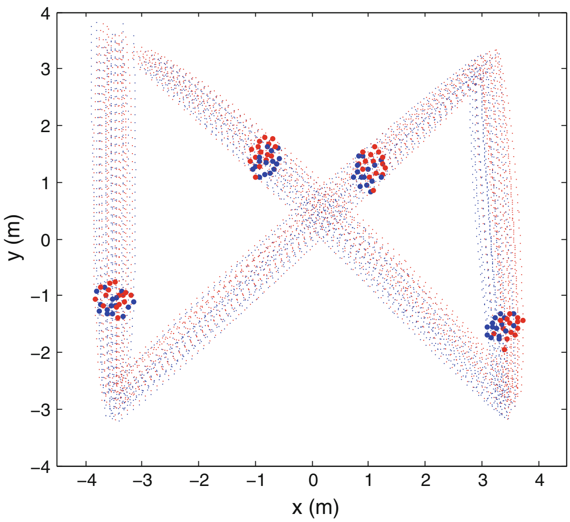

Fig. 1 Example of the path executed by the subgroups (*red* and *blue* colors)

neighborhoods. In the simulations, the radius of the neighborhoods was made large enough to allow the analysis of the impact of varying the number of landmarks.

Figure 1 presents an example of a path executed by the swarm. The position of the members of both groups (depicted in red and blue) is presented over time. The small circles represent the position of each robot in four distinct moments in which the blue group performed cooperative localization.

The first experiment was performed to evaluate the impact of different error parameters in the quality of the final localization. The range noise was defined proportional to the measurements while the bearing noise has been fixed to some absolute values. The noise **n** of the measurement model (Eq. 3) was defined using the following values $\sigma_\rho = \{0.05, 0.10, 0.20, 0.30, 0.40\} \times \rho$ and $\sigma_\phi = \{1°, 2°, 3°\}$. These parameters were analyzed varying the number of landmark robots used for correcting the localization estimates, ranging from 1 robot to $\frac{\eta}{2}$ robots (i.e. the entire stationary group). For each specified combination of noise values $(\sigma_\rho, \sigma_\phi)$ and number of landmarks, a total of 30 simulations were executed. The position mean error after the path has been completed is presented in Fig. 2.

For each simulation, a mean localization error is calculated by taking the average of the root mean square error (RMSE) value of the position errors for all robots during their motion, assuming they are using a determined number of neighbors as landmarks. It is possible to observe that the bearing noise has a small influence in the mean error. However, with the increase in the noise of the relative range, a robot needs to use more landmarks in order to better correct its position estimate. We can also observe that in situations where the noise is small, the increase in the number of landmarks over certain values does not contribute to reduce the localization error. For example, for $\sigma_\rho = 0.05\rho$, the error decreases very slowly for more than 6

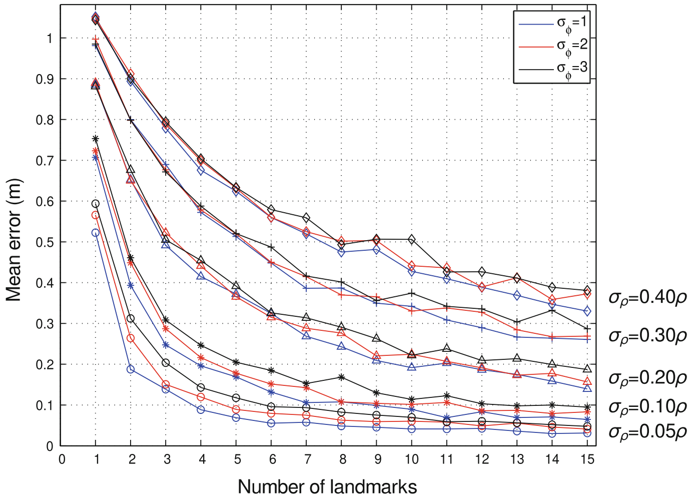

Fig. 2 Mean position error varying the number of landmarks used for correction

landmarks. In other words, depending on the observation noise, robots can improve their localization using just a small number of landmarks in their neighborhood.

We also analyzed the increase on the RMSE along the execution when a different number of landmarks is used. Figure 3 shows the results of a set of 30 simulations in which the associated noises were fixed at $\sigma_\rho = 0.10\rho$ and $\sigma_\phi = 3°$. It is possible to observe that the error is largely reduced when at least one landmark is used, but this reduction is smaller for more than four landmarks. This reinforces the analysis that a small number of robots may suffice for improving localization in swarm navigation.

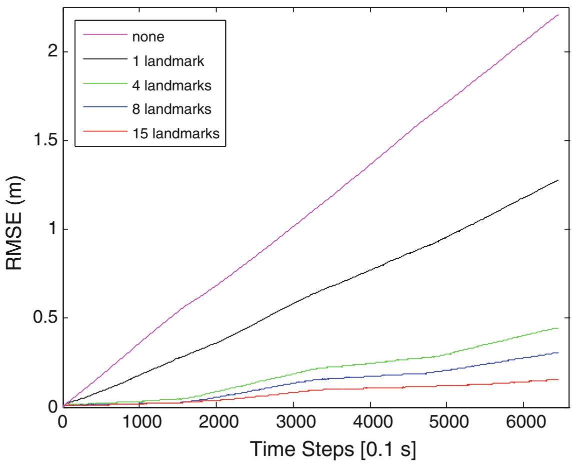

Fig. 3 RMSE accordingly to the different number of landmarks used for localization

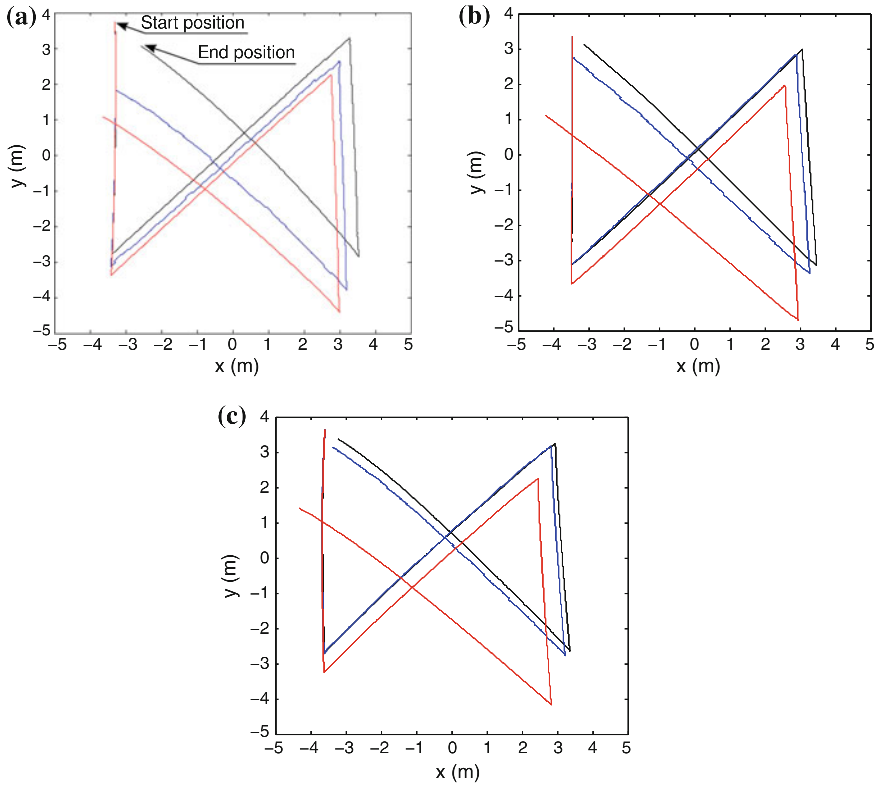

Fig. 4 The real (*black line*) and cooperative estimated (*blue line*) path performed by a robot. The *red line* is the estimate without cooperative localization. **a** 1 landmark. **b** 4 landmarks. **c** 8 landmarks

As can be seen, the RMSE considering the use of 15 landmarks presents a tendency to converge to a constant growing rate after a certain number of time steps. This is a particularly important result, since it provides a notion that it will be possible to obtain a bounded maximum error for the entire swarm.

The improvement on the localization can readily be observed in Fig. 4, which shows the path followed by one of the robots using different numbers of landmarks to correct its localization. In this case, the black line is the ground truth (actual path), the red line represents the robot localization estimates without correction and the blue line represents the localization using other robots as landmarks. Both Figs. 3 and 4 are related to the same set of experiments.

In the previous experiments, the landmarks to be used were chosen at random, without any criterion of selection. The measurements were used in the update phase of the EKF according to the order the messages arrived. With the aim of analyzing how a different ordering could impact the localization, we defined two criteria for selecting the landmarks to be used, and compared them with the random selection. The first considers an ordering process in which the landmarks that are closer to the robot are chosen. The second sorts the landmarks considering their uncertainties, so that

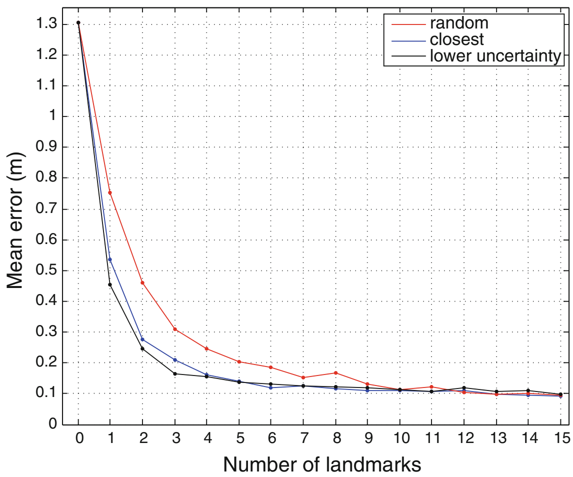

Fig. 5 Mean position error accordingly to the criterion used to select the landmarks

the ones with the lower uncertainty are chosen. The observation noise was set with the same value of the previous experiment. Figure 5 shows that both criteria improve the localization over choosing landmarks randomly. Between the two, the selection of the landmark robots with lower uncertainties is slightly better than choosing the closest landmark robots, specially when few landmarks are used.

5 Coclusion and Future Work

In this paper we performed a characterization of Cooperative Localization for swarms of robots by using an approximate decentralized algorithm. Considering the sensory and computational limitations of this kind of system, we have explored the coordinated motion of the group, so that the robots could cooperatively localize themselves using local information, i.e. using only part of the group as landmarks. We performed a series of simulations in order to evaluate the method and results showed that localization estimates can be significantly improved even using a small number of neighbors as landmarks. Experiments performed with a larger number of robots (not showed here) confirmed this trend. This suggests that the method scales well and can be used when it is necessary to cooperatively localize large groups of robots.

In general, cooperative localization methods have not been used with large group of robots due to the complexity related to the exchange of localization information. Therefore, one of the main contributions of this work is the investigation of such methods in robotic swarms. Although we have not considered the localization interdependence, our results indicate that it is possible to reach good localization even with a few landmarks, which is important in terms of scalability.

The main limitation of this work is related to the fact that we do not incorporate the orientation noise and localization interdependence. In order to address this, we will direct our future work to extend the proposed methodology to deal with non-holonomic robots and also with the localization interdependence. Specially to tackle the latter, we intend Covariance Intersection Algorithm (CI) [1, 3]. We believe that this will make the methodology more general and robust.

Acknowledgments This work was developed with the support of CEFET-MG, CAPES, FAPEMIG and CNPq.

References

1. Carrillo-Arce, L., Nerurkar, E., Gordillo, J., Roumeliotis, S.: Decentralized multi-robot cooperative localization using covariance intersection. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1412–1417 (2013)
2. Hidaka, Y., Mourikis, A., Roumeliotis, S.: Optimal formations for cooperative localization of mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation, pp. 4126–4131. Barcelona, Spain (2005)
3. Julier, S.J., Uhlmann, J.K.: A non-divergent estimation algorithm in the presence of unknown correlations. In: Proceedings of the American Control Conference, vol. 4, pp. 2369–2373 (1997)
4. Kalman, R.E.: A new approach to linear filtering and prediction problems. Trans. ASME—J. Basic Eng. **82**(Series D), 35–45 (1960)
5. Kurazume, R., Hirose, S.: An experimental study of a cooperative positioning system. Auton. Robots **8**(1), 43–52 (2000)
6. Kurazume, R., Nagata, S., Hirose, S.: Cooperative positioning with multiple robots. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1250–1257 (1994)
7. Rekleitis, I.M., Dudek, G., Milios, E.E.: On multiagent exploration. In: Proceedings of Vision Interface, pp. 455–461 (1998)
8. Rekleitis, I.M., Dudek, G., Milios, E.E.: Multi-Robot cooperative localization: a study of trade-offs between efficiency and accuracy. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 3, pp. 2690–2695 (2002)
9. Reynolds, C.W.: Flocks, herds, and schools: a distributed behavioral model. In: Computer Graphics, pp. 25–34 (1987)
10. Roumeliotis, S.I., Bekey, G.A.: Collective Localization: a distributed Kalman filter approach to localization of groups of mobile robots. In: Proceedings of the IEEE International Conference on Robotics and Automation, vol. 3, pp. 2958–2965 (2000)
11. Roumeliotis, S.I., Rekleitis, I.M.: Propagation of uncertainty in cooperative multirobot localization: analysis and experimental results. Auton. Robots **17**, 41–54 (2004)
12. Schneider, F.E., Wildermuth, D.: Influences of the robot group size on cooperative multi-robot localisation—analysis and experimental validation. Robot. Auton. Syst. **60**(11), 1421–1428 (2012)
13. Smith, R.C., Cheeseman, P.: On the representation and estimation of spatial uncertainty. Int. J. Robot. Res. **4**, 56–68 (1986)
14. Tully, S., Kantor, G., Choset, H.: Leap-frog path design for multi-robot cooperative localization. In: Howard, A., Iagnemma, K., Kelly, A. (eds.) Field and Service Robotics, Springer Tracts in Advanced Robotics, vol. 62, pp. 307–317. Springer, Berlin (2010)
15. Zhang, F., Grocholsky, B., Kumar, V.: Formations for localization of robot networks. In: Proceedings of the IEEE International Conference on Robotics and Automation, vol. 4, pp. 3369–3374. New Orleans, LA, USA (2004)
16. Zhou, X.S., Zhou, K.X., Roumeliotis, S.I.: Optimized motion strategies for localization in leader-follower formations. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 98–105. San Francisco, CA, USA (2011)