



HAL
open science

Experimental and Simulation Platforms for Anonymous Robots Self-Localization

Olivier Poulet, Francois Guérin, Frédéric Guinand

► **To cite this version:**

Olivier Poulet, Francois Guérin, Frédéric Guinand. Experimental and Simulation Platforms for Anonymous Robots Self-Localization. 29th Mediterranean Conference on Control and Automation (MED 2021), Jun 2021, Bari (virtuel), Italy. pp.949-954, 10.1109/MED51440.2021.9480244 . hal-03352615

HAL Id: hal-03352615

<https://hal.science/hal-03352615>

Submitted on 23 Sep 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Experimental and Simulation Platforms for Anonymous Robots Self-Localization

Olivier Poulet¹ and François Guérin² and Frédéric Guinand^{1,3}

Abstract— This paper presents an experimental platform and a simulation-based one for the implementation of a method enabling mobile anonymous robots self-localization. The proposed method, theoretically validated in a previous work, is based on the comparison between global information obtained through periodical aerial pictures and local information stemmed from odometry. The process analyzes robots' coordinates evolution between two consecutive pictures and compares these changes with odometric measures. On each picture, robots are anonymous and their identification is impossible without extra information. In this work, measures obtained with actual robots on the experimental platform are compared with the ones obtained on the simulation-based platform reproducing experimental conditions. We show that success rates, defined as the percentage of time robots successfully localize themselves, obtained on both platforms are qualitatively similar while quantitatively different in regard with algorithm performances. Sources responsible for this gap are identified and analyzed. It leads to the conclusion that, in the context of our study, using a simulation-based platform is a valid alternative to actual robots experiments.

I. INTRODUCTION

Global Navigation Satellite Systems (GNSS) enable global positioning of one or several robots based on trilateration. They can also be used for correcting drifts during robots' movements [1]. Indoor, such systems can no longer be used and, when initial position is unknown, sensor fusion approaches propose interesting alternative solutions for localizing robots [11]. Position tracking principle consists in recording odometric measures during robot's movement in order to determine its current position (a.k.a. dead reckoning) [20]. However, this approach is prone to errors and several methods can be used for minimizing them, associating an Extended Kalman Filter to a distance sensor [18], or performing a Monte-Carlo localization (a.k.a. particle filter localization) [19]. There also exist other localization methods based on the knowledge of coordinates of beacons [7] or based on vision [16].

When several robots are considered simultaneously, most works deal with cooperative localization. The accuracy of one robot's position depends on the knowledge of peer's position. In [12], the authors consider a group of robots that uses another group of robots as beacons for updating their coordinates. In order to improve calculation time and accuracy, probabilistic methods have been deployed [5], as

well as 2D trilateration as in [14]. In every case, localization information is updated in each robot's reference frame. Much less works have been dedicated to the case of anonymous robots. Some of these works deal with multi-robots control [17] and more recent advanced works were focusing on mutual localization of anonymous robots [6]. Anonymous robots do not have any proper tag or identifier that could be used for any computation and they all execute the same algorithm [4]. They are not able to distinguish each other [3].

Our work focuses on global localization of anonymous robots. However, the proposed method could also be deployed on autonomous vehicles [13], for instance within a car park. This work brings a solution for silencing localization for a group of ground robots. A high altitude drone periodically takes pictures of an area enabling ground robots to localize themselves within the group without the need of exchanging information with each other. In addition to anonymity, robots cannot communicate with each other, since exteroceptive sensors (infrared) are only used for avoiding obstacles. For the experiments, a camera was fixed on the ceiling of the room and defines the origin of the global reference frame. Periodically, it takes pictures of the floor on which robots are moving and a computer determines the non indexed coordinates of all robots within each picture. Each robot then receives the whole set of non indexed coordinates and aims at identifying its own coordinates among all the ones received, in order to determine its position. This algorithm, based on the fusion of odometric data with the two sets of coordinates stemmed from two consecutive pictures, was theoretically validated and fully described in [15]. It was, to the best of our knowledge, the first proposed method for enabling robots to self-localize using these sources of data. However, in order to enhance the method, new versions of the algorithm should be tested and the number of considered robots should also be increased. But implementing new ideas on an experimental platform with actual robots is time-consuming and requires additional financial and human resources. One alternative is simulation.

The main objective of this work was first to validate experimentally the method on a platform operating actual robots and then to verify if simulation can produce confident results with respect to real experiments. For that purpose we measure, on the experimental platform, the success rate of the method as well as its sensitivity to some parameters. A success is obtained when robots successfully localize themselves. By reproducing some experiments on the simulation platform, we show that similar results are obtained

¹Normandie Univ. UNIHAVRE, LITIS Laboratory, 76600 Le Havre (France) pouletolivier@yahoo.fr, frederic.guinand@univ-lehavre.fr

²Normandie Univ. UNIHAVRE, GREAH Laboratory, 76600 Le Havre (France) francois.guerin@univ-lehavre.fr

³Cardinal Stefan Wyszyński University (UKSW), Warsaw (Poland)

qualitatively. However differences in performances are observed. A careful identification and analysis of the sources of inaccuracy of measures, for the experimental platform using actual robots, leads us to consider the simulation platform reliable for driving new experiments. For illustrating that we've tested a new scenario using four mobile robots and two lures and show that results are in line with what was expected.

Section II is devoted to problem formulation and to the description of the experimental apparatus. The experimental validity of the method together with the measure of the impact of changes on algorithm's performances are investigated in Section III. The four robots scenario is analyzed in Section IV, and measures accuracy is discussed in Section V, followed by a short conclusion and some open perspectives.

II. PROBLEM FORMULATION AND EXPERIMENTAL APPARATUS

A. Problem formulation

The system is composed of a group of N mobile robots among which M are not moving. All robots are located indoor and moving robots evolve within a fixed area. A camera, fixed at the ceiling of the room, takes pictures of the whole area every T units of time. We denote λ the distance ratio between the focal length of the camera and the distance measured on the ground (a scale factor). Each robot receives at regular time interval the coordinates of all the robots: $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$. Between two emissions, the coordinates (x_i, y_i) may not refer to the same robot, thus, upon reception, each robot processes coordinates disregarding the order. Between two receptions, every δt units of time, each robot records its own linear speed v and rotational speed r . In [15], it has been shown that it was possible to determine the coordinates (x_f, y_f) of a robot at a final position f , from the knowledge of its starting coordinates (x_s, y_s) (at position s), its initial angle θ_s and the information related to v and r , periodically recorded during the movement between the two positions.

$$\begin{pmatrix} x_f \\ y_f \end{pmatrix} = \begin{pmatrix} x_s \\ y_s \end{pmatrix} + \delta t \sum_{j=s+1}^f \begin{pmatrix} \lambda \cos(\theta_s + \delta t \sum_{k=s+1}^j r_k) \\ \lambda \sin(\theta_s + \delta t \sum_{k=s+1}^j r_k) \end{pmatrix} \cdot v_j \quad (1)$$

From equation 1, if t_0 corresponds to the first picture, the equation in the temporal domain becomes:

$$\begin{pmatrix} x(t_0 + T) \\ y(t_0 + T) \end{pmatrix} = \begin{pmatrix} x(t_0) \\ y(t_0) \end{pmatrix} + \lambda \cdot \begin{pmatrix} \int_{t_0+\delta t}^T \cos(\theta(t_0) + \int_{t_0+\delta t}^T r(t)dt) \cdot v(t)dt \\ \int_{t_0+\delta t}^T \sin(\theta(t_0) + \int_{t_0+\delta t}^T r(t)dt) \cdot v(t)dt \end{pmatrix} \quad (2)$$

From Equation 2 it is possible to estimate the minimum value of δt .

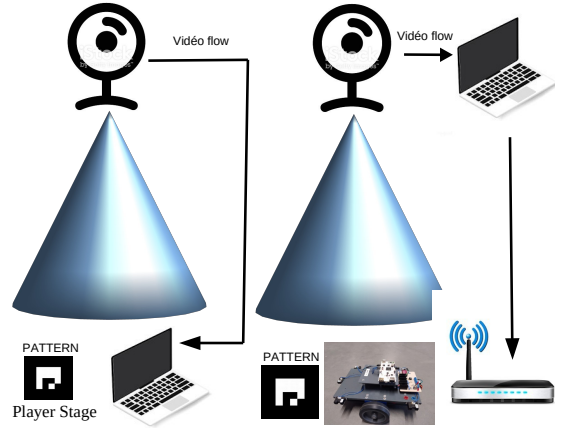


Fig. 1. Left: Simulation platform. Right: Actual Experimental platform

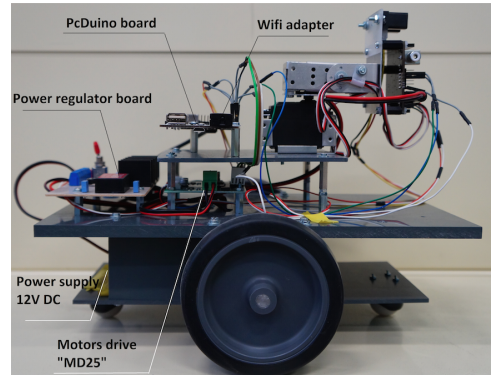


Fig. 2. The mobile robot used for the experiments.

B. Experimental Apparatus and Simulation Environment

Two platforms have been used for comparing actual data with simulated ones (Figure 1). The experimental platform is composed of two robots, a camera, a computer for processing images from the camera and for communicating information to robots and of a router for wireless network. The robots (Figure 2), of size: $l=180 \times l=290 \times H=150$ mm, were built in our lab. They have two driving wheels (diameter: 97mm) animated by two DC motors. Quadrature encoders provides odometric measures (linear and angular speeds and positions used with an inverse kinematic model). Each robot is also equipped with six infrared distance sensors for obstacle avoidance, a wifi dongle, and all processing tasks are performed by the companion computer, a PcDuino board (1GHz ARM Cortex A7 Dual Core CPU). The camera is directly connected to the computer which is connected to the robots through the access point. Robots are thus connected to the access point, but they do not communicate with each other, they only receive information from the computer for taking their movement decisions.

Within the pictures, obtained from the fixed camera and analyzed by the computer, robots are detected thanks to

patterns fixed on top of them. The computer extracts robots coordinates within the images using Artoolkit (as it was done in [8]). Then coordinates are sent to robots and each of them determines its position.

The simulation platform is composed of the same camera and the same computer running Player/Stage. The computer simulates the robots, identified by patterns, and their movements and enable the visualization of the scene on a screen. The camera periodically takes pictures of the screen, and, as in the experimental platform, Artoolkit is used for extracting robots's coordinates which are sent to the robots. The simulated area and the actual one are identical as illustrated on Figure 3.

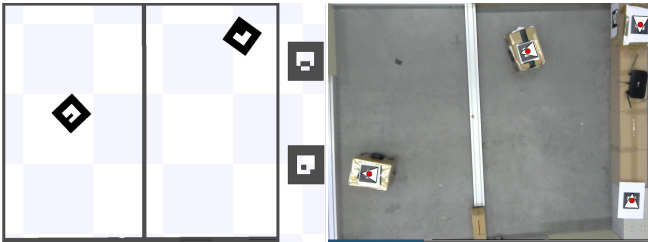


Fig. 3. Area where actual (on the right) and simulated (on the left) robots are moving ($N = 4, M = 2$).

Actual and simulated robots execute the same control algorithm, described by Algorithm 1. Without obstacles, robots move linearly. However for challenging the method, if after K iterations no obstacle is detected, the robot changes its direction by turning slightly on the right. When an obstacle is detected, the robot turns in place ($v = 0$ and $r \neq 0$) in the opposite direction of the obstacle. Finally every δt period of time, both values of v and r are recorded.

The values considered for the experiments were:

v_{min}	v_{max}	r_{min}	r_{max}	δr_{min}	δr_{max}	δt	K
0.1	0.2	0.2	0.4	0	0.05	100	30
m.s ⁻¹		rad.s ⁻¹				ms	

III. IMPACT OF PARAMETERS CHANGES

Every experiment lasts 20 ± 1 minutes.

A. Variation of T

T denotes the sampling period between two consecutive pictures. For this first set of experiments, we study the impact of the variation of T between 2 and 9 seconds. We consider 4 robots, among which only two are moving (illustrated on Figure 3).

For the experiments with actual robots (represented in Figure 4), the self-localization algorithm executed by each robot takes into account the odometric measures. These measures are prone to error and are often inaccurate. For the smaller values of T , values of v are often equal to zero while recorded odometry measures show a linear shift in the robot position. For larger values of T , these values are not null but odometric measures accuracy decreases as T increases. These differences are related to robot's imperfections as it

Algorithm 1 Control algorithm executed by every robot

```

1 . iterationNumber  $\leftarrow$  0
2 . lastTime  $\leftarrow$  getTime()
3 . while (mission not finished) do
4 .   if (obstacle detected) then
5 .     state  $\leftarrow$  obstacle
6 .   else
7 .     state  $\leftarrow$  free
8 .   endIf
9 .   if (state = free) then
10 .    if (iterationNumber = 0) then
11 .      choose  $v \in [v_{min}, v_{max}]$ 
12 .       $r \leftarrow$  0
13 .    else if (iterationNumber = K) then
14 .      choose  $\delta r \in [\delta r_{min}, \delta r_{max}]$ 
15 .       $r \leftarrow r + \delta r$ 
16 .      iterationNumber  $\leftarrow$  0
17 .    endIf
18 .  else
19 .     $v = 0$ 
20 .    compute  $r$  (opposite direction to the obstacle)
21 .    iterationNumber  $\leftarrow$  0
22 .  endIf
23 .  iterationNumber  $\leftarrow$  iterationNumber+1
24 .  move according to  $(v, r)$ 
25 .  currentTime  $\leftarrow$  getTime()
26 .  if (currentTime-lastTime  $\geq$   $\delta t$ ) then
27 .    record both  $v$  and  $r$  values
28 .    lastTime  $\leftarrow$  currentTime
29 .  endIf
30 . endWhile

```

will be discussed in Section V and explains the shape of the graphics (top of Figure 4) where the best success rate is obtained for $T = 6s$.

In order to compare experimental and simulation measures, a second series of measures has been conducted. When, between two consecutive pictures, the linear distance performed by the robot is smaller than a given threshold, corresponding measures are not considered for the computation. This allow to not take into account odometric measures of pure-rotation movements. The corresponding results are reported on Figure 5. The comparison of Figures 4 and 5 highlights that removing non relevant odometric measures (when linear speed is null) results in a better self-localization of robots. Not surprisingly, in both cases, when the period T increases, the performances decrease.

As showed on Figure 6, the same tendency is observed in the simulation context. Instead of odometry, during the simulation the algorithm uses robots' commands. Inaccurate and imprecise values are minimized in simulation which explains the better success rates obtained. On average algorithm reaches success rates 10% better in simulation than for experiments with actual robots. However, when period T increases up to 15 seconds, small errors, alignment of the camera with

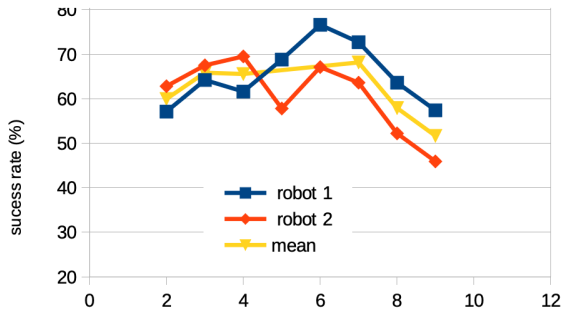


Fig. 4. Measures for actual experiments: success rate with respect to variation of T between 2 seconds and 9 seconds.

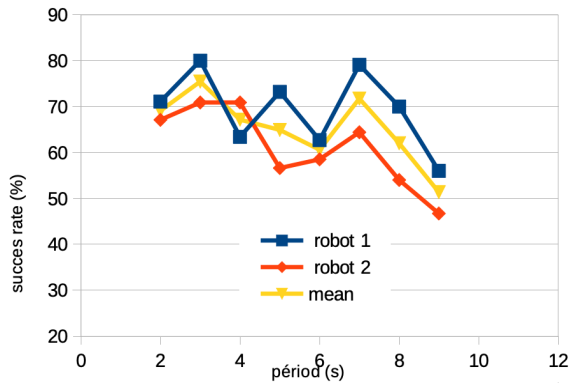


Fig. 5. Second series of experiments.

the screen, imprecise λ value, are magnified, which leads to the degradation of the algorithm performances.

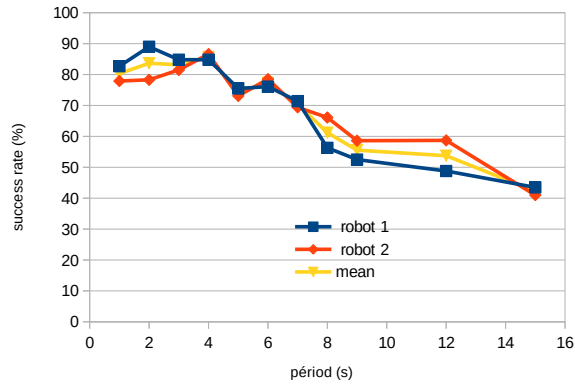


Fig. 6. Simulation measures

B. Adding Lures

In addition to the value of T , another element can impact the performances of the algorithm: the number of detected robots in the pictures. For that, we add lures in the area. We do that by adding some patterns that will be identified as robots by the computer, leading to larger sets of coordinates sent to actual robots. We still keep two mobile robots in the area and the lures are added between the two sub-areas as illustrated on Figure 7.

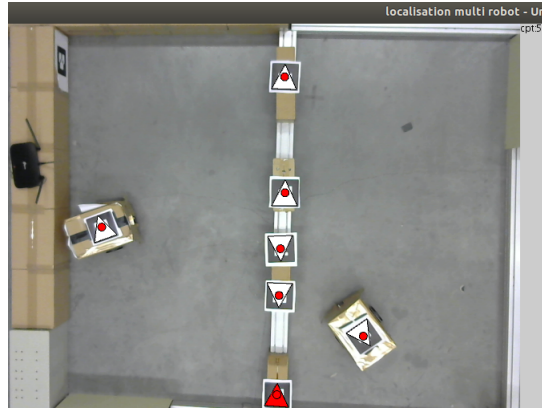


Fig. 7. Observed Area: 5 lures and 2 mobile robots ($N = 7, M = 5$)

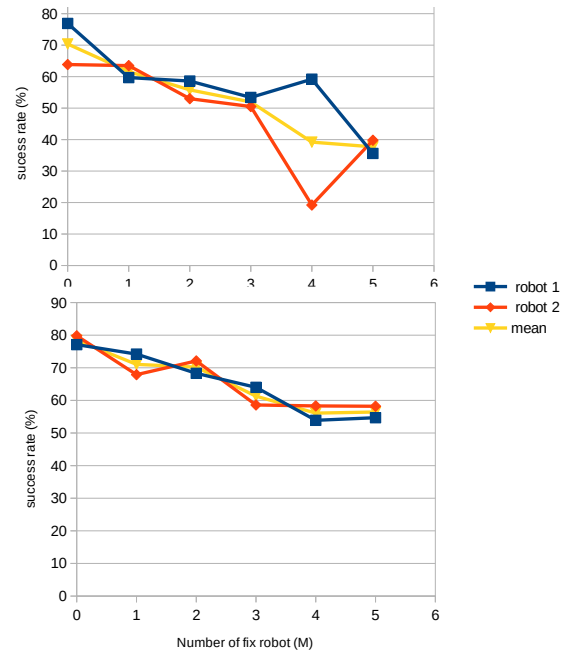


Fig. 8. Success rates when lures (M) are added. Results obtained from experiments with actual robots (top) and by simulation (bottom). Period between two pictures: $T = 7$ seconds

Figure 8 illustrates results for both experiments with actual robots and simulations. For both graphics $T = 7$. Graphics are very similar in both cases (experiments and simulations). We observe a linear decrease of the success rates with respect to the number of lures. The 10% difference noted previously between actual experiments and simulations is still observed for this configuration, however when the number of lures is greater than 4 it seems that actual robots have more difficulties to localize themselves. Additional experiments should be conducted to confirm or infirm these observations.

C. From actual experiments to simulation

The results obtained in the previous sections for both experiments with actual robots and simulation are very similar. The gap between results is close to 10% and has been confirmed by two series of experiments. This differ-

ence very likely comes from the imprecise nature of actual experimental measures, and for that reason, the maximum success rates of 75% with actual robots seem difficult to improve.

IV. FOUR MOBILE ROBOTS SCENARIO

In experimental robotics, the larger the number of robots, the trickier and time-consuming the problems related to the construction, the configuration and the maintenance of them. Simulation platform offers an acceptable alternative. It enables the implementation of situations difficult to set up on the experimental platform with actual robots. Despite the differences in the self-localization algorithm performances between the experimental platform and the simulation one, as described in the previous section, we are however very confident in the validity of the simulation approach. Indeed, if there exist numerical differences between results provided by both platforms, these results are consistent with each other. The main difference between actual robots and simulated ones is the addition of sonar sensors on the virtual ones, as illustrated (Figure 9). The general movement algorithm remains the same (Algorithm 1). As previously, some lures have been added in order to increase the combinatorics.

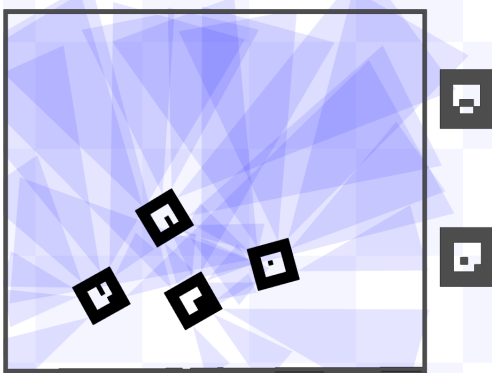


Fig. 9. Four mobile robots and two lures ($N = 6, M = 2$)

In line with previous results, the smaller the period between two consecutive pictures, the better the obtained success rates for the self-localization algorithm, as illustrated by Figure 10. In addition when the number of mobile robots increases we observe a degradation of the results. Such decrease in the success rate is likely to happen when both the number of robots increases as well as the delay between two pictures. Indeed, odometry accuracy is not perfect and is not continuously measured. So, the larger the time between two updates, the larger the area where the robot is supposed to be located in, based on odometric measures. Thus, when the period between two pictures increases, the previously mentioned area increases too. As a consequence the localization is less accurate, since the association between available coordinates sets and estimated localization becomes more and more approximate and the method error prone. However, the degradation of the success rate seems to be linear in both the increase of T and $N - M$.

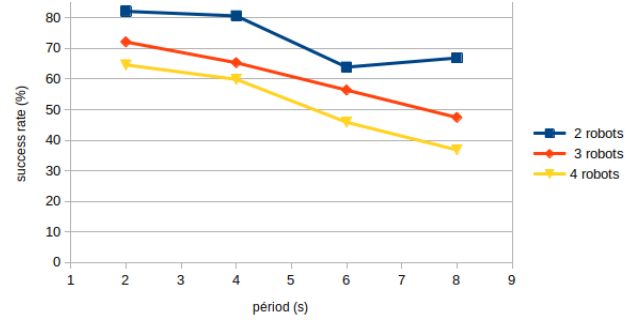


Fig. 10. Success rates with respect to T (period between two consecutive pictures).

TABLE I
WIRELESS NETWORK: PACKET LOSSES

	distance AP	#emissions	#receptions
Static	0.3m	100	99
Static	1m	100	96
Static	2m	100	95
Moving	...	100	95

V. ANALYSIS OF SOURCES OF INACCURACY OF MEASURES

This section focuses on the analysis of inaccuracy of measures for the experimental platform with actual robots. Several sources of errors have been identified and analyzed.

The floor on which the robots were moving was not perfect and an average of 2% of error was recorded for a linear movement of a robot starting from one side of the area to the other side. Of course, during the experiment, these errors are cumulative.

For the wireless network, we have noted some packet losses during the experiment (Table I). These losses occurred between the computer and the robots. When a robot does not receive the set of coordinates from the computer, this is equivalent to doubling the period T for this robot. Based on the measures, we have estimated that a 5% of lost transmissions has an impact of 2.5% on the success rate for the involved robot.

Another source of error is due to the delay between the time at which a picture is taken and the date at which robots are processing coordinates sets, a delay during which robots are still moving and odometry measures continue. But the impact of this delay is difficult to evaluate since we do not have a global clock for the system.

However, the most important source of error stems from odometry inaccuracy which concerns both linear and angular measures. They are recurrent and impact each processing. They are related to robots's structural imperfections, to defects of odometric sensors with their own accuracy, to the weight of robots, mainly due to the batteries weight that provides a good stability for linear movements but presents drawbacks for angular movements. They are also related to the adhesion properties between wheels and the ground and to the choice of the parameters that are considered as identical to every robot while some small differences may

occur between them. In order to have an idea of these differences, some measures have been performed for linear and angular movements. We observed an average linear deviation of 1.3% and an average angular deviation of 2.5%, which results in a deviation of 14cm for a 2π rotation.

VI. CONCLUSION AND PERSPECTIVES

The main problem addressed in this work is self-localization of anonymous robots using periodically available pictures, images of the area in which robots are moving, together with odometric measures.

The method, proposed in [15], has been implemented on actual robots and was validated through real experiments. The sensitivity of the method was also investigated by making varying two critical parameters, the period between two pictures and the number of lures in the scene, for the different tests.

But, in this work we also wanted to verify if results, obtained from a simulation platform reproducing actual experimental conditions, are reliable in regard with measures performed on an experimental platform operating actual robots. Thus, after driving some experiments with actual robots we've conducted the same experiments on the simulation platform. We have shown that results are qualitatively similar on both platforms but that some differences for the algorithm's performances are observed. The existence of various sources of inaccuracy of measures explains much of this difference, but the simulation-based approach can be considered as reliable. The experiments led for a four mobile robots scenario confirm that. Indeed, results obtained are in line with what could be expected in reality: the performances of the algorithm are degraded when the number of lures is growing but mainly when the period between two consecutive pictures increases.

We now plan to add error terms, modeled as random values within the error range, to the robot's commands in the simulation platform. Simulated results, with and without added error terms, will then be compared for an increasing number of mobile robots. We hope this will provide insights about the potential impact of observed inaccuracies during scale up.

This work opens many perspectives, some are improvements of the current method but others open new horizons in the domain of the control of fleets of robots. In particular we envision the possibility of using pictures obtained from a drone's camera, flying above a set of ground robots or a set of UAVs in order to provide them some more global information. This new direction would enable us to merge our different contributions to the domain of swarms of drones [2], [10], [9].

REFERENCES

- [1] Motilal Agrawal and Kurt Konolige. Real-time localization in outdoor environments using stereo vision and inexpensive gps. In *18th International conference on pattern recognition (ICPR'06)*, volume 3, pages 1063–1068. IEEE, 2006.
- [2] M. Bastourous, J. Al-Tuwayyij, F. Guérin, and F. Guinand. Image based visual servoing for multi aerial robots formation. In *2020 28th Mediterranean Conference on Control and Automation (MED)*, pages 115–120, 2020.
- [3] Xavier Défago and Akihiko Konagaya. Circle formation for oblivious anonymous mobile robots with no common sense of orientation. In *Proceedings of the second ACM international workshop on Principles of mobile computing*, pages 97–104, 2002.
- [4] Paola Flocchini, Giuseppe Prencipe, Nicola Santoro, and Peter Widmayer. Arbitrary pattern formation by asynchronous, anonymous, oblivious robots. *Theoretical Computer Science*, 407(1-3):412–447, 2008.
- [5] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A probabilistic approach to collaborative multi-robot localization. *Autonomous robots*, 8(3):325–344, 2000.
- [6] Antonio Franchi, Giuseppe Oriolo, and Paolo Stegagno. Probabilistic mutual localization in multi-agent systems from anonymous position measures. In *49th IEEE Conference on Decision and Control (CDC)*, pages 6534–6540. IEEE, 2010.
- [7] Nuwan Ganganath and Henry Leung. Mobile robot localization using odometry and kinect sensor. In *2012 IEEE International Conference on Emerging Signal Processing Applications*, pages 91–94. IEEE, 2012.
- [8] François Guérin, Frédéric Guinand, Jean-François Brethé, Hervé Pelvillain, Adel Zentout, et al. Vision based target tracking using an unmanned aerial vehicle. In *2015 IEEE International Workshop on Advanced Robotics and its Social Impacts (ARSO)*, pages 1–6. IEEE, 2015.
- [9] F. Guinand, F. Guérin, and M. Bastourous. Alignment of three robots without communication nor localization*. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 647–654, Oct 2019.
- [10] Frédéric Guinand, François Guérin, and Paweł Łubniewski. Allowing people to communicate after a disaster using FANETs. In *Lecture Notes in Computer Science*, volume 12574, pages 181–193. Springer International Publishing, 2020.
- [11] Patric Jensfelt and Steen Kristensen. Active global localization for a mobile robot using multiple hypothesis tracking. *IEEE Transactions on Robotics and Automation*, 17(5):748–760, 2001.
- [12] Ryo Kurazume, Shigemi Nagata, and Shigeo Hirose. Cooperative positioning with multiple robots. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 1250–1257. IEEE, 1994.
- [13] Sampo Kuutti, Saber Fallah, Konstantinos Katsaros, Mehrdad Dianati, Francis Mccullough, and Alexandros Mouzakitis. A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications. *IEEE Internet of Things Journal*, 5(2):829–846, 2018.
- [14] Paul M Maxim, Suranga Hettiarachchi, William M Spears, Diana F Spears, Jerry C Hamann, Thomas Kunkel, and Caleb Speiser. Trilateration localization for multi-robot teams. In *ICINCO-RA (2)*, pages 301–307, 2008.
- [15] Olivier Poulet, François Guérin, and Frédéric Guinand. Self-localization of anonymous mobile robots from aerial images. In *2018 European Control Conference (ECC)*, pages 1094–1099. IEEE, 2018.
- [16] Arnau Ramisa, Adriana Tapus, David Aldavert, Ricardo Toledo, and Ramon Lopez De Mantaras. Robust vision-based robot localization using combinations of local feature region detectors. *Autonomous Robots*, 27(4):373, 2009.
- [17] Ichiro Suzuki and Masafumi Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM Journal on Computing*, 28(4):1347–1363, 1999.
- [18] Luka Teslić, Igor Škrjanc, and Gregor Klančar. Ekf-based localization of a wheeled mobile robot in structured environments. *Journal of Intelligent & Robotic Systems*, 62(2):187–203, 2011.
- [19] Sebastian Thrun, Dieter Fox, Wolfram Burgard, and Frank Dellaert. Robust monte carlo localization for mobile robots. *Artificial intelligence*, 128(1-2):99–141, 2001.
- [20] Ching-Chih Tsai. A localization system of a mobile robot by fusing dead-reckoning and ultrasonic measurements. *IEEE Transactions on Instrumentation and Measurement*, 47(5):1399–1404, 1998.