2012 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 7-12, 2012. Vilamoura, Algarve, Portugal

Initial Deployment of a Robotic Team - A Hierarchical Approach Under Communication Constraints Verified on Low-Cost Platforms

Micael S. Couceiro, *IEEE Student Member*, Carlos M. Figueiredo, David Portugal, *IEEE Student Member,*

Rui P. Rocha, Nuno M. F. Ferreira

Abstract**—In most real multi-robot applications, *e.g.*, search-and-rescue, cooperative robots have to fulfill their tasks while driving and communicating among themselves without the aid of a network infrastructure. However, initially deploying autonomously a wireless sensor robot network in a real environment has not taken the proper attention. This paper presents an autonomous and realistic initial deployment strategy, based on a hierarchical approach, in which exploring agents, denoted as scouts, are autonomously deployed through explicit cooperation with supporting agents, denoted as rangers. To evaluate the initial deployment strategy proposed, experimental results with a team of heterogeneous robots are conducted using modified low-cost platforms previously developed by the authors. Preliminary results show the effectiveness of the method and pave the way for a whole series of possible new approaches.**

I. Introduction

The initial deployment of mobile robots has not been fully addressed and only a few studies evaluating its relevance have been conducted. For instance, in a search and rescue (*SaR*) mission, robots need to move in a catastrophic scenario in order to find survivors. When robots are transported to the catastrophe site, they need to be properly deployed. The deployment problem consists in deciding how many robots and where they will be initially located before performing the mission (*e.g.*, coverage, herding, formation and others). Despite the lack of works studying the initial deployment effect on the performance of the robotic team, a wrong decision about the number of robots and their initial location may greatly jeopardize the mission [1]. For instance, in several iterative optimization problems, it has been shown that good initial estimates can lead to faster convergence (*e.g.* [2]).

One of the first works that addressed the effect of different initial deployments was presented in [3]. The authors evaluated their coverage algorithm using both centralized and random initial deployments and concluded that the algorithm convergence was slower using a random initial deployment but tended to lead to better overall coverage for sparse topologies. The work of [4] extends the sensory capability of plume tracking systems using swarms of robots deployed in the proximities of a common starting point. However, the authors do not go to any lengths to explore the plume tracking effectiveness within other initial deployment strategies. In [5], a strategy to assign starting points and orientations of robots within circles of different radius around a prey is presented. Despite the apparent advantages of this deployment strategy in this context, no other strategies were evaluated, thus being hard to predict if the number of unsuccessful trials is somehow related with the initial deployment of robots. The authors in [6] and similarly in [7] presented a robotic swarm algorithm in which the initial position and velocity of robots were randomly generated within an area limited to one corner near the origin of coordinates of the workspace. In [8], a three dimensional deployment strategy was explored. The main difference with other works resides in the fact that robots autonomously move in a 3D space (*e.g.*, coordinated formation flight and reconfiguration of unmanned aerial vehicles [9]).

Despite the scientific accomplishments of the previously described works, none of them specifies how robots are initially deployed within a scenario – most of the works assume that robots are manually deployed or they simply "start" in some location. The ones that do not assume this hard restriction usually have the robots entering in the environment through the same gate and move to predetermined points, like [10]. An example of a more realistic approach was presented in [11] in which the authors divided the population of real robots into two different platforms: *rangers* and *scouts*. Despite the innovation of the work, the deployment strategy was accomplished through a launcher system. However, in most applications in unknown scenarios (*e.g.*, *SaR* missions) this would require robots to be able to measure the relative distance between themselves or to be equipped with global localization systems (*e.g.*, *GPS*) to allow an efficient processing of the exchanged information. Similarly to Rybsky's work [11], the approach herein proposed handles the initial deployment problem hierarchically dividing the heterogeneous population of robots into rangers and scouts. Each ranger handles the initial deployment of scouts in a distributed and autonomous fashion. To that end, the *TraxBot* platform previously presented in [12] acts as a ranger in order to allow the transportation of a maximum of five scouts enacted by *eSwarBot* platforms [13]. The ranger successively deploys the scouts, instructing them of their initial pose while maintaining a maximum communication range between scouts, thus guaranteeing the full connectivity of the wireless sensor robot network (*WSRN*).

There are two key contributions in this work. Firstly, an innovative systematic method for hierarchically deploying swarm agents in an unknown scenario, under communication constraints, which guarantees wide distribution in space to

This work was supported by PhD scholarships (SFRH/BD/64426/2009) and (SFRH/BD/73382/2010), the research project CHOPIN (PTDC/EEA-CRO/119000/2010) by the Portuguese Foundation for Science and Technology (FCT), the Institute of Systems and Robotics (ISR) and RoboCorp.

M. S. Couceiro, D. Portugal and R. P. Rocha are with the Institute of Systems and Robotics, University of Coimbra, Pólo II, 3030-290 Coimbra, Portugal, e-mail: {micaelcouceiro,davidbsp,rprocha}@isr.uc.pt

C. M. Figueiredo, N. M. F. Ferreira are with the Electrical Engineering Department, Coimbra Institute of Engineering, Rua Pedro Nunes - Quinta da Nora, 3030-199 Coimbra, Portugal, e-mail: {cfigueiredo,nunomig}@isec.pt

enable efficient swarm exploration is proposed. Secondly, the approach is verified in a real and heterogeneous multi-robot system, which is carefully described, focusing on the extension of the ranger platforms to support the transportation of scouts.

II. SCOUTS

Scouts should be small, easily deployable and able to sense their environment. For this reason, *eSwarBot* (Educational Swarm Robot) platforms were used [13]. These robots are ideal for studying emergent behavior and self-organization in bio-inspired societies (*i.e.,* swarm robotics)

A. eSwarBot

The *eSwarBot* platform (Fig. 1) consists on differential ground platforms with an *Arduino Uno* processing unit recently developed and described in detail in [13]. Although the platforms present a limited odometric resolution of 3.6 degrees while rotating and 2.76 millimeters when moving forward, their low cost (around 175€) and high energetic autonomy (maximum run time up to 4 hours) allow performing experiments with a large number of robots.

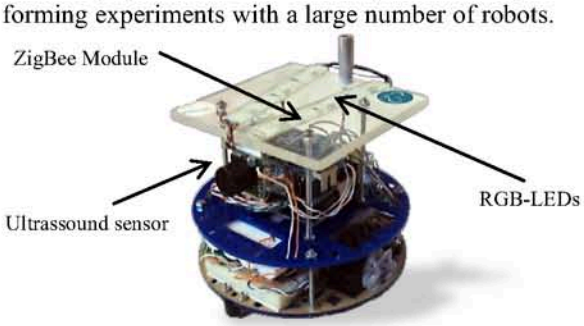

Figure 1. The *eSwarBot* scout.

As it can be seen in Fig. 1, *eSwarBots* are equipped with *RGB-LED*s that allow representing a wide range of different colors that represent different states. The initial deployed robots are identified using the red color while the blue color is used to identify when robots start their mission (*i.e.*, end of deployment). It is noteworthy that, in this paper, we do not consider the scouts' mission after the initial deployment, we are focused instead on validating a practical and effective way to deploy robots for swarm foraging tasks with hard communication constraints. Some hardware specifications of the *eSwarBot* are presented in Table I.

TABLE I. ESWARBOT HARDWARE SPECIFICATIONS.

Voltage Range [V]	9-14
Electric Current in Operation [mA]	550
Electric Current in Standby [mA]	90
Maximum Speed [m/s]	0,15
Weight [g]	600
Width [mm]	126
Length [mm]	126
Height [mm]	100

B. Inter-Robot Communication

*WSRN*s can be implemented using several wireless technologies, such as *Bluetooth*, *ZigBee* or *WiFi*. The definition itself does not imply any restrictions to the implementing devices. In this work, inter-robot communication to define the initial position of scouts was carried out using *ZigBee* 802.15.4 wireless protocol [14].

All robots of the team are endowed with *XBee* modules that communicate with the microcontroller via SPI interface. *XBee Series 1* is based on *ZigBee*/802.15.4 silicon from *Freescale* [15]. Its 802.15.4 firmware feature set makes it ideal for point-to-point and point-to-multipoint (star) topologies. Hence, these modules are a suitable solution for multi-robot systems (*MRS*) since they present power consumption near 10μA when in sleep mode and 50mA while sending and receiving data. Furthermore, since *MRS* may be formed by dozens of robots (*i.e.*, nodes), the *ZigBee* protocol is the most adequate option since it can theoretically support up to 65536 network nodes.

III. RANGER

As previously described, rangers act as supporting platforms that need to carry the team rapidly into place and deploy the scouts. They must be extremely robust and be able to transport multiple scout platforms and process the sensor data, acting as coordinators for the team. Therefore, *TraxBot* platforms (Fig. 2) were used as rangers, being suitable for both outdoor and indoor operation with high autonomy. These platforms have also been recently developed [12].

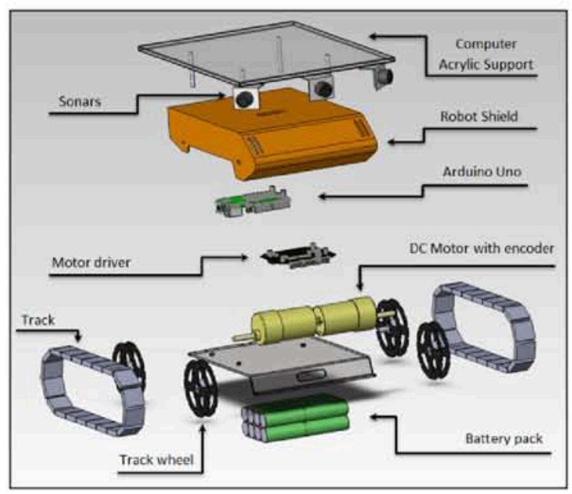

Figure 2. The *TraxBot* ranger.

A. TraxBot

The *TraxBot* platform is a differential drive system built upon the *Traxster II Robot educational Kit* [12], equipped with 2 *DC* gearhead motors with quadrature wheel encoders and rubber tracks. It is worth mentioning that rangers need to be able to communicate with scouts. Therefore, similarly to the *eSwarBot*, the processing unit consists of an *Arduino Uno* board endowed with a *Xbee Series 1 Shield*. Also, the *TraxBot* can reactively avoid obstacles with a maximum range of approximately 6 meters using three *Maxbotix Sonars MB1300* mounted below the top acrylic support, as seen in Fig. 2. Some other specifications are presented in Table II.

TABLE II. TRAXBOT HARDWARE SPECIFICATIONS.

Voltage Range [V]	9-14
Electric Current in Operation [mA]	1200
Electric Current in Standby [mA]	110
Maximum Speed [m/s]	0,95
Weight [g]	2045
Width [mm]	203
Length [mm]	229
Height [mm]	110

B. TraxBot Conveyor Kit

Despite the robustness of the *TraxBot* (*i.e.*, aluminum and stainless steel platform equipped with high power *DC* motors), an extension conveyor kit was necessary to support 5 *eSwarBots* on top of the platform (Fig. 3).

A major part of the conveyor was built with equipment from damaged printers (*e.g.*, conveyor pulleys, stepper motors, gearboxes). The conveyor belt consists of two layers of strong tissue offering a suitable adhesion to the scouts' wheels. An under layer made of *PVC* provides a linear strength and shape to the conveyor belt. It should be noted that, in *SaR* applications, *i.e.*, real unknown and uneven terrain, a cleated conveyor belt (*i.e.*, partitioned upper layer) could be used to improve the capability of the ranger to keep the scouts on top of it.

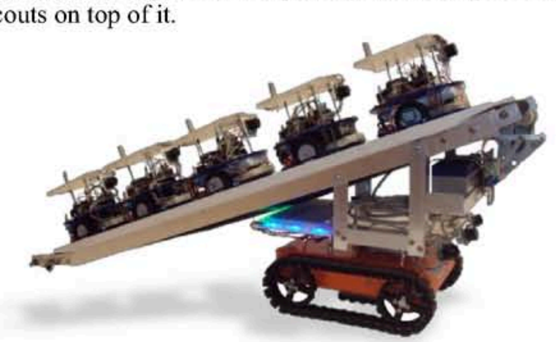

Figure 3. The *TraxBot Conveyor Kit* loaded with 5 *eSwarBots*.

The driving pulley is connected to a single 6-wire unipolar stepper motor through a gearbox. As a stepper motor conveyor, it allows precise adjustments of velocity, torque, acceleration, deceleration and current. Hence, only a 6-wire cable needs to be plugged in the *TraxBot* containing 4 digital outputs of the *Arduino* board and *TTL* power supply. The 4 digital outputs are connected to two *H*-bridges used to drive the stepper motor. To allow a higher autonomy of the system, the conveyor kit is equipped with an independent 12V 1300mAh lead acid battery followed by a 5.4V chopper since the stepper motor can consume up to 1000mA at 5.4V.

The *TraxBot Conveyor Kit*, even being entirely made of aluminum, increases the original weight of the *TraxBot* platform to 4.2Kg (*i.e.*, unladed weight) and 7.1Kg at full load (*i.e.*, with 5 *eSwarBots* on top). Due to stepper motor and gearbox limitation, the *TraxBot Conveyor Kit* is only able to support a maximum weight of 4.5Kg without suffering from any sliding effect on the driving pulley. Nevertheless, this is more than enough as the *TraxBot* mobile platform is unable to efficiently rotate when carrying a weight of approximately 5.0Kg above the unladed weight.

The initial deployment process is simple: First of all, scouts are manually loaded and equally distributed on the conveyor belt, *i.e.*, ranger carrier system. After the ranger reaches the desired position to deploy a scout (*cf.*, Algorithm 1 from next section), the stepper motor conveyor is controlled by the ranger robot, to move the scout robot towards the ground. An explanation of how the ranger decides where to deploy the scouts is presented in next section.

C. Initial Deployment Strategy

In this work, one can define the initial deployment problem as it follows: Consider a population of *N scouts*, where each scout is both an exploring agent of the environment and a mobile node of a *WSRN* that performs packet forwarding, according to a paradigm of *multi-hop communication*. The goal is to ensure that the *N* scouts are initially deployed by a robot, denoted as *ranger*, in an unknown environment, while avoiding areas of no interest (*e.g.*, obstacles) and ensuring that the *WSRN* is connected.

Since this work focuses on unknown environments, rangers reactively deploy scouts, while avoiding obstacles, based on the maximum distance between the previously deployed scout and itself. In other words, scouts are successively deployed, one after another, by the same ranger such that the pose of the n^{th} robot always depends on the pose of the $(n-1)^{th}$ robot and the existence of obstacles in the path between them.

The behavior of a ranger transporting *N* scouts can then be described as it follows: The ranger first moves to a random initial position while avoiding obstacles using a simple wall-follower algorithm. When the ranger reaches the desired initial position, or its vicinities (due to obstacles constraints), it will unload the first scout and inform it that it will start at time $t = 0$ with the following position:

$$x_1[0] = x_r - l_r \begin{bmatrix} \cos\theta_r \\ \sin\theta_r \end{bmatrix}, \quad (1)$$

and orientation of:

$$\theta_1[0] = \theta_r, \quad (2)$$

wherein x_r and θ_r are the current position and orientation (*i.e.*, pose) of the ranger and l_r is the distance from the center of the ranger to the idler pulley ($l_r = 550$ mm in *TraxBot* platforms). At this point, the scout will then wait for a starting message from the ranger to begin their mission. After deploying the first scout, the ranger will choose a new random position within a circumference with d_{max} radius and centered in $x_1[0]$ and starts moving apart from the unloaded robot while avoiding obstacles.

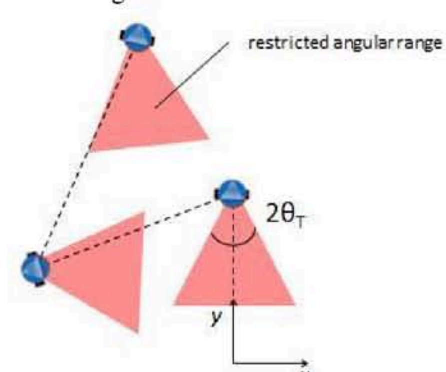

Figure 4. Dispersion of robots resulting from the angular restriction.

The computation of this new position takes into account a proximity constraint to avoid deploying robots near other previously deployed robots, thus ensuring an adequate dispersion of all robots. To that end, the angle that allows determining a new position is defined as:

$$\theta_i^d[0] = rand(\alpha),\ \alpha \in [0, \theta - \theta_T[\ \cup\ ]\theta + \theta_T, 360[, \quad (3)$$

wherein θ_T is an angular threshold and θ can be defined as:

$$\theta = \sum_{j=1}^{i-1} \theta_j[0]\text{-}180, \quad (4)$$

The angle θ takes into account all previous rotations made by the ranger. Note that due to the cumulative sum of all previous rotations, θ needs to be reduced between 0º and 360º. The distribution of robots increases as the angular threshold θ_T increases, as it can be seen in Fig. 4. It is noteworthy that the ranger may still be unable to reach this desired location because obstacles may constrain the ranger's movements. Nevertheless, when the Euclidean distance between itself and the previously deployed robot reaches the maximum desired value, *i.e.*, $d = d_{max}$, then the ranger will unload the second robot, once again informing it of its pose $\langle x_2[0], \theta_2[0] \rangle$. The same process will be replicated for the remaining scouts until the ranger unloads all N scouts.

ALGORITHM I
RANGER INITIAL DEPLOYMENT ALGORITHM

$\theta = 180$
$x_1^d[0] = random\_position$ // desired initial position of scout 1
$goto\_position(x_r, x_1^d[0])$ // travel from the current position to scout 1 position while avoiding obstacles (*e.g.*, wall following algorithm) and deploys scout 1
$x_1[0] = x_r - l_r \begin{bmatrix}\cos\theta_r \\ \sin\theta_r\end{bmatrix}$ // initial pose of scout 1
$send\_pose(1, x_1[0], \theta_1[0])$ // informs scout 1 of its initial pose
For $i = 2{:}N$
  $\theta = \theta_{i-1}[0] - \theta$
  $\theta_i^d[0] = rand(\alpha), \alpha \in [0, \theta - \theta_T[\ \cup\ ]\theta + \theta_T, 360[$
  $x_i^d[0] = x_{i-1}[0] + d_{max} \begin{bmatrix}\cos\theta_i[0] \\ \sin\theta_i[0]\end{bmatrix}$
  While $d_i < d_{max}$
    $[d_{i+1}, q_{+1}] = move(x_r, x_i^d[0])$ // move forward from scout $i-1$ position to scout i desired position while avoiding obstacles and deploys scout i
  $x_i[0] = x_r - l_r \begin{bmatrix}\cos\theta_r \\ \sin\theta_r\end{bmatrix}$ // initial pose of scout i
  $send\_pose(i, x_i[0], \theta_i[0])$ // informs scout i of its initial pose
$send\_start()$ // broadcast information to start the mission

After deploying the whole team, the ranger broadcasts a message to start the mission. The message will be replicated by scouts inside its communication range, thus reaching all robots within the team. When the message is received, all scouts become aware that their teammates are already deployed in the environment and, consequently, they can start their missions. Algorithm I presents the initial deployment strategy of a ranger and N scouts.

IV. PRELIMINARY RESULTS AND EXPECTATIONS

In order to evaluate the strategy proposed, a set of 10 preliminary trials for each $\theta_T = 0, 15, 30, 60$ was conducted on a laboratory scenario with an area of 2.5 × 4.5 m, composed of three polygonal obstacles, as it can be seen in Fig. 5. These experimental tests were carried out specifically to analyze the efficiency of the approach and, in the future, the approach should be tested in more challenging real world scenarios. For this purpose, five *eSwarBots* were used as deployed scouts and one *TraxBot* as deploying ranger. The communication maximum range d_{max} was set to 2 meters. The *ground truth* pose of scouts was obtained using a *Gigabit Ethernet Color Camera* of 1.7 *MP* at 5 *fps* mounted in an overhead position of 2.55 meters height by means of a real-time tracking system developed using the machine vision software *MVTech Halcon*. Note that this tracking system has no influence on the positioning of scouts and localization of the ranger, which exclusively relies on odometry estimates.

Figure 5. Experimental setup.

To measure the dispersion of the deployment strategy, a metric based on the average distance from each scout to the centroid $x_c[0]$ was used:

$$\sigma_S = \frac{1}{N_S} \sum_{i=1}^{N_S} \|x_i[0] - x_c[0]\|. \quad (5)$$

Also, the minimum distance (d_{min}) between two neighbor robots was also used to depict the efficiency of the deployment dispersion (Fig. 6).

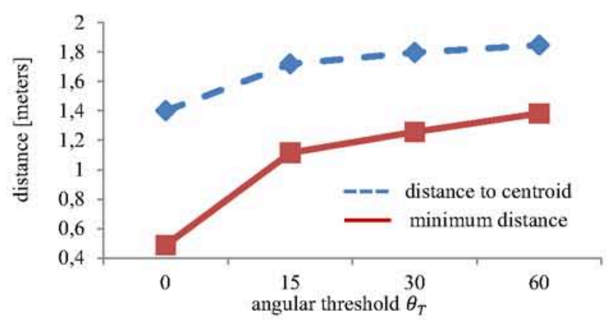

Figure 6. Dispersion of the deployment strategy.

Fig. 6 shows the influence of θ_T on the deployment dispersion. On the *yy'* axis the distance in meters is presented and the *xx'* axis presents four different values of angular threshold θ_T. It is noteworthy that a θ_T equal to zero corresponds to a random deployment strategy that disregards the previously deployed scouts. By averaging σ_s *and* d_{min} over 10 experimental trials for each value of θ_T, it can be seen that, as the angular threshold increases, not only the minimum distance between robots becomes greater, but also the distance to the centroid grows, contributing to the wide space distribution that is intended in the approach. Given this correlation between θ_T and the robots' dispersion, the value of parameter θ_T should therefore be adjusted according to the application requirements.

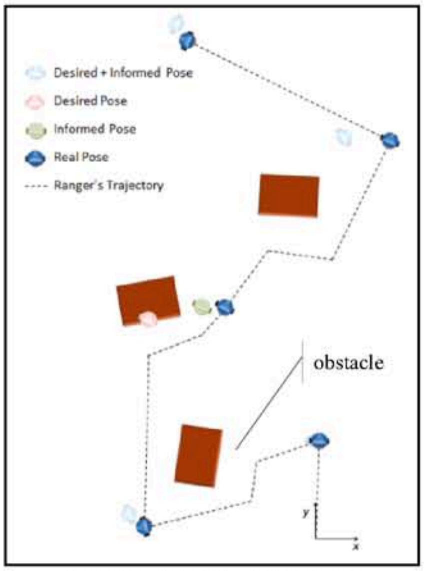

Figure 7. Illustration of one trial.

One of the experimental trials using $\theta_T = 30$, is represented in Fig. 7, in which it becomes clear by the ranger's trajectory, that it was able to deploy scouts, using Algorithm I, while moving away from obstacles. As expected, some positional errors are propagated during the experiments, due to the *TraxBot*'s odometry limitations and the need to avoid obstacles. Consequently, the real pose of the deploying scout slightly differs from the desired pose according to Algorithm I. A particularly interesting case occurs when the ranger is about to deploy the third scout. In this particular situation, the ranger found an obstacle near the deploying area, therefore it moved away from it, and instead of deploying the scout in the initially desired position, which was in an occupied location, it deployed the scout in free space near the obstacle. Thus, in this situation the desired pose differs from the pose that was communicated to the scouts (*i.e.*, informed pose), using (1) and (2), since there was a last second change of plans.

In order to analyze the experimental results as a whole, particularly the positional errors involved in the initial deployment, a methodology of least-squares fitting of ellipses to 2D points was adopted (*cf.*, [16]). This allows not only to analyze the precision, *i.e.*, the dispersion related to the real position, but also the accuracy of the data, *i.e.*, how close the informed pose is to the actual pose. The graphs in Fig. 8 illustrate the positional error presented in the deployment of the 5 scouts in the course of 10 experimental trials for $\theta_T = 30$. Note that the origin of the coordinate system is the informed pose of the scout and the red points correspond to their real pose. It can be seen that the area of the ellipse tends to grow as the robots are deployed, which means that the positional precision tends to decrease. The only exception occurs in the deployment of the last robot; one possible explanation is that when the fifth scout is deployed, the ranger is generally far from the critical zone of the environment, with a higher obstacle density. Additionally, the center of the ellipse suffers a slight shift in all presented charts, which shows that the accuracy of deployment tends to be bounded.

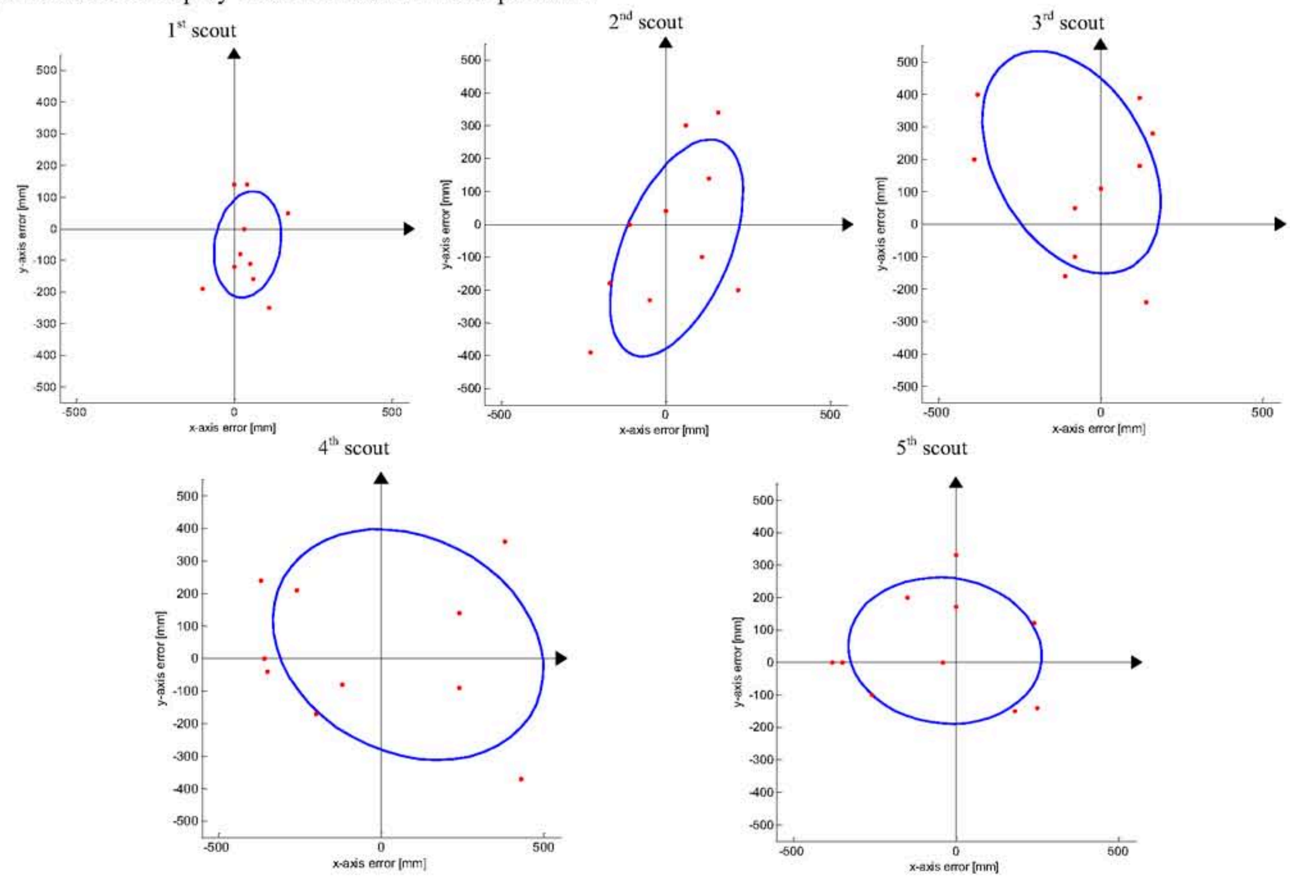

Figure 8. Error pose least-squares fit of elipse.

The experimental tests presented in this section confirm that the algorithm for initial deployment of robots produces an effective distribution of the scout robots in the testing arena, which is very important for exploratory swarm tasks, especially when no information about the environment is known. Note that the error analysis presented shows an odometric limitation that is inherent to the platforms and not to the approach itself. By validating this valuable approach in real platforms, the foundations were laid for a whole series of possible new methods for positioning scout robots efficiently. The herein presented approach progressively deploys scouts in the environment in a distributed fashion, by moving away from the last deployed scout. Note that in this paper, the focus was on describing and verifying the approach. Further analysis should be conducted to study the effect of initial deployment in the performance of cooperative swarm robotic teams. Moreover, to deal with the uncertainty resulting from real and more complex scenarios, the odometry is clearly insufficient to achieve a reliable target deployment. Hence, Simultaneous Localization and Mapping (*SLAM*) algorithms and laser range finders should be used (*cf.*, [17]).

Since it is the authors' belief that extra attention should be given to the initial deployment problem by the robotics community, it is foreseen that many contributions will be presented, in the near future, to better understand this phenomenon.

V. Conclusion

This paper proposes a hierarchical and heterogeneous multi-robot system composed by rangers and scouts to test an initial deployment approach for swarm foraging, which was evaluated using real physical platforms. The deployment strategy considers the maximum range between robots to allow the full connectivity of the *WSRN*. To allow the deployment of multiple scouts, the ranger platform was augmented with a conveyor kit, which was described in detail, supporting up to 5 scouts. Results show that, despite odometry errors, scouts turn out to be uniformly deployed within the test scenario. As future work, the deployment strategy may be extended to consider the *RSSI* level, in terms of signal quality between scouts, when unloading them. This is particularly important in cases when an obstacle is in the communication path between two robots. When this situation occurs, they may have a significantly lower *RSSI* value than two other robots that are at the same distance but without any interference. Another extension of the deployment strategy is to allow robots to explore the scenario immediately after they are deployed, instead of waiting for the deployment of the whole team. This would allow the ranger to adapt the next release point according to the already explored environment. Additionally, in the future, the method described in this paper is intended to be used for initially deploying scouts in foraging tasks, in order to further analyze the effect of initial deployment in swarm performance.

Acknowledgment

The authors gratefully acknowledge *Soluções de Automação e Robótica* (*SAR*) for their contribution and feedback.

References

[1] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. S. G. Lee, "Deployment Strategy for Mobile Robots with Energy and Timing Constraints," in *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA05*, Barcelona, Spain, 2005, pp. 2827-2832.

[2] D. Marquardt, "An Algorithm for Least-Squares Estimation of Nonlinear Parameters," *SIAM J. Appl. Math.*, vol. 11, pp. 431-441, 1963.

[3] N. Correll, J. Bachrach, D. Vickery, and D. Rus, "Ad-hoc Wireless Network Coverage with Networked Robots that Cannot Localize," in *IEEE International Conference on Robotics and Automation Kobe International Conference Center*, Kobe, Japan, 2009, pp. 3878-3885.

[4] S. Kazadi, "Extension of Plume Tracking Behavior to Robot Swarms," in *Proceedings of the 7th World Multi Conference on Systematics, Cybernetics and Informatics, SCI2003, Special Session on Swarm Engineering*, Orlando, Florida, USA, 2003.

[5] R. Groß, R. O'Grady, A. L. Christensen, and M. Dorigo, "The Swarm-bot experience: Strength and mobility through physical cooperation," in *Handbook of Collective Robotics*, S. Kernbach, Ed. Pan Stanford Publishing, 2011, ch. 2.

[6] S. D. Xue and J. C. Zeng, "Control over swarm robots search with swarm intelligence principles," *Journal of System Simulation*, vol. 20, no. 13, pp. 3449-3454, 2008.

[7] M. Kloetzer and C. Belta, "Hierarchical abstractions for robotic swarms," in *Proceedings in IEEE International Conference on Robotics and Automation, ICRA06*, Orlando, FL, 2006, pp. 952-957.

[8] G. Lee, Y. Nishimura, K. Tatara, and N. Y. Chong, "Three Dimensional Deployment of Robot Swarms," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS10*, Taipei, Taiwan, 2010, pp. 5073-5078.

[9] G. Hattenberger, S. Lacroix, and R. Alami, "Formation Flight: Evaluation of autonomous configuration control algorithms," in *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, USA, 2010, pp. 2628-2633.

[10] Y. Mei, Y.-H. Lu, Y. C. Hu, and C. S. G. Lee, "Deployment of Mobile Robots with Energy and Timing Constraints," *IEEE Transaction on Robotics*, vol. 22, no. 3, pp. 507-521, 2006.

[11] P. E. Rybski, et al., "Enlisting Rangers and Scouts for Reconnaissance and Surveillance," *IEEE Robotics & Automation Magazine*, vol. 7, no. 4, pp. 14-24, 2000.

[12] A. Araújo, D. Portugal, M. S. Couceiro, C. Figueiredo, and R. P. Rocha, "TRAXBOT: Assembling and Programming of a Mobile Robotic Platform," in *Proceedings of the 4th International Conference on Agents and Artificial Intelligence (ICAART'2012)*, Vilamoura, Algarve, Portugal, 2012.

[13] M. S. Couceiro, C. M. Figueiredo, J. M. A. Luz, N. M. F. Ferreira, and R. P. Rocha, "A Low-Cost Educational Platform for Swarm Robotics," *International Journal of Robots, Education and Art*, 2011.

[14] Z. Alliance. (2011, Aug.) ZigBee Alliance. [Online]. http://www.zigbee.org

[15] XBee. (2011, Aug.) XBee™/XBee-PRO™ OEM RF Modules datasheet. [Online]. http://ftp1.digi.com/support/documentation/90000982_A.pdf

[16] A. W. Fitzgibbon, M. Pilu, and R. B. Fisher, "Direct least-squares fitting of ellipses," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 476-480, 1999.

[17] M. S. Couceiro, A. R. Lopes, N. M. F. Ferreira, A. G. Ferreira, and R. P. Rocha, "Toward the Concept of Robot Society: A Multi-Robot SLAM Case Study," in *Mathematical Methods in Engineering*. Coimbra, Portugal: Springer, 2012.

[18] Maxstream. (2011) Product Manual v1.xAx - 802.15.4 Protocol. [Online]. http://elmicro.com/files/sparkfun/xbee-manual.pdf