

# Relative Localization and Communication Module for Small-Scale Multi-Robot Systems

Jim Pugh and Alcherio Martinoli  
Swarm-Intelligent Systems Group  
École Polytechnique Fédérale de Lausanne  
1015 Lausanne, Switzerland  
Email: {jim.pugh,alcherio.martinoli}@epfl.ch

**Abstract**—We characterize and improve an existing infrared relative localization/communication module used to find range and bearing between robots in small-scale multi-robot systems. Modifications to the algorithms of the original system are suggested which offer better performance. A mathematical model which accurately describes the system is presented and allows us to predict the performance of modules with augmented sensorial capabilities. Finally, the usefulness of the module is demonstrated in a multi-robot self-localization task using both a realistic robotic simulator and real robots, and the performance is analyzed.

## I. INTRODUCTION

Multi-robots systems is an emerging field which has recently drawn the attention of the robotics community. Multi-robots systems offer considerable advantages in comparison to single unit systems: simultaneous sensing and acting in physically different positions, reconfigurability of the system, redundancy, sometimes even the ability to achieve superlinear performances via division of labor. For several tasks such as distributed search [9], distributed coverage [2], and movement in formation [3], knowledge of one's own location and/or that of neighboring teammates is required.

Artificial localization systems can be roughly classified into two main categories: absolute and relative positioning systems. Absolute localization determines position in a global coordinate framework; examples include GPS-like solutions (also available for indoor environments, see for instance [16]) and external monitoring solutions, such as an overhead camera system. For certain types of tasks, only relative distance and bearing information is needed and, although this information could also be obtained with an absolute positioning system and an effective communication channel, the communication overhead resulting from this solution might be much higher and less scalable than a solution based on a relative localization system. Also, certain situations don't allow for effective global localization (e.g., a distributed system in an unknown environment in which no GPS reception is possible). Currently, very few on-board relative localization systems exist that can accomplish accurate and fully scalable performance.

Previous work has explored the potential for using relative localization for tracking of self-position and positions of other robots. Kurazume and Nagata [14] used an immobile broadcasting beacon approach to allow a pair of robots to keep track of their global coordinates by taking turns in moving or

broadcasting in simulation. Grabowski et al. [6] and Rekleitis et al. [19] used beacon-based approaches and combined position estimates from dead reckoning and relative positioning for mapping and exploration tasks, using an ultrasonic relative positioning system in the case of the former and a vision-based system for the latter. Roumeliotis and Bekey [20] used a Kalman filter to combine dead reckoning and information from an emulated relative positioning system to allow continuously moving robots to track their position more accurately over time. Fox et al. [4] used Monte Carlo localization to combine dead reckoning and relative positioning to allow robots to localize themselves within a previous mapped environment, using a laser range-finder and camera for relative positioning. Howard et al. [11], [12] used iterative relative positioning measurements in a chain of robots in close proximity to determine the relative position of any other robot in the team in both simulation and with real robots; a laser range-finder and camera were also used here to calculate relative positioning data.

Almost all the above techniques used sophisticated hardware requiring substantial processing, or emulated unimplemented relative positioning systems. The experiments often used sophisticated algorithms requiring robots to explicitly coordinate with other robots, and all of them required radio communication. All these aspects make it difficult to adapt the techniques to simple, microrobotic platforms that can be scaled down in complexity and size.

In this paper, we explore the capabilities of a simple relative localization module and how it can be used to enhance self-localization performance. The system used is a small-scale robot platform (Moorebots, see [22]). These robots are roughly cylindrical, with a 24 cm diameter. However, nothing prevents further miniaturization of the modules presented so that the same localization techniques could be used on robots of a few cm in diameter. The robots used are endowed with a PC/104 stack with wireless modules and an Arcom Viper processor board running a low-power 400MHz Intel PXA255 XScale; this configuration allows us to achieve 4 hours of autonomy. All experiments were performed in an arena 3.4x3.4 m<sup>2</sup>. The relative localization system is an on-board module which allows the robots to determine the range and bearing of other nearby robots based on the strength of a modulated infrared signal, and serves simultaneously as a local communication

channel with low bit rate (approximately 16 bits/s) [13]. The system can be used to communicate at a significantly higher bit rate if the relative positioning aspect is disabled.

For our self-localization experiment, we make use of our simple relative positioning system, which could be adapted for much smaller, simpler robots. The movements made by each robot are not dependent on the actions of other robots (except with respect to obstacle avoidance). We also explore how radio communication can enhance performance.

## II. RELATIVE LOCALIZATION SYSTEM

### A. Set-up Overview

The relative localization and communication system is built entirely from off-the-shelf components [13]. It is mounted on the PC/104 stack which is off-center on the Moorebot platform (see Fig 1). Signals are sent with infrared (IR) light and modulated/demodulated using RF technology. The system is controlled by a dedicated on-board PIC microcontroller. Each module has twelve IR Light Emitting Diodes (LEDs) arranged into four independently controllable clusters at 90° spacing for full 360° coverage. The range of the module goes up to 310 cm. Each module has four photodiode receivers at 90° spacing. Signal strengths at these receivers allow for calculation of the relative distance and angle to other robots. At distances less than 30 cm, signal strengths become distorted by optical effects, making accurate calculation impossible. Range is determined by the Received Signal Strength Indication (RSSI). The RSSI of the four receivers can be compared to calculate the direction of the other robot, as a receiver not directly facing the incoming signal will receive a lower signal intensity. The intensity of the received signal decays asymptotically as the range between robots increases. The module currently operates at 2 Hz, which allows a group of up to 10 robots to all exchange their relative positions concurrently. Three different transmission power levels are possible, allowing the operational range to be lessened to decrease interference between densely packed robots.

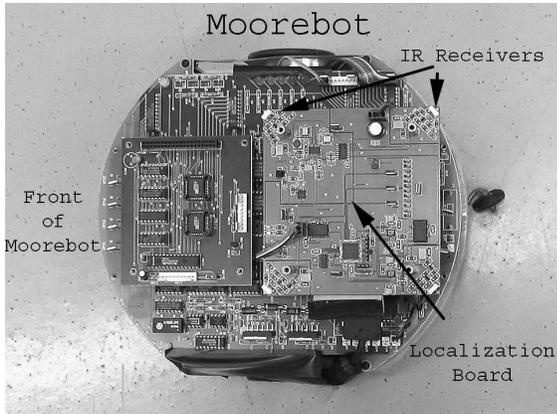


Fig. 1. Localization Module on a Moorebot

### B. Sensory System Modeling

The original relative localization system employed algorithms for calculating range and bearing of other robots which were not optimally fitted to the system (see [13] for details). We present here new algorithms to improve performance.

We assume receivers on the robot are evenly spaced. For any received transmission, approximately half these sensors will detect the signal; we select the sector of the robot which contains an even number of sensors with the highest received signal. We need now only calculate the angle offset ( $\theta$ ) from the center of this sector to find the bearing of the transmitting robot ( $\phi$ ).

Upon measuring the signal intensity at the individual receivers for different receiver orientations, we discovered that the intensity is very closely modeled by:

$$r' = r\sqrt{\cos(\theta)} \quad (1)$$

where  $r$ , which we will call the range term, would be the signal intensity at a receiver which is directly facing the transmitting robot. Let there be  $2n$  sensors facing the transmission. Let  $r'_{-1}, \dots, r'_{-n}$  be values of the sensors which have a lesser angle than the center of the receiving sector,  $r'_1, \dots, r'_n$  be the values of the sensors which have a greater angle than the center of the receiving sector, and  $\beta_i$  be the angular offset of sensor  $i$ . Then  $r'_i$  is given by:

$$r'_i = r\sqrt{\cos(\theta + \beta_i)} \quad \text{with} \quad \beta_i = -\beta_{-i} \quad (2)$$

Therefore:

$$\begin{aligned} r'^2_{-i} + r'^2_i &= r^2 \cos(\theta - \beta_i) + r^2 \cos(\theta + \beta_i) \\ &= 2r^2 \cos(\theta) \cos(\beta_i) \\ r'^2_{-i} - r'^2_i &= r^2 \cos(\theta - \beta_i) - r^2 \cos(\theta + \beta_i) \\ &= 2r^2 \sin(\theta) \sin(\beta_i) \end{aligned}$$

Let:

$$\begin{aligned} b &= \frac{\sum_{i=1}^n r'^2_{-i} + r'^2_i}{\sum_{i=1}^n 2 \cos(\beta_i)} = r^2 \cos(\theta) \\ c &= \frac{\sum_{i=1}^n r'^2_{-i} - r'^2_i}{\sum_{i=1}^n 2 \sin(\beta_i)} = r^2 \sin(\theta) \end{aligned}$$

so that:

$$\theta = \arctan\left(\frac{c}{b}\right), \quad r = (b^2 + c^2)^{\frac{1}{4}}$$

which exploits the trigonometric identity  $A \cos^2(x) + A \sin^2(x) = A$ .

We can apply this algorithm to our current system. Given the signal strength at all four IR receivers, we can immediately find the quadrant of the bearing of the transmitting robot by taking the strongest received signal ( $r'_1$ ), and the strongest adjacent signal ( $r'_2$ ) (see Fig 2). In a four sensor system,  $\beta_1 = \frac{\pi}{4}$ , and thus:

$$r'_{-1} = r\sqrt{\cos\left(\theta - \frac{\pi}{4}\right)}, \quad r'_1 = r\sqrt{\cos\left(\theta + \frac{\pi}{4}\right)}$$

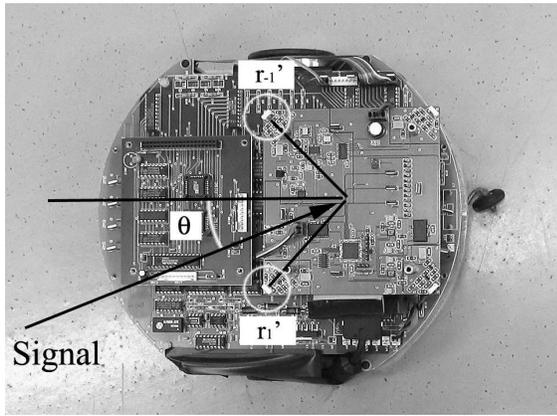


Fig. 2. Localization Module Receiving Signal

Based on this, we get the formulas:

$$\begin{aligned}
 b &= \frac{r_{-1}'^2 + r_1'^2}{2 \cos(\frac{\pi}{4})}, & c &= \frac{r_{-1}'^2 - r_1'^2}{2 \sin(\frac{\pi}{4})} \\
 \theta &= \arctan\left(\frac{r_{-1}'^2 - r_1'^2}{r_{-1}'^2 + r_1'^2}\right), & r &= (r_{-1}'^4 + r_1'^4)^{\frac{1}{4}} \\
 \phi &= \text{quadrant angle} + \theta
 \end{aligned}$$

The localization board is not centered on the Moorebots. This causes a variation in received signal intensity based on the bearing term  $\phi$ . Although the more accurate way to correct for this is to adjust the distance after the non-linear transformation, we can approximate this by:

$$r = (r_{-1}'^4 + r_1'^4)^{\frac{1}{4}} + A \max(r_{-1}', r_1')^2 \cos \phi \quad (3)$$

for some constant  $A$ .

### C. Sensory System Calibration

Sensor readings were taken using one transmitting robot and one receiving robot. The receiving robot was positioned at different ranges and orientations, but with the same side of the transmitting robot always facing it. Ranges were from 30 cm to 300 cm, with a 30 cm step size. Orientations covered a complete rotation, with a 30° step size. 100 readings were taken at every position.

All errors in distance were calculated using linear interpolation between known points of the range term. The positioning of the robots was done using an overhead camera localization system; a “hat” with two white circles of different sizes is attached on top of each robot in order to calculate their position and bearing. The precision of the system is 15 mm accuracy in positioning and 2.6° accuracy in bearing. This is significantly smaller than the step sizes of the measurements.

### D. Results and Discussion

The error for each bearing of the receiving robot over all ranges was calculated. The average standard deviation across all angles was 6.10° and the worst case was 8.37°. The difference between the measured bearing and the actual bearing was 7.00° on average and 17.4° in the worst case

(see Fig 3a). The error for each range of the robots across all receiving robot orientations was calculated. The standard deviation of the range across all angles was minimum 2.77 cm at 30 cm range and maximum 17.70 cm at 300 cm range (see Fig 3b). This is a decrease in error of over 30% for bearing and approximately 25% for range from the original system.

Our relative localization system only exploits the minimum number of sensors used in our algorithmic model. More receivers would increase the accuracy of the offset angle within the sector, as there would be more sensors facing the transmitting robot at any given time. Using a numerical model with the multiple receiver algorithm presented in subsection 2B, with random independently distributed  $\theta$  values and constant Gaussian noise on the receivers (which matched noise observed during calibration), we found that the error in range for this algorithm decreased as  $\frac{1}{\sqrt{n}}$ , where  $n$  is the number of sensors. For the error in bearing, the initial decrease (from 4 to 12 sensors) was slightly less than  $\frac{1}{\sqrt{n}}$ , perhaps due to the non-linearity of the arctan function (see Table I). These predictions assume that sensor noise is uncorrelated, and is not due to ambient light or varying sensor thresholds.

TABLE I  
NORMALIZED ERROR FOR DIFFERENT NUMBERS OF SENSORS

Number of Sensors	Bearing Error	Range Term Error
4	1.00	1.00
8	0.89	0.70
12	0.75	0.56
16	0.66	0.48

Besides causing a small error in range, the localization module not being centered on the robots could cause a large error in bearing if the robots were very close and the orientation of the transmitting robot changes. Since there is no way for the receiving robot to know the orientation of the transmitting robot, the only solution to this problem is to center the module.

We have focused mainly on software optimization in the relative localization system, and not attempted to modify the hardware (including the placement of the module on the Moorebots). There may be room for improvement in how the intensities are measured using adjustable receiver baselines, which could compensate for any variations that might appear in receiver sensitivity. If there were variation in the strengths of the infrared transmitters, the module could be upgraded to allow for adjustable control. It also seems likely that the rate of measurements could be increased significantly.

A similar relative localization system to ours using infrared has been described by McLurkin [17]. This system achieves accuracy of 1 cm in range and 2° in bearing at a range of 30 cm, but has a maximum range of 250 cm, and a smaller, unspecified maximum range that provides reliable positioning. Therefore, although the performance of our system at 30 cm range is not as accurate as that in [17], the usable range is farther. It also does not offer any communication capabilities.

In [1], very accurate relative positioning is accomplished using ultrasound and time-of-flight (TOF). The system achieves

an average bearing error of  $1.84^\circ$  and an average range error of 0.375 cm, with an 810 cm maximum range. However, the test was only performed with two robots, and it remains to be seen how echo effects and interference may deter performance if more agents are introduced to the system, as this may put a low limit on the maximal rate at which the system can operate. The ultrasonic localization system described in [7] suffered from accuracy problems due to these effects.

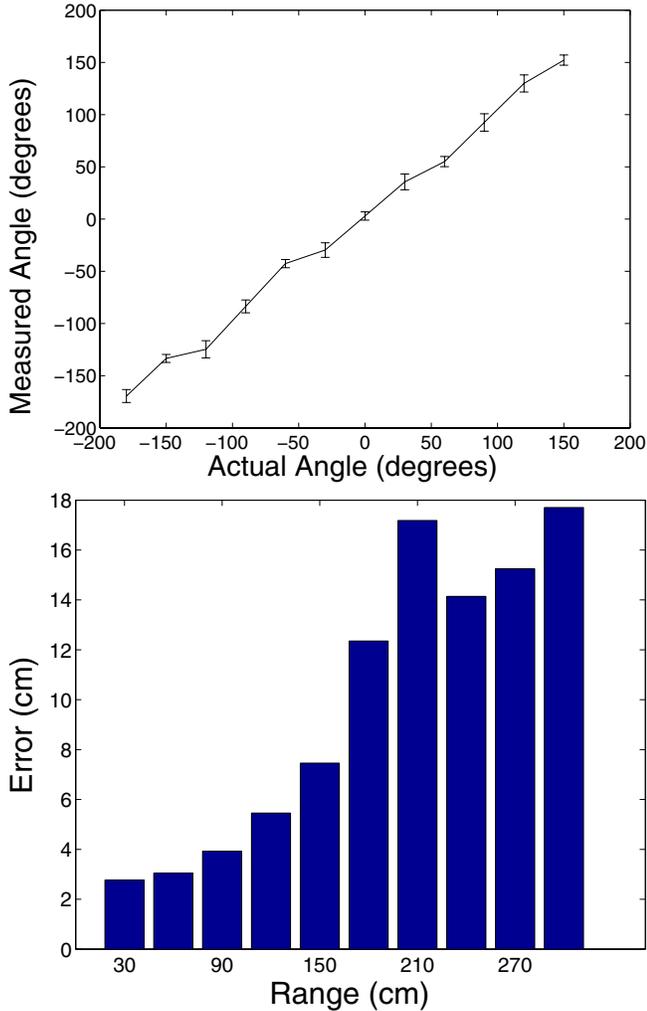


Fig. 3. (a) performance of new algorithm for calculating bearing (error bars represent standard deviation), (b) standard deviation of new algorithm when calculating range

### III. MULTI-ROBOT SELF-LOCALIZATION

We now demonstrate the usefulness of the relative positioning module with a simple multi-robot self-localization task, where robots attempt to improve their position estimates in global coordinates over time. We observe the change in performance from both using the relative positioning module by itself and from using the relative positioning module along with interrobot radio communication. The experiment is implemented both in simulation and with real Moorebots. The agreement between the simulation and the real-world

experiment allows us to predict the outcome if the number of robots and length of the experiment were increased.

#### A. Set-up Overview

We use four Moorebots operating within the arena. Relative positioning is accomplished using the infrared module previously described. Radio communication is done using the IEEE 802.11 wireless protocol in an ad-hoc network containing all the robots as well as the computer monitoring the overhead camera. All communication is done using UDP packets with manual acknowledgements. Although it would have been possible to have used a more reliable communication method (e.g., a managed network with TCP connections), this setup more closely resembles what would be used in a real-world environment where no global access point is available and robots are not always within radio range.

For our simulator, we use Webots [18], with a realistic implementation of the Moorebot. The Webots arena is  $3.4 \times 3.4 \text{ m}^2$  in the same shape as the real arena for 4 or fewer robots, and  $8.0 \times 8.0 \text{ m}^2$  and square shaped for more than 4 robots.

#### B. Experiment Description

Robots follow a common path which connects a set of pre-defined waypoints within the arena. When a robot arrives at a waypoint (according to its odometric estimate), it will pause (30-50 seconds for our experiments), then continue on to the next waypoint in the path. If an obstacle is detected by the proximity sensors, it will move to avoid it. The robot keeps track of its position estimate and its confidence in that estimate over time. This information is regularly sent back to the computer connected to the overhead camera, to allow for monitoring of the error in the robot's position estimation. If a robot is unable to reach its target waypoint after a certain amount of time (approximately 60 seconds), either because its position estimate places the waypoint outside the arena or because another robot is in the way, it gives up and moves on to the next waypoint in the path. Robots use three different modes to track their position estimates.

1) *No Relative Positioning*: Robots do not interact with others to adjust their position estimates during the experiment.

2) *Relative Positioning Without Radio Communication*: When the robot A arrives at a waypoint, it will broadcast a relative positioning message which contains the index of its current point (this allows any receiving robot B to determine from the waypoint list where that robot believes itself to be). While robot A remains at that point, it will update its position estimate using relative positioning messages it receives from any other robot B. The position update is accomplished by weighting the position estimates according to their uncertainty:

$$\hat{X}'_A = \frac{\sigma_{RP}^2 \hat{X}_A}{\sigma_{RP}^2 + \sigma_A^2} + \frac{\sigma_A^2 \hat{X}_{RP}}{\sigma_{RP}^2 + \sigma_A^2}$$

where  $\hat{X}_A$  is the previous position estimate of the robot,  $\hat{X}_{RP}$  is the position estimate from the relative positioning measurement,  $\sigma_A^2$  and  $\sigma_{RP}^2$  are their variances, respectively,

and  $\hat{X}'_A$  is the new position estimate. To calculate  $\sigma_{RP}^2$ , we need to know the certainty in position of the other robot B. As this information is unavailable without radio communication, we assume it to be the same as the certainty of robot A ( $\sigma_A^2$ ). This assumption is not unreasonable, as all robots had identical initial values. For all calculations, we assume the error in position to be independent between robots.

In order to avoid being overly influenced by the position estimation of a single other robot B, a receiving robot A will only update its position once per waypoint of transmitting robots (e.g., only one update from a robot at waypoint 5, but also potential updates from robots at waypoints 1, 2, 3, etc). This prevents robot A from repeatedly updating its position estimate from a single robot B's transmission. This list is reset whenever a robot finishes pausing at a waypoint.

3) *Relative Positioning With Radio Communication:* When robot A arrives at a waypoint, it will broadcast a relative positioning message which contains its unique robot ID number; it will also use its radio to send its position estimate, relative positioning information, and position certainty to the other robots. This is superior to relative positioning without communication in two ways. First, we have two sets of relative positioning data (A's observation of B and B's observation of A) to use to calculate  $\hat{X}_{RP}$ , decreasing the expected error. Second, we now have the proper certainty in position of the other robot ( $\sigma_B^2$ ), making our equation more accurate.

In order to avoid being overly influenced by the position estimation of a single other robot B, a receiving robot A will only update its position once for any other robot B, while A remains at that waypoint.

We used 12 waypoints with 0.6 m spacing as shown in Fig 4a for the real robotic experiments. Robots move clockwise along the path. We used the same configuration for the simulated experiments for 4 or less robots, and a grid of 44 points with a much longer path for experiments with more than 4 robots, as shown in Fig 4b. Robots move at approximately 5 cm/s. Initial robot locations are fixed at designated waypoints to prevent excessive collisions at the start of the experiments.

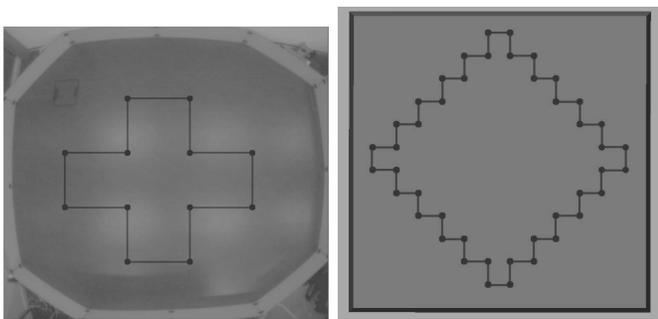


Fig. 4. Arena for (a) real Moorebots, and (b) more than 4 robots in simulation. Black dots represent waypoints, with connecting black lines showing the path Moorebots must follow. All robots move along the path in the same direction.

### C. Results and Discussion

The error in position over time for four real robots and four simulated robots can be seen in Fig. 5. We see the same trend in the real Moorebots as in the realistic simulation. No relative positioning offers the worst performance, while relative positioning has less error, and even less with communication. This indicates that the on-board relative positioning module is indeed helping to increase the accuracy of the robots' self-position estimates. Communication can further increase this improvement, though the amount of improvement is questionable, as the difference is much less in the realistic simulation than with real robots. The good agreement between the simulated and real results validates the accuracy of our simulated environment. The ripples in error are due to the robot pauses, during which time no movement occurs, and hence error does not increase.

To observe the effects of scalability, we ran the algorithm on various numbers of robots in simulation for a longer time period (Fig. 6). As the number of robots increases, we see a large decrease in error for relative positioning, while the error without relative positioning stays about the same. This is as expected, since more robots means more sources of independent position estimates, which should result in a lower combined error.

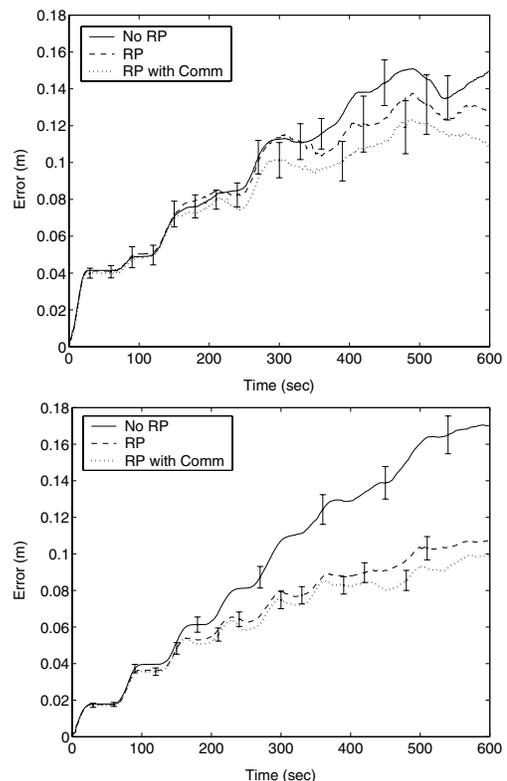


Fig. 5. (a) Average positional error over 20 ten minute runs with 4 real Moorebots for no relative positioning, relative positioning, and relative positioning with communication. (b) Average positional error over 100 ten minute runs with 4 Moorebots in realistic simulation for no relative positioning, relative positioning, and relative positioning with communication. Error bars represent standard error of positional error over runs.

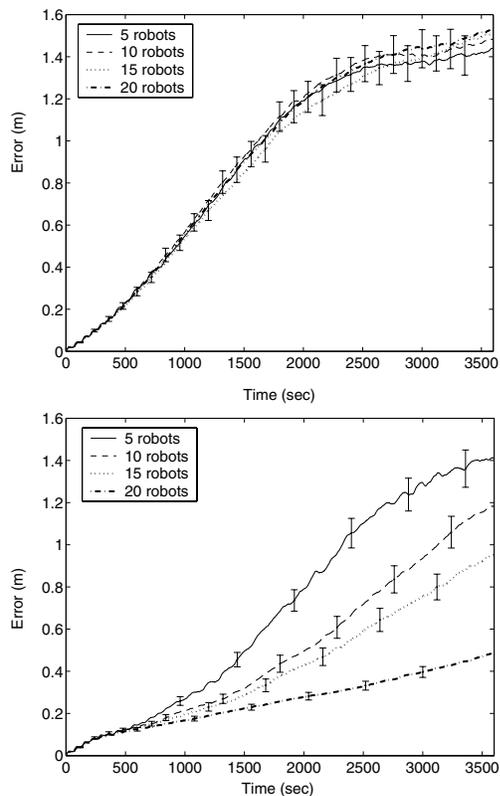


Fig. 6. (a) Average positional error over 100 one hour runs without relative position. (b) Average positional error over 100 one hour runs with relative position without communication. Error bars represent standard error of positional error over runs.

#### IV. CONCLUSION AND OUTLOOK

A new algorithm was developed for calculating the range and bearing with the relative localization system. This algorithm offered significant improvement in performance over the original system from [13]. An accurate model for the relative localization system was created which allows us to predict how the error of the system could be decreased by increasing the number of infrared receivers. The functionality and usefulness of the system were demonstrated by improving the performance of robots engaged in a simple self-localization task using several different methods, and an analysis of the results is presented. Potential for further improvements to the system have been explored.

#### Acknowledgements

Jim Pugh and Alcherio Martinoli are currently sponsored by a Swiss NSF grant (contract Nr. PP002-68647).

#### REFERENCES

- [1] Bisson, J., Michaud, F. & Létourneau, D. (2003) "Relative Positioning of Mobile Robots Using Ultrasounds", IEEE Int. Conf. on Robotics and Automation IROS03, Oct, Las Vegas, Nevada, USA, pp. 1783-1788.
- [2] Correll, N. & Martinoli A. (2004) "Collective Inspection of Regular Structures using a Swarm of Miniature Robots", Proc. of the Ninth Int. Symp. on Experimental Robotics ISER-04, Singapore. Springer Tracts in Advanced Robotics 6, Vol. 21, 2006. To appear.

- [3] Fredslund, J. & Mataric, M. J. (2002) "General Algorithm for Robot Formations Using Local Sensing and Minimal Communication", Special Issue on Advances in Multi-Robot Systems, Arai T., Pagello E., and Parker L. E., editors, IEEE Trans. on Robotics and Automation, Vol. 18, No. 5, pp. 837-846.
- [4] Fox, D. et al. (2000) "General Algorithm for Robot Formations Using Local Sensing and Minimal Communication", Autonomous Robots, Vol. 8, No. 3, pp. 325-344.
- [5] Gage, D. W. (1995) "Many-Robot MCM Search Systems", Proc. of the Autonomous Vehicles in Mine Countermeasures Symposium. Monterey, CA, pp. 4-7.
- [6] Grabowski, R. et al. (2000) "Heterogeneous teams of modular robots for mapping and exploration", Autonomous Robots, Vol. 8, No. 3, pp. 293-308.
- [7] Grabowski, R. & Khosla, P. (2001) "Localization Techniques for a Team of Small Robots", Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, HI, Oct 29-Nov 3, pp. 1067-1072.
- [8] Hayes, A. T., Martinoli, A. & Goodman, R. M. (2000) "Comparing Distributed Exploration Strategies with Simulated and Real Autonomous Robots", Proc. of the 5th International Symposium on Distributed Autonomous Robotic Systems DARS'00. Knoxville, TN, pp. 261-270.
- [9] Hayes, A. T., Martinoli, A. & Goodman, R. M. (2002) "Distributed Odor Source Localisation", Special Issue on Artificial Olfaction, Nagle H. T., Gardner J. W., and Persaud K., editors, IEEE Sensors Journal, Vol. 2, No. 3, pp. 260-271.
- [10] Hayes, A. T. (2002) "How Many Robots? Group Size and Efficiency in Collective Search Tasks", Proc. of the 6th International Symposium on Distributed Autonomous Robotic Systems DARS'02. Fukuoka, Japan, pp. 289-298.
- [11] Howard, A., Mataric, M. J., & Sukhatme, G. S. (2003) "Localization for Mobile Robot Teams: A Distributed MLE Approach", Springer Tracts in Advanced Robotics, Vol. 5, Jan 2003, pp. 146-155
- [12] Howard, A., Mataric, M. J., & Sukhatme, G. S. (2003) "Putting the 'I' in 'Team': an Ego-Centric Approach to Cooperative Localization", Proc. of the IEEE International Conference on Robotics and Automation, Taipei, Taiwan, Sept 14-19, pp. 868-874.
- [13] Kelly, I. D. & Martinoli, A. (2004) "A Scalable, On-Board Localisation and Communication System for Indoor Multi-Robot Experiments", Special Issue on Sensor Simulation and Smart Sensors, C. Loughlin, editor, Sensor Review, Vol. 24, No. 2, pp. 167-180.
- [14] Kurazume, R. & Nagata, S. (1994) "Cooperative positioning with multiple robots", Proc. of the IEEE International Conference on Robotics and Automation, Los Alamitos, CA, May 8-13, pp. 1250-1257.
- [15] Loh, J. et al. (1998) "Behavior-based Search using Small Autonomous Mobile Robot Vehicles", Proc. of Second International Conference on Knowledge-Based Intelligent Electronic Systems, Adelaide, Australia, pp. 21-23.
- [16] Martinoli, A. (1999) "Swarm Intelligence in Autonomous Collective Robotics: From Tools to the Analysis and Synthesis of Distributed Collective Strategies", Ph.D. Thesis Nr. 2069, October, 1999, DI-EPFL, Lausanne, Switzerland. Downloadable from <http://swis.epfl.ch/people/alcherio>.
- [17] McLurkin, J. & Smith, J. (2004) "Distributed Algorithms for Dispersion in Indoor Environments using a Swarm of Autonomous Mobile Robots", Proc. of the Seventh Int. Symp. on Distributed Autonomous Robotic Systems DARS04, June 2004, Toulouse, France. To appear in Distributed Autonomous Robotic Systems 5 (2006), Alami R., Asama H., and Chatila R., editors.
- [18] Michel, O. (2004) "Webots: Professional Mobile Robot Simulator", Int. J. of Advanced Robotic Systems, Vol. 1, pp. 39-42.
- [19] Rekleitis, I., Dudek, G. & Milios, E. (2001) "Multi-robot collaboration for robust exploration", Annals of Mathematics and Artificial Intelligence, Vol. 31, pp. 7-40.
- [20] Roumeliotis, S. I. & Bekey, G. A. (2002) "Distributed Multirobot Localization", IEEE Transactions on Robotics and Automation, Vol. 18, No. 5, pp. 781-795.
- [21] Wessnitzer, J. & Melhuish, C. (2003) "Collective Decision-Making and Behaviour Transitions in Distributed Ad Hoc Wireless Networks of Mobile Robots: Target-Hunting", Advances in Artificial Life ECAL 2003, Springer, pp. 893-902.
- [22] Winfield, A.F.T. & Holland, O.E. (2000) "The application of wireless local area network technology to the control of mobile robots", Microprocessors and Microsystems, Vol. 23, pp. 597-607.