

Comparing Distributed Exploration Strategies with Simulated and Real Autonomous Robots

Adam T. Hayes, Alcherio Martinoli, and Rodney M. Goodman

Microsystems Lab 136-93, California Institute of Technology, Pasadena CA 91125
athayes,alcherio,rogo@micro.caltech.edu
<http://www.micro.caltech.edu/micro/CORO/index.html>

Keywords: Collective Autonomous Robotics, Probabilistic Modeling, Embodied Simulation, Distributed Exploration

Abstract. This paper presents an experiment in collective robotics in which a group of autonomous robots searches for an infrared target beacon placed in a corner of the exploration area. This task is a more experimentally tractable version of the plume tracing problem, in which robots search for the source of an odor plume. Two different exploration strategies (collaborative and non-collaborative) are implemented and compared on the basis of several team performance metrics. The collaborative strategy uses a simple, binary signaling schema among robots. The experiment is implemented at three different levels: in a physical setup composed of groups of 1 to 8 Moorebot robots, in Webots, a 3D sensor-based, kinematic simulator, and with probabilistic simulations. Results show that the collaborative approach drastically improves the search across several metrics. Furthermore, the probabilistic model qualitatively and quantitatively reproduces the enhanced team performance via collaboration. Additional investigations using the probabilistic model indicate that the optimal number of robots is a function of the ratio between target and exploration areas.

1 Introduction

The application of swarm intelligence principles to autonomous collective robotics aims to develop robust task solving by minimizing the complexity of the individual units [1]. The central idea of the swarm intelligence approach is to distribute control over a group of numerous minimalist robots rather than gathering and redistributing information with the help of a central unit. The main advantages of this approach are two: first, scalability from a few to thousands of units, second, increased system robustness, not only through unit redundancy but also through the unit minimalistic design. Several examples of collective robotics tasks solved with swarm intelligence principles can be found in the literature: aggregation [2,3] and segregation [4], exploration [5], stick pulling [6], and collective transportation [7,8].

One way to increase the performance of a robot swarm is collaboration. In particular, if collaboration is obtained with simple explicit communication schemes such as binary signaling, the team performance can be enhanced

without losing autonomy or significantly increasing the complexity at the individual level. The experiments presented in this paper deal with the implementation of distributed exploration strategies for a group of autonomous robots, and compare the effectiveness of the collaborative approach vs. the non-collaborative one. The beacon finding task is meant to address some of the issues present in the more complex task of plume tracing, without having to deal with the experimental complications of creating repeatable chemical plume stimuli.

The aim of this case study is three-fold. Firstly, we want to show that collaboration can drastically improve the team performance. Secondly, we want to show that Webots [9], a 3D sensor-based, kinematic simulator originally developed for Khepera robots [10], can be successfully used for simulating experiments with other, kinematically similar robots. Thirdly, the presented experiments are used as a testbed for the development of a new characterization and prediction tool in the form of a probabilistic model, capable of wider application across tasks and environments than previous models [3,5].

2 Non-Collaborative Collective Exploration

2.1 Experimental Setup with Real Robots

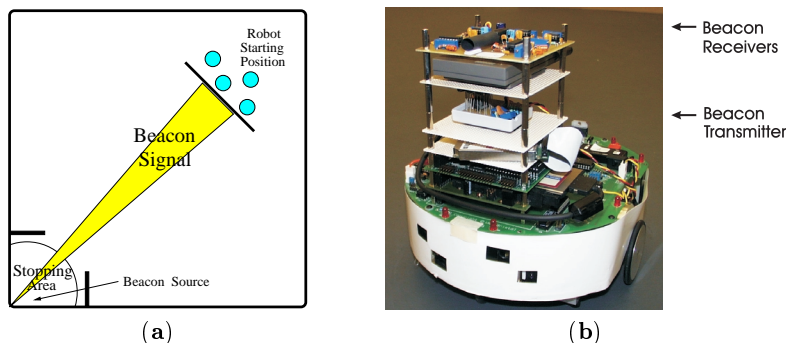


Fig. 1. (a) The arena layout. (b) A Moorebot

The layout of the test arena is shown in Fig. 1a. The size is 6.7 by 6.7 m, and the robots are 24 cm in diameter. The robots begin each trial in the corner of the arena opposite the stopping area. They are started simultaneously, at random positions and orientations within the starting area, and the time required for each to enter the stopping area (as defined by a separate beacon signal) is recorded.

We use Moorebots, as shown in Fig. 1b. In addition to the standard configuration, as described in [11], each robot is equipped with an infrared

transmitter and three binary infrared receivers (for determining teammate communication, the target beacon, and the stopping area). All explicit inter-robot communication is performed via the single binary channel.

When exploring the arena, the robots move in straight lines, avoiding walls and teammates with random turns between 0 and 270 degrees. If a signal is detected, the robots move directly toward the source beacon, although obstacle avoidance takes precedence when necessary. In the non-collaborative case, the only beacon signals present are the target and the stopping beacons.

2.2 Sensor-Based Simulations

To more systematically investigate the exploration strategies, we also implemented this experiment in Webots, a 3D kinematic simulator originally designed to simulate Khepera robots. Because of the morphological similarities between the two platforms, it is possible to scale environment, robot size, and speed to reflect the relationships of the real robot experiments. In addition to size, the main morphological differences between the Khepera and Moorebot platforms are the position and number of proximity sensors. Moorebot sensor outputs are generated by combining and thresholding the outputs of the Khepera’s four front and central built-in proximity sensors, as shown in Fig. 2a. With the creation of a Moorebot-to-Khepera translation library, control algorithms can be transferred from the Moorebots to the simulator with very few changes.

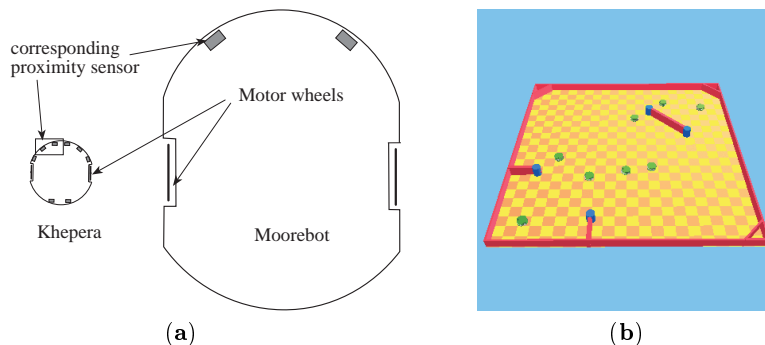


Fig. 2. (a) Khepera and Moorebot sensor layouts. (b) Implementation of the experiment in Webots

The simulator computes trajectories and sensory input of the robots in an arena corresponding to the physical set-up (Fig. 2b). The simulation is sufficiently faithful for the controllers to be transferred from the simulation to real robots and achieve very similar performance (see Section 4). Working in simulation not only increases the experimental reliability but also saves

time. For this experiment with 8 robots, the mean acceleration ratio between Webots and real time is about 14 on a Pentium III 500.

2.3 The Probabilistic Model

In an effort to gain an understanding of the essential aspects of the system, as well as to further decrease simulation time, we have developed a probabilistic model which can generate a robot arrival time distribution for a given set of experimental parameters (e.g. arena size, robot size, robot speed, avoidance time, communication range, and number of robots). In order for this model to be useful, it has to produce data similar to that observed in the embodied experiments, predict results of future experiments, and contain no free parameters. All parameters in this model are determined from measurements taken from the arena and experiments with a single robot.

The basic idea behind the model is similar to that of [3,5], where the experiment is treated as a series of stochastic events with probabilities based on geometrical relationships. However, because this experiment depends heavily on the spatial location of the individual robots (their distribution cannot be assumed to be uniform across the arena, as was the case in [3,5]), a different method must be used. To capture spatial information, the arena is divided into four distinct states, as shown in Fig. 3a. At any given point in time each robot occupies one state. As the arena is partitioned with greater granularity, more information is included in the simulator, but more computation is required to complete each simulation. Since this model should be as simple and as fast as possible, the minimum number of states for accurate data generation are used. As a result, it is about 15,000 times faster than Webots.

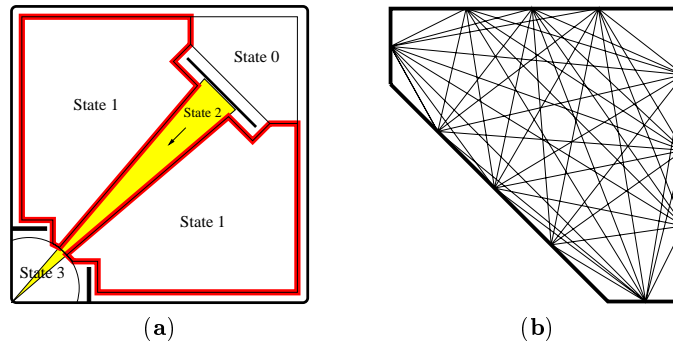


Fig. 3. (a) Arena partitioning in the non-collaborative experiment. The dark bands represent the distribution of locations occupied by any robot entering State 1. (b) Example set of chords in State 0

The transition probabilities between states in which the robots move randomly (States 0 and 1) are calculated from the perimeters of each state, which are offset from the physical perimeters of the arena to account for the spatial extent of the robots. Each time a robot encounters the outer edge of a state, it can transition into another state or bounce off a wall back into its current state (with an avoidance time penalty corresponding to the average avoidance time of a real robot). Collisions are also modeled as an avoidance time penalty, applied stochastically based on state size, effective robot area, and number of robots currently in a given state.

For states in which the robots move in a particular direction (State 2, when a robot sees a beacon signal), the expected time before transitioning into the next state was taken to be the average distance to the next state divided by the robot speed. This assumes that robots never lose an acquired signal, which does not entirely capture the behavior of the real robots, but is a close approximation in this case. In principle, data could be collected from a real robot and this probability of signal loss incorporated into the model.

A difficulty with the probabilistic model is the generation of the perimeter encounter probability. Based on the average robot speed and the average distance across the state (average chord length - see Fig. 3b), the expected time to perimeter encounter can be generated. Treating the system as a simple Markov process would result in a geometric time-to-perimeter-encounter distribution (with a single parameter p), as at each time step the probability of encountering the perimeter would be a constant value, as given by:

$$p = v/m \tag{1}$$

where v is the robot speed and m is the average chord length. This distribution does not accurately reflect the spatial realities of the embodied systems, as many times are unreasonably short, and a few are unreasonably long (for instance, if a robot moves at 1 m/s, and an area has a maximum chord length of 10 m, it cannot go for more than 10 seconds without encountering something). Despite the fact that the expected value of the distribution is accurate, overall robot arrival times have bigger variances than those observed in the embodied experiments.

To account for this defect, a hybrid distribution is used— basically a geometric distribution in which the parameter p is a function of the amount of time a robot has been in a given state. If the maximum chord length of a state is l , robots must not be able to wait longer than $l/v = k$ seconds between encounters, so $p(k)$ should be set to 1. This suggests an equation of the form:

$$p(x) = x/k \tag{2}$$

In order that the expected value of the distribution can be set independently of this cutoff time, two parameters are needed:

$$p(x) = \frac{x^\alpha}{A} \tag{3}$$

The form of the resulting probability density function is quite similar to that of the geometric distribution:

$$P(x) = p(x) * \prod_{i=1}^{x-1} (1 - p(i)) \quad (4)$$

For the experiments in this paper, the values of α and A are set numerically based on geometrical considerations, as further analysis of this distribution is ongoing. See Fig. 4a to compare the transition distributions from State 0 to State 1 generated by Webots, this hybrid process, and a simple Markov process.

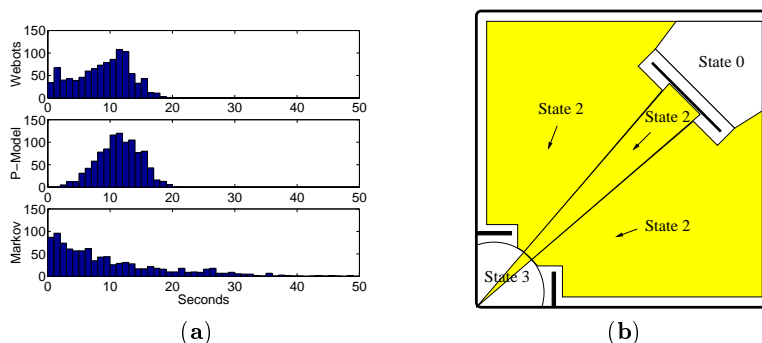


Fig. 4. (a) Histogram of the number of encounters as a function of time-to-perimeter for State 0, $N = 1000$. (b) Arena partitioning during collaboration

Note that the Webots data in Fig. 4a includes repeated encounters with the same wall, due to the nature of the collision sensors and avoidance behavior, and this increases the number of small waiting times. Such encounters are not included in the probabilistic model because they are a result of spatial correlation between successive encounters which are not captured within the probabilistic framework. Even though the hybrid distribution generates perimeter encounter data which are closer to the target distribution, it cannot be truly accurate because the model does not incorporate exact robot trajectories.

The lack of complete spatial information also influences the state transition probabilities in a way that is difficult to capture in the model. For example, the location of a robot that has just transitioned into State 1 is represented by a probability distribution at the perimeter of the state, as shown by the dark band in Fig. 3a. Thus it is possible for a robot in the model to transition from State 0 to State 3 more quickly than if it had to follow a real trajectory (as is necessary in the embodied simulator and the real world). To be truly accurate, the state transition probability (which in

the model is based on static perimeter ratios) must change with the entrance point and amount of time spent within the state. Calculating the nature of these changes would be complex, and incorporating them into the model would restrict simulation speed. Instead, we model the position dependent effects as a constant time penalty added to the simulated trials (roughly counteracting the fact that robots in the model can change states too fast). Since the behavior of one robot captures the spatial dependencies of the task, we take the time penalty to be the difference between the mean arrival time for one real robot and one uncompensated simulated robot (in this case, 31.5 seconds).

3 Collaborative Collective Exploration

3.1 Experimental Setup with Real Robots

For the collaborative experiments, when a robot sees either the target beacon or the stop beacon, it turns on a beacon of its own, thus attracting surrounding robots. The range of the robot beacons generally extends across the arena, but due to slight hardware differences (e.g. LED orientation), the communication range is not uniform across all robot pairs. Because these heterogeneities are difficult to quantify, they are not included in either of the simulations. Other than the robot beacon activity, the collaborative and non-collaborative trials are exactly the same.

3.2 Sensor-Based Simulations

Beacon signaling was implemented via the external supervisor module of Webots. The supervisor module is based on an External Authoring Interface API similar to the one implemented in VRML97 for enabling Java applets to communicate with VRML 97 scenes within a Web browser. This module can read and edit the world loaded in the simulator, including some customized robot features such as virtual signaling beacons.

For the sake of simplicity, the range of the robot beacons was assumed to be sufficient for communication across the entire arena.

3.3 The Probabilistic Model

Collaboration is modeled by reorganizing the distribution of states whenever a robot beacon becomes active (see Fig. 4b). State 1 does not appear because it is assumed that as soon as a robot leaves State 0, it sees a beacon and begins directed motion toward the final state. State 0 slightly changes shape because beacon signals become visible in some areas, so those regions join State 2. Note this reorganization does not wholly capture the behavior of the embodied robots, but it provides a close approximation because it captures the principle features of the collaborative system.

4 Results

4.1 Exploration with collaborative and non-collaborative strategies

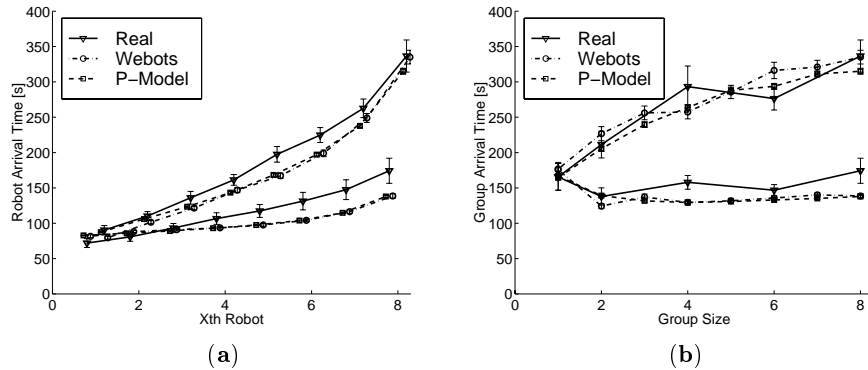


Fig. 5. Number of trials per experiment: real=15, Webots=100, p-model=500. (a) Arrival times for a group size of 8 robots. The lower set of curves is the collaborative data. (b) Group arrival times for groups of 1 to 8 robots.

Figure 5a shows arrival times for both non-collaborative and collaborative trials generated by all three experimental levels for a group size of 8 robots. Figure 5b shows the group arrival times for group sizes from 1 to 8 robots. Figure 6a gives the average power consumption per robot for both types of trials. All error bars represent standard error. Note the good qualitative and quantitative agreement among the different experimental levels.

4.2 Increasing the ratio between exploration and goal area

The benefit of using an increasing number of collaborating robots is not clear from the above experiments because performance saturates at a small group size. As shown in Fig. 5b, the 3 point running average of group arrival time in the collaborative case shows less than 1% improvement at group sizes larger than 3 (Webots) or 4 (p-model) robots. Further simulations were performed both in Webots and with the probabilistic model in a larger environment, 20.1 m per side (9 times larger area), but with the start, stop, and target beacon areas kept the same. Figure 6b shows that even in a different environment the probabilistic model shows good correspondence with Webots. Also, in this larger arena, performance (using the same definition as above) saturates at increased group sizes: 7 (Webots) or 9 (p-model). Based on the interaction between increasing search speed and increasing inter-robot interference, we

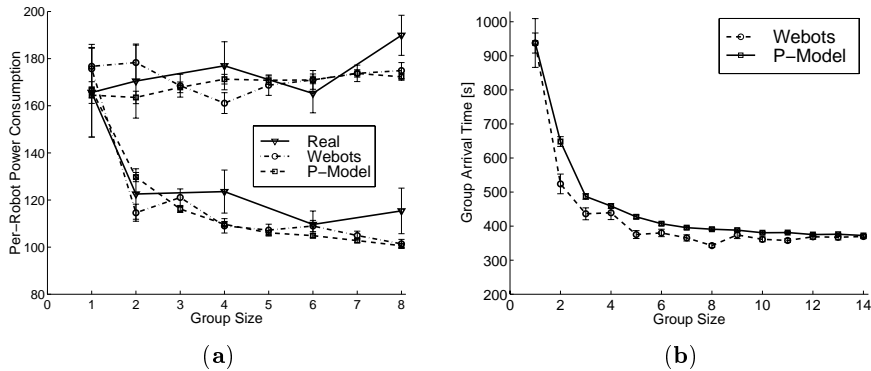


Fig. 6. (a) Per-robot power consumption for groups of 1 to 8 non-collaborating (upper curve) and collaborating (lower curve) robots (same trials as Fig. 5). (b) Group arrival times for groups of 1 to 14 robots searching a larger arena. Number of trials per experiment: Webots=100, p-model=500.

conjecture that the optimum group size is dependent upon the relationship among the target area, exploration area, and the communication range.

5 Discussion and Conclusion

In this paper, we investigated collaborative and non-collaborative exploration strategies for a group of autonomous robots at three different experimental levels: real robots, sensor-based simulations, and probabilistic simulations. The data shows that simple collaboration among robots drastically improves the team performance by reducing the arrival time of the group as well as the group power consumption. Collaborative group performance saturates as the team size grows, and the smaller the ratio between target area and exploration area or the signaling range, the greater the optimal team size.

Furthermore, we have shown that the results obtained with a zero-free parameter probabilistic model are qualitatively and quantitatively in good agreement with those obtained from more sophisticated sensor-based simulations and real robots.

Considering the tremendous acceleration ratio between the probabilistic model and the other implementations, we believe that it represents an extremely useful tool for capturing the role of the crucial experimental parameters in distributed exploration experiments. This could be particularly useful for investigating team performances with thousands of units before actually building them or developing a dedicated sensor-based simulator for the given task. It is unknown how the model will perform on more complex environments and tasks (such as plume tracing), and this is a subject of ongoing research.

6 Acknowledgments

We would like to thank Owen Holland, Sanza Kazadi, and Andrew Lundsten for their work on the Moorebot platform. This work is supported in part by the Center for Neuromorphic Systems Engineering as part of the National Science Foundation Engineering Research Center Program under grant EEC-9402726. Other support in part by DARPA under grant DAAK60-97-K-9503, and the Office of Naval Research under grant N00014-98-1-0821 in the Chemical Plume Tracing Program. We also acknowledge support from the DARPA program in Distributed Robotics. Adam Hayes is supported by a National Science Foundation Graduate Research Fellowship.

References

1. E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, US, 1999.
2. R. Beckers, O.E. Holland, and J.L. Deneubourg. From local actions to global tasks: Stigmergy and collective robotics. In R. Brooks and P. Maes, editors, *Proc. of the Fourth Workshop on Artificial Life*, pages 181–189, Boston, MA, 1994. MIT Press.
3. A. Martinoli, A. J. Ijspeert, and F. Mondada. Understanding collective aggregation mechanisms: From probabilistic modelling to experiments with real robots. *Robotic and Autonomous Systems*, 29:51–63, 1999.
4. O.E. Holland and C. Melhuish. Stigmergy, self-organization, and sorting in collective robotics. *Artificial Life*, 5:173–202, 1999.
5. A. Billard, A. J. Ijspeert, and A. Martinoli. A multi-robot system for adaptive exploration of a fast changing environment: Probabilistic modelling and experimental study. *Connection Science*, 11:359–379, 1999.
6. A. Martinoli and F. Mondada. Collective and cooperative group behaviours: Biologically inspired experiments in robotics. In O. Khatib and J. K. Salisbury, editors, *Proc. of the Fourth International Symposium on Experimental Robotics ISER-95*, pages 3–10, Stanford, U.S.A., June 1995. Springer Verlag.
7. M. J. B. Krieger and J.-B. Billeter. The call of duty: Self-organised task allocation in a population of up to twelve mobile robots. *Robotics and Autonomous Systems*, 30:65–84, 2000.
8. C. R. Kube and E. Bonabeau. Cooperative transport by ants and robots. *Robotics and Autonomous Systems*, 30:85–101, 2000.
9. O. Michel. Webots: Symbiosis between virtual and real mobile robots. In *Proceedings of the First International Conference on Virtual Worlds, VW'98*, pages 254–263, Paris, France, July 1998. Springer Verlag.
10. F. Mondada, E. Franzi, and P. Ienne. Mobile robot miniaturization: A tool for investigation in control algorithms. In T. Yoshikawa and F. Miyazaki, editors, *Proc. of the Third International Symposium on Experimental Robotics ISER-93*, pages 501–513, Kyoto, Japan, 1993. Springer Verlag.
11. A.F.T. Winfield and O.E. Holland. The application of wireless local area network technology to the control of mobile robots. *Microprocessors and Microsystems*, 23:597–607, 2000.