The impact of localization errors on the performance of the Ants exploration algorithm

Ettore Ferranti Niki Trigoni
Computing Laboratory
University of Oxford, Oxford, UK
{Ettore.Ferranti|Niki.Trigoni}@comlab.ox.ac.uk

ABSTRACT

When an emergency occurs within a building, it is safer to send autonomous mobile agents instead of human responders, to explore the area and identify hazards and victims. Existing exploration algorithms [11, 4] allow mobile agents to make distributed navigation decisions by communicating with nearby fixed sensors embedded in the environment. These algorithms are very efficient in terms of exploration time, but they have mainly been evaluated in simulation environments, where idealized assumptions were made regarding the ability of mobile agents to detect and localize fixed sensors in their vicinity. To address this problem, recent work [3] has focused on practical mechanisms for detecting and localizing sensors, implemented them in a real testbed, and derived realistic models of localization errors.

The objective of this work is to investigate the impact of these realistic errors [3] on the performance of the Ants exploration algorithm [11]. In particular, we simulate the performance of Ants with and without realistic errors, and show that introducing small errors can have a significant effect on the total exploration time.

Categories and Subject Descriptors

I.2.9 [**Computing Methodologies**]: Artificial Intelligence—*Autonomous vehicles*

General Terms

Design, Measurement, Experimentation

Keywords

Sensors, Autonomous Agents, Robots

1. INTRODUCTION

When an emergency occurs within a building, the area is typically off-limits for anyone not wearing garments to protect themselves from exposure to hazards. In such adverse conditions, it is safer to deploy a group of autonomous robots, (*mobile agents*) to explore the area as fast as possible. Agents should overcome three important limitations: 1) lack of location information in indoor environments; 2) lack of direct connectivity between agents and 3) lack of map information. In order to address these challenges, recent work has proposed instrumenting the emergency area with tiny fixed sensors [11, 4]. By using the instrumented environment, mobile agents are able to explore the environment without map or location information, and to communicate with each other indirectly by using the sensors to leave and retrieve messages.

For simplicity, consider an area instrumented with fixed sensors lying in a grid topology. Wall cells, i.e. cells that are occupied by some obstacle, are the only ones without fixed sensors. We assume that a mobile agent is able to communicate with the fixed sensor on the current cell, as well as with at most eight fixed sensors in the surrounding cells. We also assume that the mobile agent is able to detect hazards and victims within the current cell. Exploration algorithms that use the above model [11, 4] typically follow four steps: 1) Sensor localisation: the mobile agent identifies the fixed sensors lying in the current and eight surrounding cells; 2) Sensor querying: the mobile agent queries the state of the previously localized sensors; 3) Sensor updating: the mobile agent updates the state of the fixed sensor in the current cell; 4) Navigation: the mobile agent selects one of the surrounding fixed sensors and navigates towards it. Note that exploration decisions are made in a completely distributed manner, by simply relying on the local state of the instrumented environment.

The weakness of previous studies [11, 4] is that they have only focused on the sensor tasking and marking steps, and have largely ignored the practical issues pertaining to sensor localization and navigation. They make unrealistic assumptions about the ability of an agent to accurately localize sensors in its vicinity, and move towards a selected sensor without odometry errors. In order to address these challenges, recent work [3] has proposed realistic localization and odometry error models based on experiments in a real testbed. The objective of this paper is to investigate the effects of applying the proposed error models [3] to the Ants exploration algorithm [11]. In particular, we integrate the error models into an existing simulation environment, and assess how the performance of Ants degrades as a result of introducing realistic errors.

The paper is organized as follows: Section 2 provides an overview of existing localization techniques, and summarizes the error models derived from applying one of them in a real testbed. Section 3 briefly describes the Ants algorithm, which is one of the most popular and simple approaches to exploring a sensor-instrumented environment. Section 4 assesses the performance of the Ants exploration algorithm in a simulation environment with and without realistic errors.

2. BACKGROUND

In this section, we first give an overview of existing technologies for localizing sensor nodes. We then focus on a practical localization technique in which mobile agents equipped with cameras detect fixed sensors lying in their vicinity and localize them [3]. We provide a summary of detection and localization errors reported in [3], which are based on experiments run in a real testbed.

2.1 Localization technologies

Radio Signals: Radio signal strength is a not reliable way of identifying the robot relative position with respect to tags deployed in an environment. In fact, it heavily depends on factors like the relative orientation of the deployed motes, their height from the floor, the material of the floor, and the obstacles in the environment. Batalin et al. [1] create an algorithm called Adaptive Delta Percent, which takes into account the signal strength of the messages received from the various tags while the robot is moving in order to guide it toward one of them. A strong limitation of this approach is that the authors consider an experiment to be successful if the robot is able to reach a tag in the environment within a distance of 3m, an accuracy which is unreasonable for our scenario.

Infrared Signals: Several systems have been created to define mobile robot localisation in indoor environments. Some of them use ultrasonic and infrared technologies simultaneously [5], others radio frequency (RF) and infrared together [7], and some just infrared techniques [8]. However, infrared signals are not completely suitable for our scenario because they have a particularly limited transmission range (i.e. ∼20-30cm), thus the robot risks not being able to identify the deployed tag if the dimension of the cell is bigger than the allowed range. Moreover, interference from the IR component of other light sources could compromise the localisation process [6].

Ultrasonic Signals: Ultrasonic sensors [9] alone could be used to avoid obstacles, but not to identify specific tags in the environment due to the poor resolution of their readings. Therefore, we argue that IR or sonar are not suitable technologies for localizing sensors around an agent (avoiding *localisation errors*), or for guiding the agent to one of the sensors *odometry errors.*

Cameras and image processing: Since the previous approaches are not suitable for our scenario, we decided to explore sensor localisation using camera technologies. Several approaches investigated this area adopting feature cluster recognition [2]. In particular, some of them use image processing techniques to recognize landmarks in the environment [10]. However, most of the approaches are very sophisticated, and cannot run in resource-constrained mobile agents. A simple approach to localizing sensor nodes using cameras is proposed in [3]. In the next subsection, we summarize the error model derived by applying this approach in a real testbed.

2.2 Localization errors

In previous work [3], we proposed practical techniques that allow agents to use their on-board camera to localize sensors lying in their vicinity. In this section, we summarize the localization errors that were observed when we applied these techniques in a real testbed. Our system consisted of three different platforms: 1) mobile agent: Surveyor SRV-1 robot connected with a Tmote Sky mote; 2) fixed sensor: Tmote Sky mote with external bright LED and 3) gateway: laptop connected with a Tmote Sky mote (via its USB interface) used primarily for visualisation of experimental results. The sensors were deployed on the ground in a grid topology as shown in Figure 1, and the agent was placed in the middle of the central cell. The size of each cell was set to 48 cm.

The main results regarding detection and localization errors, are reported in [3], and summarized below: The percentage of undetected sensors, due to adverse light conditions, is not negligible and amounts to 5.56% of all sensors. Sensors that are correctly detected are then localized relative to the position of the mobile agent. Figure 1 shows estimated (circles) and real (squares) positions of sensors surrounding a given agent. In this case one can notice how, even if the sensors were not always correctly localised, the errors are always small enough, so that a sensor can not be thought to be in another cell from its own.

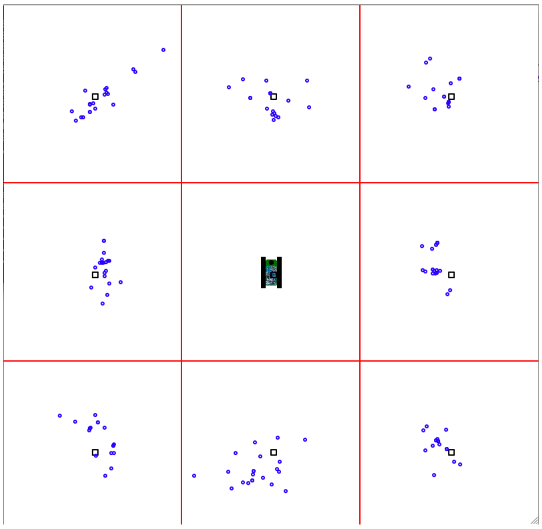

Figure 1: Localisation of sensors around a mobile agent.

3. THE ANTS ALGORITHM

In this section, we briefly describe the Ants algorithm proposed by Svennebring and Koenig in [11]. This is a distributed algorithm that simulates a colony of ants leaving pheromone traces as they move in their environment. Initially, all cells are marked with value 0 to denote that they are unexplored. At each step, an agent reads the values of the four cells around it and chooses to step onto the least traversed cell (the one with the minimum value). Before moving there, it updates the value of the current cell, for example by incrementing its value by one. The authors discuss a few other rules that could be used instead to mark a cell and navigate to the next one, but they all exhibit similar performance in terms of exploration time. Hence, we select the above variant of the Ants algorithm (move to the least visited cell) as a basis for comparison. The authors provide a proof that the agents will eventually cover the entire terrain (provided that it is not disconnected by wall cells).

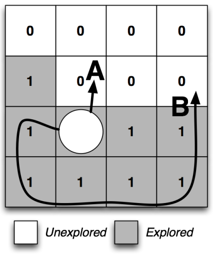

Figure 2: Impact of a small error.

The first advantage of the algorithm is its simplicity: agents do not require memory or radio communication, but only one-cell lookahead. Since they are easy to build, many of them can be used to shorten the coverage process. Secondly, there is no map stored inside the agents: if one of them is relocated (accidentally or on purpose) it will not even realise it and it will continue to do its work as if nothing happened. This means that the whole system is flexible and fault tolerant, and the area can be covered even if some markings or agents are lost. At the sensor of each cell, we only need to store an integer counting the number of times that agents have visited the cell. When the number of times exceeds a threshold, the counter is reset to 0.

The main limitation of the Ants algorithm is that agents do not know when the exploration is terminated, and they continue the exploration phase until they run out of energy. Thus, this approach is not suitable in an emergency scenario, in which the primary consideration is to cover the overall area as soon as possible, and be notified immediately after the task is completed. A further drawback of the algorithm is the limited collaboration among agents. For example, in scenarios with many rooms most of the agents tend to sweep the first few rooms repeatedly, while only a few of them venture to explore new areas.

4. EVALUATION OF ANTS

In this section we illustrate how localization errors impact the behavior of the Ants algorithm and evaluate the algorithm's performance, with and without errors, in a variety of scenarios. In Section 2, we summarized two types of localization errors reported in previous work [3]: 1) Agents tend to introduce small errors in the locations of sensors they identify in their vicinity; these errors are not big enough to impact the behavior of Ants. The reason is that agents see sensors in slightly different locations, but in the correct cells where sensors are actually placed. 2) Agents sometimes completely fail to identify some of the sensors in their vicinity - this type of localization error is referred to as sensor detection error. Although the percentage of missed sensors is reported to be low (5.56%), it significantly affects the performance of Ants. This is illustrated via an example, and quantitatively measured with simulation experiments.

Figure 2 shows the impact of a sensor detection error. In the absence of errors, the agent at the center of the area would choose to explore the cell immediately north of it, by choosing path A. But if it wrongly believes that the cell north of it is not occupied by a sensor and thus is an obstacle or a wall, it will choose to explore first the cell on its left, by choosing path B. This error will bring the agent away from the main front of exploration, causing it to follow a long path of already explored cells, before it can get back to exploring new cells. Note that path B, which is the effect of one detection error, is seven times longer than the regular path A that would be followed in the absence of errors.

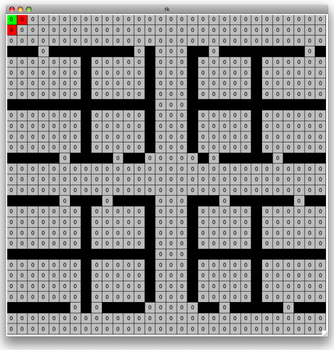

Figure 3: Example of area used during the simulations.

Our next step is to quantify the impact of sensor detection errors on the performance of Ants in a variety of scenarios. To this end, we simulated the Ants algorithm with and without errors and ran a number of simulations varying the number of agents, the size of the area and the number of rooms. The simulations were performed on automatically generated environments representing office-like scenarios (see Figure 3 for an example) with a default area size of 30x30 cells and 4x4 rooms in them. The positions of doors and walls were changed randomly during the experiments, while the default number of agents was 20. For each experiment, we computed the total time necessary to explore the whole area (every cell was traversed at least once by one of the agents). Each point of the graph is the average time of 100 different runs, and is plotted with the corresponding standard deviation bar.

Figure 4 shows the performance of Ants with and without localization errors as we increase the number of agents. Note that with one agent, sensor detection errors actually double the exploration time. Increasing the number of agents helps in reducing the negative effect of these errors, but even with the maximum number of agents the difference remains noticeable. These findings show that simplifying assumptions about the ability of agents to perfectly detect sensors in their vicinity lead to results that are very different from reality.

Figure 5 shows the performance of Ants with and without sensor detection errors as we increase the size of the area. Observe that the negative impact of these errors becomes more pronounced in larger areas. We believe that this is due to the fact that in large areas, a sensor detection error can lead agents to take much long detour paths before returning to the exploration front.

Figure 6 shows the impact of sensor detection errors as we increase the number of rooms, whilst keeping the size of the area constant. Observe that the impact of these errors decreases as we increase the number of rooms. This is due to

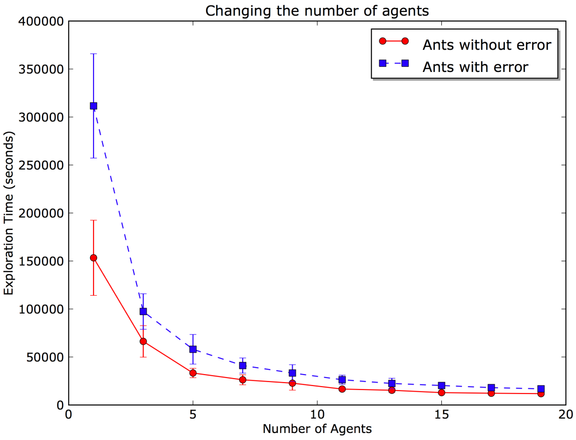

Figure 4: Effect of changing the number of agents.

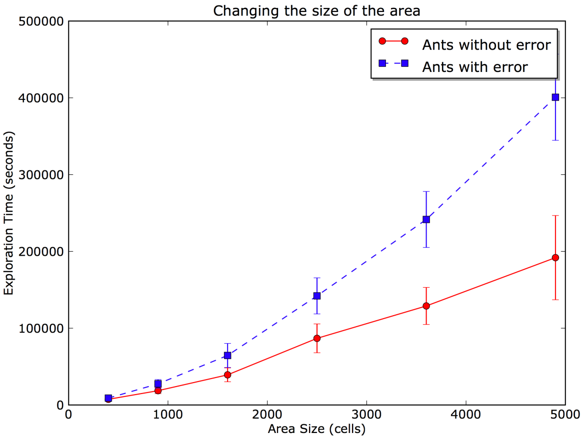

Figure 5: Effect of changing the size of the area.

the fact that with more rooms, accessible areas of the map are narrower, agents are more constrained in their movements, and have smaller chances of following long detour paths as a result of a sensor detection error.

5. CONCLUSIONS

In this paper, we studied the impact of localization errors on the performance of the Ants algorithm. We distinguished two types of errors: i) inaccuracies in determining the exact location of detected sensors wrt the agent's current position, and ii) complete failure to detect and localize sensons. We showed that small errors in locating sensors are not critical, but completely failing to detect sensors can significantly slow down the exploration process. The impact of failing to detect sensors is more pronounced in scenarios where few agents are used to explore large areas with few rooms.

Acknowledgments

Effort sponsored by the Air Force Office of Scientific Research, Air Force Material Command, USAF, under grant number FA8655-06-1-3003. The U.S. Government is authorized to reproduce and distribute reprints for Government purpose notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research or the U.S. Government.

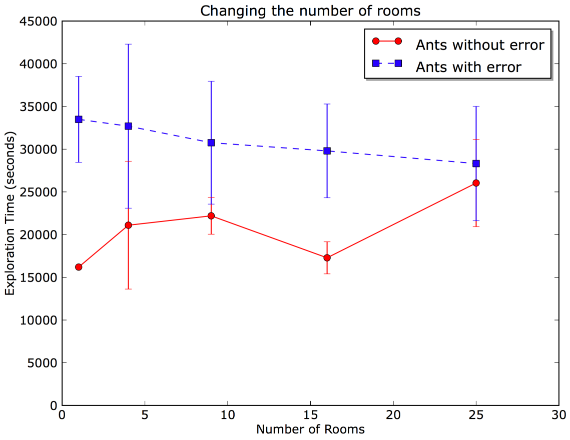

Figure 6: Effect of changing the number of rooms.

6. REFERENCES

[1] M. Batalin, G. Sukhatme, and M. Hattig. Mobile Robot Navigation using a Sensor Network. In *ICRA04*, pages 636–642. IEEE Press, April 2004.

[2] R. O. Castle, D. J. Gawley, G. Klein, and D. W. Murray. Video-rate recognition and localization for wearable cameras. In *BMVC07*, pages 1100–1109, September 2007.

[3] E. Ferranti and N. Trigoni. Practical issues in deploying mobile agents to explore a sensor-instrumented environment. Technical Report RR-09-02, Oxford University Computing Laboratory, February 2009.

[4] E. Ferranti, N. Trigoni, and M. Levene. Brick&Mortar: An On-Line Multi-Agent Exploration Algorithm. In *ICRA07*, pages 761–767. IEEE Press, April 2007.

[5] S. S. Ghidary, T. Tani, T. Takamori, and M. Hattori. A new Home Robot Positioning System (HRPS) using IR switched multi ultrasonic sensors. In *SMC99*. IEEE Press, October 1999.

[6] S. Kataoka and K. Atagi. Preventing IR interference between infrared waves emitted by high-frequency fluorescent lighting systems and infrared remote controls. *IEEE transactions on industry applications*, 33(1):239–245, January/February 1997.

[7] I. Kelly and A. Martinoli. A scalable, on-board localisation and communication system for indoor multi-robot experiments. *Sensor Review*, 24(2):167–179, January 2004.

[8] N. Kirchner and T. Furukawa. Infrared Localisation for Indoor UAVs. In *ICST05*, pages 60–65, November 2005.

[9] J. H. Lim and J. J. Leonard. Mobile Robot Relocation from Echolocation Constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):1035–1041, September 2000.

[10] A. C. Rice. Dependable Systems for Sentient Computing. *PhD Thesis, University of Cambridge*, 2007.

[11] J. Svennebring and S. Koenig. Building Terrain-Covering Ant Robots: A Feasibility Study. *Autonomous Robots*, 16(3):313–332, May 2004.