Université Libre de Bruxelles
Faculté des Sciences Appliquées
CoDE, Department of Computer & Decision Engineering
IRIDIA, *Institut de Recherches Interdisciplinaires
et de Développements en Intelligence Artificielle*

On the Design of Self-Organized Decision Making in Robot Swarms

Alexandre CAMPO

Promoteur de Thèse :
Prof. Marco Dorigo

Co-promoteur de Thèse :
Prof. Jean-Louis Deneubourg

Thèse présentée en vue de l'obtention du titre de Docteur en Sciences de l'Ingénieur

Année Académique 2010/2011

Abstract

In swarm robotics, the control of a group of robots is often fully distributed and does not rely on any leader. In this thesis, we are interested in understanding how to design collective decision making processes in such groups. Our approach consists in taking inspiration from nature, and especially from self-organization in social insects, in order to produce effective collective behaviors in robot swarms. We have devised four robotics experiments that allow us to study multiple facets of collective decision making. The problems on which we focus include cooperative transport of objects, robot localization, resource selection, and resource discrimination.

We study how information is transferred inside the groups, how collective decisions arise, and through which particular interactions. Important properties of the groups such as scalability, robustness, and adaptivity are also investigated. We show that collective decisions in robot swarms can effectively arise thanks to simple mechanisms of imitation and amplification. We experimentally demonstrate their implementation with direct or indirect information transfer, and with robots that can distinguish the available options partially or not at all.

To my parents

"Impose ta chance, serre ton bonheur et va vers ton risque.
À te regarder, ils s'habitueront."
René Char

Acknowledgements

Before becoming the pages you hold in your hands, this thesis was a scientific journey. I owe much to the many persons who accompanied me through it.

Financially, I was supported by COMP²SYS, a Marie Curie Early Stage Research Training Site funded by the European Community's Sixth Framework Programme under contract number MEST-CT-2004-505079. I was also granted a fellowship from the belgian national fund of research (aspirant du F.R.S.-FNRS). Lastly, the Jules Reyers foundation supported me while I was writing this thesis.

I carried out my research at IRIDIA, an international research institute co-directed by Hugues Bersini and Marco Dorigo. IRIDIA receives numerous researchers from all around the world and offers the necessary freedom to nurture and develop original research. My stay at IRIDIA was made enjoyable by all my colleagues. Many thanks to all of you who spontaneously contributed to create a friendly atmosphere in the lab. I also had the opportunity to interact with the members of the Unit of Social Ecology (USE) co-directed by Claire Detrain and Jean-Louis Deneubourg. There, I met persons who were available to discuss and debate about science, and who kindly offered their help whenever possible.

My studies were supervised by three senior researchers. To begin with, I am grateful to Marco Dorigo for undertaking the task of supervising my doctorate. During all this time, Marco has been careful to stay open minded and critical, which means he was never too enthusiastic, and that he was never closed to new ideas. He offered me the necessary freedom to be innovative, and responsible. Through the years, Marco constantly followed these principles and I thank him for his steadiness. In addition, I also thank Jean-Louis Deneubourg for co-supervising my work and lending an attentive hear to me. His bright and rapid scientific mind helped me better shape my projects and my thoughts. He also offered a decisive help when I was struggling to make mathematical models. Lastly, Mauro Birattari played a role that was halfway between the collaborator and the supervisor. With him I could frequently discuss new ideas and receive critical feedback or help on technical matters. Notably, I thank him for initiating the very first work on infrared communication, and for designing a measure of the polarization of a group which works independently of the group size. In addition, Mauro is very good at picking the right words, which allowed him to pass me otherwise difficult messages.

While making this scientific journey, I observed learned, matured. My perspective and my understanding of research has changed substantially. Through numerous discussions, my colleagues often exposed new ideas and raised new problems that were challenging my initial research directions. Often, I decided to follow these new research leads rather than only sticking to my plans. Here, I would like to thank the many persons who took part in this adventure.

Àlvaro Gutiérrez offered me a very fruitful collaboration. I believe that his exceptionally kind and patient personality makes it possible to undertake difficult challenges. I talked Àlvaro into teaching me some basics in PIC programming and helping me to create an infrared communication system. Together, and with the help of Valentin Longchamp, we designed the first infrared communication system on the e-puck robot. I am also quite proud of its electronic counterpart, an extension module that we created for the e-puck. Thanks to Àlvaro, who produces the board and offers technical support, this board is now used in various labs across Portugal, Spain, Belgium, Switzerland, Italy, Canada and the United States of America. Last but not least, we used these tools to carry out several experimental robotics studies, including the artificial evolution of orientation behavior and social odometry. In the first study, I had the chance to collaborate with Elio Tuci, who shared his passionate vision of the field and offered his philosophical scientific perspectives. In the second study, I also had the pleasure to work with, and learn from the talented, yet humble Francisco C. Santos who proposed to use the Fermi function in social odometry.

Etienne Toffin welcomed me warmly when I spent time in the unit of social ecology. His work on the digging behavior of ants was arousing my curiosity and I had the chance to contribute to it by lending my skills of computer scientist. We devised other fun projects which did not have a chance to see the light until now. Nevertheless, I am confident that we will collaborate again in the future.

Roderich Groß kindly helped me to implement the vision system of the s-bot. He proposed the idea of using vision to perform cooperative transport which led me to create the visual range and bearing system. This allowed us to perform new experiments on cooperative transport.

Shervin Nouyan also helped a lot on the cooperative transport work. Our main collaboration was on the subject of robot chains. Together, we discussed new ways to create robot chains and we designed self-organized mechanisms to select resources with these chains. We also had a great time supervising an e-puck student project together with Mauro Birattari.

Simon Garnier has been a long time partner. Our similar point of views led us to work on subjects alike, and eventually collaborate together. The work on collective discrimination was started and completed thanks to him. We also further studied the properties of this collective behavior. I am particularly grateful to Mehdi Moussaïd for discussing this work and giving the idea of controlling collective discrimination in order to produce a behavior of swarm search.

Furthermore, I had the pleasure to supervise a number of student projects during my thesis. The students I met surprised me with their enthusiasm, their collaborative spirit, and their respectful interactions. It was for me a captivating experience: while they were learning technical knowledge from me, I was observing and learning from them too. In particular, Olivier Dédriche and Mouhcine Zekkri made an important initial contribution to the study of collective discrimination.

I am very indebted to my colleagues Etienne Toffin, Vito Trianni, and Yann-Aël Leborgne for proof reading large parts of this thesis and giving me their precious feedback along with smart ideas to improve the document.

Outside the lab, numerous friends stood by me, and offered me other perspectives on life. Moments of laughs, passionate discussions, and other adventures fueled my personal and professional life.
My heartfelt thanks to Agathe, Anaïs, Antal, Carine, Carole, Christelle, Eric, Giacomo, Julien, Kamel, Marie-Hélène, Matteo, Meritxell, Nathalie & Hervé, Roxane, Stéphane, Stéphanie, Valérie, Virginia & Pablo, Yann-Aël, and Yannick.
Music allowed me to escape boring situations, I am especially thankful to Jef and Thierry for sharing their knowledge and patiently introducing me to this world of endless wonders.
Last but not least, my climbing mates offered me pure adrenaline, and chronic pain to the zygomatics. Thank you Emilie, François, Jonathan, Stéphanie et Julien, Marie-Laure, Olivier, Samuel, and Sandrine.

My special thanks go to my nearest and dearest. Anne, who accompanied me through the good times and the less good, giving affection and care. My parents for their constant presence and support in all situations.

A.C.

Contents

Chapter 1

Introduction

Decision making is a central process for groups of robots that evolve in dynamic environments. Like humans, animals, or other autonomous machines, they are regularly confronted to new situations that demand specific, adapted responses. When a single, monolithic entity makes a decision, it first gathers information about the current situation, evaluates what are the available options and the associated benefits and costs. After this evaluation, a coherent decision is finally made.

What about groups? The classic idea of decision making in groups relies on the concept of leadership. This time, several members of the group may perceive information and implement the decision, but information is still centralized towards a leader or a small leading council which makes the evaluation and then informs the group members of its decision.

Hierarchical and centralized organizations have been implemented numerous times in human society and proved very successful. In animal kingdom, complex social structures are also observed. For instance, groups of chimpanzees are socially structured by dominance relationships, while elephants and whales are matriarchal. In the artificial world, machines have lent themselves perfectly to centralized architectures because of their ability to process information reliably and rapidly. Leadership allows groups to avoid indecisiveness, to be coherent, and to make fast decisions.

However, there are situations in which leadership is impractical or prone to failure. The leader itself is a point of weakness for the group. If the leader is removed, or for some reason he is unable to make a sensible decision, the whole group is penalized. Furthermore, the task of gathering information and informing members of the central decision may be very difficult to carry out. This is the case if the group is very large and the leader is overwhelmed by the amount of information. Also, information may take too much time to be transmitted, or it may become corrupt during relaying.

Interestingly, there are alternative types of organization that allow groups to make effective decisions. In nature, social species such as ants and bees live in groups without central leadership: their behavior is often referred to as self-organizing. We have all been fascinated at least once by the collective activity of these insects. They build their nest without an architect, they search for resources and decide which one to exploit without any supervisor. We know that the individual insects are not aware of what goes on at the level of the colony. Individuals are merely executing a limited set of simple behaviors depending on their local situation. Yet, the whole colony is able to carry out complex activities, including making effective collective decisions.

In this thesis we study new ways to make collective decisions with groups of robots. By transposing the principles of self-organization into robots, we expect to avoid centralized architectures and costly communication infrastructures, and still achieve effective collective decisions. We are interested in understanding how information is processed inside the groups, how collective decisions arise, and through which particular interactions. Important properties of the groups such as scalability, robustness, and adaptivity are also investigated.

We use the framework of robot foraging in order to tackle practical, experimental problems which require the implementation of collective decision making in groups of robots. The foraging problems on which we focus include the cooperative transport of objects, the localization of the robots, the selection of resources, and the discrimination of resources. All these problems are orthogonal and allow us to study multiple facets of collective decision making. In particular, we study how a decision is made by a group of identically informed robots, and how the collective decision can be improved when a number of individuals are better informed. We also explore a case of indirect information transfer, in which robots gradually build their collective decision by simulating the pheromone laying behavior of ants. Finally, we examine an experiment in which the collective decision results from the interplay of opposite feedbacks, with robots which perceive neither the characteristics of the options available, nor the collective decision made by the group.

1.1 Original contributions and related publications

Our contributions can be seen from two different perspectives. On the one hand, we study four different problems of foraging in groups of robots. For each problem, we propose a solution based on collective decision making, and demonstrate the feasibility of our approach through real world experiments. On the other hand, the chosen foraging problems all require robot groups to make a collective decision. The different implementations of this collective process allow us to make comparisons and identify fundamental, invariant mechanisms at the core of self-organized decision making processes in robots.

A preliminary step in our contributions consists in endowing robots with new means of communication, better suited for self-organizing collective behaviors. To this end, we have devised two specific systems of localization and communication for our robots. The first one relies on an omnidirectional camera, while the second one is implemented in a specific board with infrared sensors. These contributions were reported in the following articles:

- **Open e-puck range and bearing miniaturized board for local communication in swarm robotics**. A. Gutiérrez, A. Campo, M. Dorigo, J. Donate, F. Monasterio-Huelin, and L. Magdalena. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3111–3116. IEEE Press, Piscataway, NJ, 2009.

- **An open localization and local communication embodied sensor**. A. Gutiérrez, A. Campo, M. Dorigo, D. Amor, L. Magdalena, and F. Monasterio-Huelin. *Sensors*, **8**(11):7545–7563, 2008.

The first of the four experiments demonstrates cooperative transport in a group of robots. The robots have uncertain knowledge of the direction in which they must pull the object. We implement a collective decision mechanism that allows the robots to share and improve their knowledge, so as to coordinate and pull in the same direction. This work was presented in:

- **Negotiation of goal direction for cooperative transport**. A. Campo, S. Nouyan, M. Birattari, R. Groß, and M. Dorigo. In *Proceedings of the* 5^{th} *International Conference on Ant Colony Optimization and Swarm Intelligence*, LNCS, **4150**, pp. 191–202. Springer, Berlin, 2006.

- **Enhancing cooperative transport with negotiation of goal direction in swarm robotics**. A. Campo, S. Nouyan, M. Birattari, R. Groß, and M. Dorigo. In *Proceedings of the* 18^{th} *Belgium-Netherlands Conference on Artificial Intelligence*, pp. 365–366. University of Namur, Namur, Belgium, 2006.

In the second experiment, we examine collective localization by means of odometry. In this experiment, some individuals are better informed than others. We introduce a mechanism called social odometry that allows robots to share knowledge and spread better information across the whole group. In this work, Alexandre Campo participated to the design of the mechanism and of the experiments, contributed code for simulations and partly wrote the articles. This research work is covered in the following publications:

- **Collective decision making based on social odometry**. Á. Gutiérrez, A. Campo, F. Monasterio-Huelin, L. Magdalena, and M. Dorigo. *Neural Computing & Applications*, **19**, 807–823, 2010.

- **Social odometry in populations of autonomous robots**. A. Gutiérrez, A. Campo, F. C. Santos, C. Pinciroli, and M. Dorigo. In *Proceedings of the* 6^{th} *International Con-*

ference on Ant Colony Optimization and Swarm Intelligence, LNCS, **5217**, pp. 371–378. Springer, Berlin, 2008.

- **Social Odometry: Imitation Based Odometry in Collective Robotics**. A. Gutiérrez, A. Campo, F.C. Santos, F. Monasterio-Huelin, and M. Dorigo. *International Journal of Advanced Robotic Systems*, **6**(2):1–8, 2009.

In the third experiment, we focus on a foraging task in which robots navigate by following paths. The paths are made by robots that act as regular landmarks. In this experiment, we demonstrate resource selection by simulating virtual ants moving along the paths and depositing artificial pheromone inside the landmarking robots. This work is one of the first examples of collective decision making in robots with indirect information. It was reported in:

- **Path formation in a robot swarm: self-organized strategies to find your way home**. S. Nouyan, A. Campo, and M. Dorigo. *Swarm Intelligence*, **2**(1):1–23, 2008.

- **Artificial pheromone for path selection by a foraging swarm of robots**. A. Campo, Á. Gutiérrez, S. Nouyan, C. Pinciroli, V. Longchamp, S. Garnier, and M. Dorigo. *Biological Cybernetics*, **103**(5), pp. 339–352, 2010.

In the fourth experiment, inspired by biological studies on cockroaches, we show a mechanism that allows groups of robots to discriminate among resources. With collective discrimination, swarm decisions are not necessarily bound to maximize or minimize a given characteristic. In addition, we demonstrate how the choice of the robots can be controlled. Individuals are unable to perceive the characteristics of the resources, and do not have information about the collective decision made. Despite these limitations, we show that groups are able to make coherent collective decisions with this mechanism. We communicate on this research in the following articles:

- **Self-organized discrimination of resources**. A. Campo, S. Garnier, O. Dédriche, M. Zekkri, and M. Dorigo. *PLoS ONE*, 6(5):e19888.

- **Controlling the choice of robot swarms using collective discrimination**. A. Campo, Á. Gutiérrez, S. Garnier, M. Moussaïd, and M. Dorigo. To be submitted.

1.2 Other scientific contributions

While researching the material presented in this thesis, we also contributed to other research topics, which are not specifically concerned with decision making in robot swarms. These contributions are briefly presented in this section.

1. Investigation of a behavioral mechanism that allows robots to forage selectively and thus improve their benefits.

   - **Efficient multi-foraging in swarm robotics**. A. Campo and M. Dorigo. In *Proceedings of the* 9^{th} *European Conference on Advances in Artificial Life*, pp. 696–705. Springer, Berlin, 2007.

2. Design of an electromechanical device for the storage and cooperative transport of multiple objects.

   - **Enhancing the Cooperative Transport of Multiple Objects**. A. Decugnière, B. Poulain, A. Campo, C. Pinciroli, B. Tartini, M. Osée, M. Dorigo, M. Birattari. In *Proceedings of the* 6^{th} *International Conference on Ant Colony Optimization and Swarm Intelligence*, LNCS, **5217**, pp. 307–314. Springer, Berlin, 2008.

3. Exploration of the alignment task with artificial evolution techniques and our infrared communication system.

   - **Evolution of neuro-controllers for robots' alignment using local communication**. A. Gutiérrez, E. Tuci, and A. Campo. *International Journal of Advanced Robotic Systems*, **6**(1):1–10, 2009.

4. Analysis of fish motion.

   - **Analyzing fish movement as a persistent turning walker**. J. Gautrais, C. Jost, M. Soria, A. Campo, S. Motsch, R. Fournier, S. Blanco, and G. Theraulaz. *Journal of mathematical biology*, **58**(3):429–445, 2009.

5. Demonstration and analysis of a morphological transition during nest digging by ants.

   - **Shape transition during nest digging in ants**. E. Toffin, D. Di Paolo, A. Campo, C. Detrain, and J.-L. Deneubourg. *Proceedings of the National Academy of Sciences*, **106**(44):18616, 2009.

1.3 Thesis layout

This thesis is organized into 9 chapters. In Chapter 2, we gradually introduce the reader to the background and the related works of the research presented in this thesis. We start by a presentation of the swarm intelligence field, and proceed with a rough timeline of events that led to the birth of swarm robotics. We then detail the fundamental concepts that underpin self-organized collective decision making. Finally, we provide an overview of the foraging task in swarm robotics, and review the specific literature related to our experiments.

In Chapter 4, we investigate group decision making with an experiment of cooperative transport. In this experiment, identically informed robots must reach consensus about the transport direction.

Chapter 5 discusses our work on social odometry. In the experiments presented, we show how information of higher accuracy may propagate in the group, allowing the robots to localize more effectively different foraging areas.

Chapter 6 deals with the implementation of virtual ants laying artificial pheromone in a network of robots. In this experiment, artificial pheromone deposited inside the network of robots leads to the selection of the most profitable resource.

In Chapter 7, we present an experiment of collective resource discrimination. Although single robots are not able to evaluate the options available, the group is able to make a collective decision and select a resource of specific size which is neither too small nor too large.

Chapter 8 extends the work on collective discrimination by demonstrating how the decision made by the group of robots can be controlled.

Finally, Chapter 9 concludes this thesis with a short synthesis of our achievements and suggestions for future directions to explore.

Chapter 2

Background and related works

In this chapter, we present the context in which this thesis takes place, we review the major concepts that underpin our work and we discuss related literature. The chapter is divided in four sections that gradually bring the focus on the core subject of the thesis. In Section 2.1, we describe the field of swarm intelligence. Section 2.2 is devoted to the presentation of swarm robotics. Then, in Section 2.3, we examine the fundamental mechanisms at the core of collective decision making in robot swarms. Finally, we present the foraging task in Section 2.4. There, we show how we use the foraging task as a framework to study collective decision making through four different problems. In addition, for each problem studied in this thesis we review the specific literature and related works.

2.1 Swarm intelligence

Swarm intelligence is more easily described with examples from our daily life, especially the ones of group living animals that produce intriguing and unexpectedly complex collective behaviors.

Birds, and especially starlings, display swarm intelligence when they fly together (see Figure 2.1a). These birds are very gregarious and form large flocks which move fluidly, expanding, contracting and changing shape at any moment. Despite the speed of the flocks, and the sudden turns they may take, collisions rarely occur among birds. It is as if an external force were driving them to achieve a timely ballet with no coordination mistake. But we know there is no external guidance nor any leader driving the flocks. Rather, all the individuals in the swarm contribute to the same effort at the very same time. Despite the lack of a leader, these flocks seem to have a mind of their own, for instance when birds collectively agree on a roosting site.

Ant colonies also exhibit some of the most complex collective behaviors found in the animal kingdom (see Figure 2.1b). For instance, ants explore their environment to find resources and then select and exploit the most profitable one. Although single ants

(a)

(b)

(c)

(d)

(e)

(f)

Figure 2.1: Examples of swarm intelligence around us. **(a)** Birds flying together and forming a large flock. **(b)** Weaver ants pulling leafs so as to glue them together and build their nest. **(c)** Wikipedia, an encyclopædia produced by collaborative editing. **(d)** The internet network represented by a partial map of connections among computers. **(e)** Fish moving together in schools so as to better detect predators and evade their attacks. **(f)** Swarming bees, looking for a new nest site.

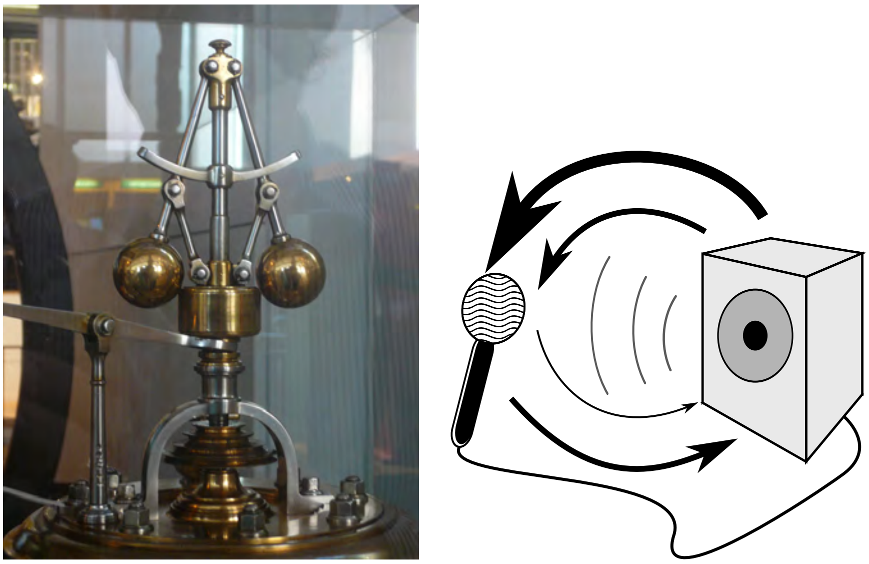
often visit only one resource and do not compare the different opportunities available, the colony is able to make the correct choice (Beckers *et al.*, 1990, 1992). Ants are also able to build architectured nests with tunnels and chambers. While ants do have a queen, the name of that particular member of the colony is misleading. The queen does not supervise the activity of the colony, but is merely responsible for laying eggs. Instead, all the ants apply basic behavioral rules depending on the local situation they perceive. In fact, there is not a single ant that can picture what is going on at the level of the colony. Even though isolated individuals follow simple behavioral rules, a whole ant colony exhibits a coherent and complex activity.

Swarm intelligence is not only observed in animals, but it also seems present in the human society. Nowadays, the internet provides among the most striking examples of human swarm intelligence. Wikipedia, the famous online encyclopædia, is the product of numerous and independent small contributions (see Figure 2.1c). It is well known that there is not a single leader who coordinates the edition process in Wikipedia. Although some individuals may organize in structured groups to work on a serie of related articles, the ordinary case is that anyone may edit the pages of this encyclopædia. Prior to its success, a large number of observers would have bet on the failure of Wikipedia, especially because it does not integrate protections against vandalism or other malicious editing. Yet, the uncoordinated activity of all the internauts produced in less than 10 years the largest compendium of general knowledge with honorable quality (Giles, 2005).

All these examples have in common the idea of groups performing an activity with a great degree of coordination. The fascinating part is that the groups behave as a single entity, but we know that the collective behavior observed is the outcome of numerous individual actions performed at the same time. Individuals have access to limited information, and their understanding of the collective activity is weak or nonexistent. Despite these limitations, individuals do not ruin the harmony of the collective behavior. With a small number of simple behaviors, they contribute to it. Such groups are not organized in the way we are used to. They do not have a leader, and there is no clear hierarchy among the individuals. This type of organization that seemingly evades any predefined structure is called self-organization.

2.1.1 Self-organization

Self-organization attracted attention in many different fields, such as chemistry, physics, biology, linguistics, economy, and robotics. Because of this diversity, it is difficult to find one single definition of self-organization that satisfies everyone. Nevertheless, it is possible to summarize from the literature the main characteristics of self-organization with a minimalistic definition.



(a) (b)

Figure 2.2: Examples of feedback mechanisms. **(a)** The centrifugal governor introduced by James Watt in 1788 exploits a negative feedback to regulate the energy delivered by a steam engine. **(b)** The Larsen effect is a well known instance of positive feedback. Any sound produced by the speaker is picked up by the microphone and goes back to the speaker reinforcing itself.

Self-organization is a process whereby a system made of interacting components and embodied in its environment produces, through numerous interactions, a spatiotemporal pattern in the environment. In addition, the system is not guided by a leader, and the components of the system typically have limited information about the collective activity. Most of the time, the components behave probabilistically. Hence, the whole system is stochastic and its activity is rarely exactly reproduced, unless initial conditions are very strictly controlled.

Contemporary self-organization finds its origins in the concept of homeostasis elaborated by Claude Bernard (Bernard, 1864, 1871; Cooper, 2008) and further developed by Cannon (1929). Bernard noticed that an important property of living organisms is their ability to maintain their internal state. For example, the internal temperature of our bodies is regulated so that our organism functions in a rather broad range of environmental conditions. This idea was generalized to any system able to regulate some of its internal parameters, and these systems were said to be homeostatic.

Regulation in homeostatic systems was explained with feedbacks. Inside a system that responds to a given stimulus, a positive feedback enhances the system's output by constantly reinforcing the response of the system. This feedback makes the system unstable, with the risk to spiral out of control, but also with the possibility to end up in one of several stable states. Conversely, a negative feedback depresses the response of the system when it exceeds some threshold.

An early example of negative feedback is the centrifugal governor (see Figure 2.2a) popularized by James Watt in 1788 when he was working on steam engines (Maxwell, 1868). At that time, steam engines were used for driving factory machinery and the mechanics employed in the industry were fragile and very sensitive to variations in the engines' output. To ensure reliable function of the factories, constant motor output was necessary. To this end, Watt used the centrifugal governor, a mechanical system that regulates the admission of steam in the engines. The centrifugal governor consists of a conical pendulum with two metallic balls revolving within a circular case. When the engine's output increases, the balls rotate faster, move upwards, and the arms of the pendulum open and modify the aperture of a steam valve. In this way, the energy delivered by the engines is better controlled and almost constant.

Positive feedback is very well illustrated by the Larsen effect that arises when a microphone's output is plugged to the very same speaker it is recording (see Figure 2.2b). Larsen effects happen sometimes live on stage, during concerts, because the microphone of an artist picks up the signal of the feedback speakers. Even if there is almost no noise around, any small perturbation recorded by the microphone is reproduced by the speakers at a higher volume, and in turn this new and louder sound is captured by the microphone. Therefore, even in a place that is mostly silent, the tiniest noise will be amplified by this positive feedback mechanism to the point that the speakers are loudly squealing.

In the late 1940's, regulatory systems (*i.e.*, systems that contain feedbacks) became the subject of interest of a new field called "cybernetics" (Wiener, 1948). Cybernetics emerged during the Macy conferences that took place between 1946 and 1953. These conferences brought together scholars from various disciplines with the goal to openly discuss new problems and questions encountered in modern science. Quickly after the birth of cybernetics, Ludwig Von Bertalanffy proposed the general system theory (Von Bertalanffy, 1950a,b, 1968), which offered a complementary vision of systems to cybernetics. Historically, these are the foundations of swarm intelligence and swarm robotics.

As part of the emerging field of cybernetics, Ashby (1947) introduced the concept of self-organization which is now central to swarm intelligence. However, self-organization only became popular in 1977, when Ilya Prigogine was awarded a Nobel prize for his work on dissipative structures and open systems operating out of thermodynamic equilibrium (Nicolis *et al.*, 1977).

After a century of works that gradually contributed to the definition and study of self-organizing systems, in 1989, Beni and Wang coined the term "swarm intelligence" which later became popular in biology and computer science (Beni and Wang, 1989; Beni, 2005). Swarm intelligence designates natural or artificial self-organizing systems that display some intelligence in their activity.

2.1.2 Properties of swarm systems

Swarm systems have properties that can render them superior to the more classical monolithic and centralized systems (Beni and Wang, 1989; Beni, 2005).

Scalability may be the most appealing property of a swarm. Centralized systems have very limited scaling capacities because the number of interactions required to achieve coordination grows exponentially with the group size (Simon, 1962; Klavins, 2002; Seeley, 2002). In self-organizing systems, interactions are either local or they only involve small fractions of the population. Hence, for a single individual, the number of interactions experienced does not strongly depend on the group size. A system scales successfully if it is possible to increase its size without perturbating its function. Ideally, increasing group size also increases the performance of the system. In the best case, the system displays a super linear improvement of its performance (Mondada *et al.*, 2005).

The **parallel activity** of the swarm is also an interesting property. With it, swarms can perform different actions in various places at the same time. A straightforward use of this omnipresence is the exploration of an environment to localize resources, as exemplified by the behavior of great tits (*Parus major*) which find food sources faster when in group (Clark and Mangel, 1986). More generally, any task that can be partitioned in independent subtasks may be carried out faster by a swarm thanks to parallelism.

Because the environment in which swarms operate is rarely static, it is interesting to have the swarm cope with changes and dynamic situations. Depending on the context and the task performed by the swarm, this implies either a **robust behavior** or an **adaptive behavior**. Robustness to failure is a common property of swarms that contain several individuals that can be interchanged. Thanks to this redundancy, if an individual fails to carry out its work, another individual promptly replaces the incapacitated one. Ants provide good examples of robustness and adaptivity: if resources are scarce, a limited number of workers are recruited to forage outside the nest. If a new resource appears in the environment, scouts detect it and recruit more workers to exploit it. The colony therefore adapts its collective behavior to better exploit the available resources. To pursue with the same example, imagine that the ants are now exploiting the new resource and have established a pheromone trail that connects the resource and the nest. Suppose that a small branch falls on the ground and interrupts the pheromone trail. The ants notice the problem and either try to go around the branch or even move the branch if possible. This time, the colony maintains its collective behavior despite the environmental change (Hölldobler and Wilson, 1990).

It is important to emphasize the fact that these properties do not come for free in any swarm system. The presence of these properties depends on the environment, the task that the swarm has to carry out, and the behavior of the swarm. In artificial systems, it is particularly important to carefully design the behavior of the swarm so as to benefit from these potential advantages as much as possible. For instance, the best known obstacle to scalability in swarms of robots is overcrowding. If the density of robots becomes too high in a place, the robots spend all their time avoiding each other instead of performing their task (Goldberg and Matarić, 1997).

2.2 Swarm robotics

Robotics is one of the most challenging field of research that can make use of swarm intelligence and demonstrate its usefulness in a practical context. As distributed systems are becoming more and more important in robotics, the properties of swarm intelligence systems become more and more appealing.

Humans create robots to replace them in dangerous, or repetitive tasks (Bekey, 2005). In addition, robots can sometimes perform tasks better than humans when it comes to accuracy, strength, or speed. Just as it was the case for self-organization, there is no clear definition of what a robot is. Some consider that robots are artificial systems that can sense the environment and react to the situation perceived. Under this definition, a blunt thermometer becomes a robot. This definition is therefore too general, but any attempt at restricting it leaves useful or interesting devices out of the definition's scope.

In this thesis, however, we work with autonomous mobile robots, a specific type of robots that lends themselves more easily to an unambiguous definition: autonomous robots are electromechanical devices able to move and operate in their environment without any human assistance. They are usually designed to carry out one or several specific tasks, such as exploration of the environment, or transport of objects.

Robotics has witnessed significant advances in the last 50 years. We are especially interested in the paradigm shift that led a number of researchers to change their center of interest from isolated mobile robots to swarms of robots. This evolution involved many researchers from various fields: the summary provided here is very partial and underlines only a few particular contributions, but it hopefully allows to better understand why and how researchers turned to swarm robotics [1].

2.2.1 From single robots to groups of robots

William Grey Walter created one of the first autonomous mobile robots in the late 1940's (Grey Walter, 1950, 1951, 1954). Actually, he started his studies with two small robots called *Elmer* and *Elsie*, shown in Figure 2.3. These robots were designed to move towards light and to detect collisions with obstacles. Grey Walter was amazed by the

[1] The interested reader may consult Trianni (2008) and Pickering (2010) for more detailed stories.

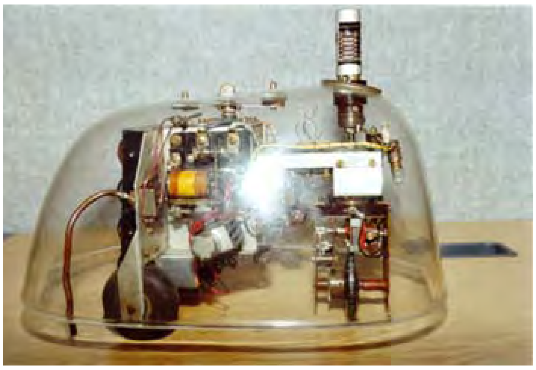


Figure 2.3: One of the turtle robot used by Grey Walter in his studies (restored version shown here).

complex activity displayed by his robots, which were merely executing a combination of two very basic behaviors [2].

From that moment on, the design of robots improved in several directions: the sensors, the actuators, the processor, the autonomy and the controllers. Advances in electronics and mechanics guided the evolution of the robots morphologies. In parallel, mathematicians started to devise algorithms to control the robots, and more generally to make machines "intelligent".

At the time, machine reasoning was implemented with symbols and logic because of the strong influence of Frege's and Russell's works (Frege, 1879; Whitehead and Russell, 1910). Machines, and robots, were designed to make inferences using rules and symbols. This approach led Allen Newell, Herbert Simon, and Cliff Shaw to introduce in 1955 a program named the *Logic Theorist* (Newell and Simon, 1956; Russell and Norvig, 1995). The program was able to prove mathematical theorems, mimicking the problem solving skills of humans. This work was a foundation for the new field of artificial intelligence. Later, Newell and Simon went further, claiming that physical symbol systems have the necessary and sufficient means for general intelligent action (Newell and Simon, 1976).

Following the success of the *Logic Theorist*, a number of laboratories attempted to build robots that used symbols and inference rules. The environment surrounding the robots was represented with the symbols, and inference rules were used to devise plans. The idea was that robots would be able to carry out their tasks by executing these plans step by step (Russell and Norvig, 1995). However, the capabilities of intelligent machines were limited, and they could mostly solve toy problems. Researchers were running into fundamental problems such as intractability (Cobham, 1965; Cook, 1971; Garey and Johnson, 1979), and the frame problem (McCarthy and Hayes, 1969). Intractability often resulted from the implementation of artificial reasoning as the search

[2]The behavioral diversity that such a simple robot can display was remarkably analyzed by Valentino Braitenberg (1986)

of a satisfying solution inside a huge tree of chained inferences. The size of the tree was growing exponentially with the dimension of the problems tackled, which prevented machines to find solutions in reasonable amounts of time. The frame problem underlined the difficulties to correctly infer the consequences of an action in a dynamic environment. Additional rules were required to explicitly define a frame, *i.e.* define what was left unchanged in the environment by an action. However, the number of rules required was such that it made problems intractable. Moreover, assuming a minimal amount of changes without specifying more rules led to unexpected anomalies. To deal with these problems, a lot of strategies were imagined such as heuristics to explore trees faster and devise adaptive plans or new kinds of logic to deal with uncertainty, changes, and time. In robotics, one of the most remarkable proposition stood out for its effectiveness despite a striking simplicity.

From 1986 to 1991, Rodney Brooks published a series of papers in which he advocated the use of a subsumption control architecture and described how his team and himself managed to create robots that could wander safely in dynamic environments, grab empty soda-cans left on tables and drop them to trash bins (Brooks, 1990). With subsumption, these robots only relied on basic behavioral rules to operate (Brooks, 1986) and they were not planning their trajectories, nor did they rely on any deep symbolic reasoning to accomplish their tasks (Brooks, 1991). Robotics researchers, absorbed with complicated workarounds in logic, or symbolic models of the mind were caught by surprise by these results. Brooks showed an effective alternative to symbolic systems and in this way laid the foundations of what became known as behavior based robotics (Arkin, 1998).

The emergence of behavior based robotics corresponds to the booming of collective robotics. This field suddenly attracted the attention of researchers in the late 1980's. In fact, the reviews of Dudek *et al.* (1996) and Cao *et al.* (1997) do not cite any collective robotics studies dating before 1986, when the first paper of Brooks was published. This is not a coincidence: behavior based robotics played an important role in the development of collective robotics because it was not necessary anymore to endow robots with very accurate sensors, huge computational resources, or complex programs (Matarić, 1992b). In addition, the control of multiple robots with classical planning techniques was extremely challenging due to the exponential number of possibilities to take into account (Matarić, 1992a). Behavior based robotics, along with the continuous progress of electronics, provided researchers with simple and cheap robots that were able to perform a variety of simple tasks in dynamic environments. Some researchers naturally attempted to improve the robots so that they could perform more and more complex tasks. Other researchers realized that instead of redesigning robots, it was possible to tackle more complex tasks by increasing the number of robots working together.

2.2.2 Blending swarm intelligence and collective robotics

The name "swarm robotics" can be seen as the contraction of swarm intelligence and collective robotics. This summarizes quite well the scientific subjects covered by the field.

In collective robotics, tasks are performed by groups of robots. This confers clear advantages over robots operating alone: some tasks may be difficult or impossible for a single robot to accomplish, and some tasks may be carried out more efficiently with a group of robots. In addition, it can be easier and cheaper to manufacture a group of simple (and often identical) robots rather than building a single specialized and complex robot (Cao *et al.*, 1997).

Examples of tasks that benefit from using groups of robots rather than single robots include ditch crossing and cooperative transport. When a ditch separates robots from their goal and single robots are not able to overcome the ditch alone, they have to option to self-assemble. In doing so, robots form a super entity that has new physical properties allowing the whole group to move over the ditch (Trianni and Dorigo, 2005). Suppose now an heavy object should be moved away but single robots are unable to move it. If several robots work together and pull the object in the same direction, their efforts add together and the object is successfully transported (Groß *et al.*, 2006b,a; Nouyan *et al.*, 2009a).

When working in groups, robots need to coordinate their activities. Indeed, there are only few tasks that can be tackled by robots unaware of each other. Although a group of vacuum cleaner robots might perfectly carry out its task without any information transfer, most non trivial tasks require cooperation and coordination.

To coordinate their actions, robots can use a centralized organization, with a leader that gathers information from all its teammates, makes decisions and gives orders to all the robots. Robots may also communicate with each other through a blackboard system, a specific device that can be accessed concurrently to exchange messages or record data (Hayes-Roth, 1985). While these coordination mechanisms have been successfully implemented in a variety of tasks, they inherently lack robustness because the central component is a point of weakness. In addition, a dedicated communication infrastructure is often required to allow all the robots to communicate quickly and reliably at any time. Lastly, these systems are difficult to scale because they have communication bottlenecks (Klavins, 2002). These limitations are most problematic in dynamic, unknown or hostile environments.

A completely different approach to achieve coordination consists in introducing self-organization into groups of robots. In this case, groups of robots become swarm systems and inherit from the properties of these systems. They are likely to be more robust and fault tolerant and they may scale to larger extents. Hence, swarm robotics is concerned with the creation and design of self-organizing groups of robots that collectively perform tasks: robots that display swarm intelligence.

2.2.3 The problem of designing artificial swarm systems

Artificial swarm systems, in our case self-organizing groups of robots, pose several challenges. The fundamental design problem finds its origin in the gap between the microscopic and the macroscopic levels. These two levels of description bear on the same system: they simply are two different points of view of the same phenomenon. The macroscopic level describes the spatiotemporal pattern produced by the collective behavior of the system. The microscopic level describes the behavior of the individuals which governs the interactions between individuals and the interactions between individuals and their environment.

Although these levels of description are two sides of the same coin, they are very difficult to relate (see Figure 2.4). One can define what happens at the macroscopic level—the spatiotemporal pattern created by the collective behavior, but there is no direct method to infer from this specification what happens at the microscopic level. A slightly easier problem is to find out what happens at the macroscopic level when the microscopic level is specified. In some cases, it is possible to use mathematical tools such as differential equations (Martinoli and Easton, 2003; Martinoli *et al.*, 2004; Lerman *et al.*, 2005; Winfield *et al.*, 2008), and in most of the cases simulations are very effective tools (Meteopolis and Ulam, 1949).

However, these tools have their limitations and they are not a definitive solution to understand the consequences of individual behavior at macroscopic scale. Mathematical models can sometimes be very hard to devise if, for instance, there are delays in the system, or if some event arises with a probability density function that can not be expressed with the standard mathematical tools. Simulations have also their limits: they lose much of their significance when dealing with chaotic systems or systems highly sensitive to initial conditions. In addition, the computational resources required by simulations explode with the number of interacting components of a system and it is nowadays very challenging to simulate non trivial systems made of more than a million of components (in contrast, mathematical models can be studied independently of the group size considered; Lerman *et al.*, 2001).

One last difficulty encountered in the understanding and design of self-organizing systems is that they are equifinal (Von Bertalanffy, 1950a). A given collective behavior at the macroscopic level can be produced by several different individual behaviors at the microscopic level. Also, a given individual behavior can generate different collective behaviors. For instance, when a system must decide between two strictly equivalent

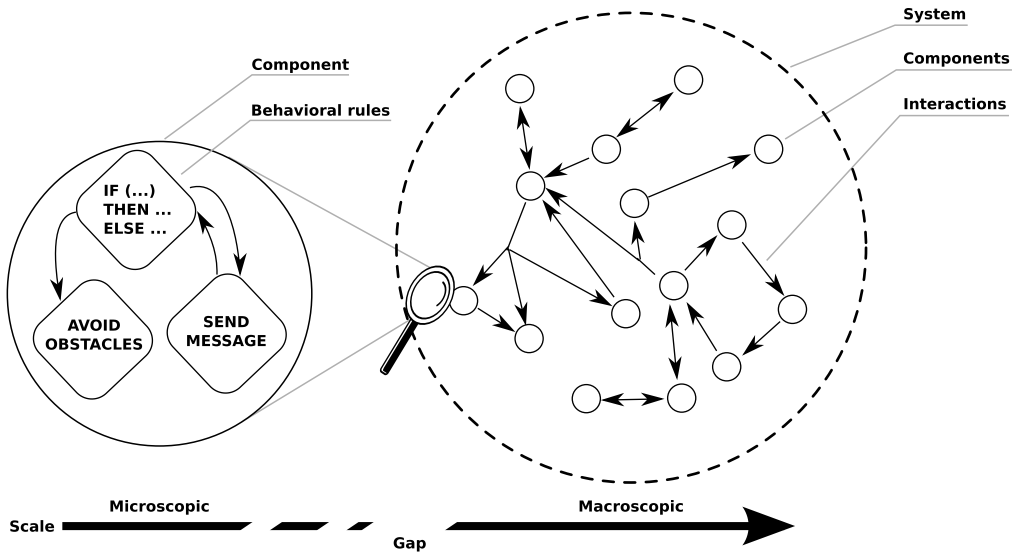


Figure 2.4: The gap between two levels of description of the same self-organizing system. On the left, the microscopic view: the focus is set on a single component of the system and its behavioral rules. On the right, the macroscopic view represents the full system made of numerous interacting components. The environment in which the system is embodied is not represented for the sake of clarity.

options, it will not always choose one of the two options like a deterministic system would do, but it will select any of the two options with equiprobability.

Most likely, the gap between the different levels of description will not be directly bridged any time soon, but this problem can be circumvented. To facilitate the design of swarm systems, two main directions are currently explored by researchers in swarm robotics.

The first direction is artificial evolution (Nolfi and Floreano, 2000; Trianni, 2008). It basically consists in letting a computer try and simulate many individual behaviors until a satisfactory one is found, *i.e.*, one that leads to a specified collective behavior. However, because the number of possibilities is extremely large and computers can not test all of them, specific evaluation strategies are required to converge to interesting solutions. Artificial evolution has already demonstrated its potential, and a large number of systems have been designed with it. However, this research direction still faces important challenges. The solutions found by artificial evolution strongly depend on the fitness function specified, that is the way the desired result is described. The results of evolution are typically black boxes, and very often no further understanding is gained from a successful evolution. It is then necessary to reverse engineer the output of artificial evolution so as to understand the mechanisms found by the machine. Artificial evolution is also hard to predict, and solutions can rely on unexpected, or even undesired strategies (Christensen and Dorigo, 2006).

The second direction, which is the one adopted in this thesis, consists in taking inspiration from nature, and in particular from the many works on self-organizing systems reported in the fields of physics, chemistry, sociology and biology (Bonabeau *et al.*, 1999; Camazine *et al.*, 2001; Kennedy *et al.*, 2001; Garnier *et al.*, 2007). This strategy is very effective, and a number of self-organizing systems have been created by reproducing and adapting the mechanisms observed in nature. However, this research direction is also facing intense challenges. The mechanisms at work in natural systems are not necessarily meaningful in artificial systems. A good example of this problem is the chemical communication system of ants, using pheromones, that can hardly be transposed to robots. There is no cheap and miniaturized technology that would allow robots to deposit and perceive pheromones reliably (Stella *et al.*, 1995; Svennebring and Koenig, 2003; Purnamadjaja and Russell, 2004), and it is not clear if such a technology could ever be used outside laboratories. Another problem is the construction of complex collective behaviors. Copying mechanisms from nature is not enough to be able to put them together and produce new and more complex systems. It is not clear yet how modularity can be introduced into self-organizing systems in order to combine collective behaviors and handle possible interferences.

2.3 Designing decision making processes in swarms

To design collective decision making, we need to define behavioral rules, along with the interactions they trigger inside the swarms. In the following, we review the various types of interactions that can take place in a swarm and we identify the main behavioral rules responsible for collective decision making in swarms by taking inspiration from biological studies (Conradt and Roper, 2005; Conradt and List, 2009).

2.3.1 Interactions in swarms

Swarm systems are made of components or individuals that interact with each other. These interactions can happen through **direct information transfer** or **indirect information transfer**.

Direct information transfer

Direct information transfer, or direct communication, involves an emitter, a receiver and a medium through which information is transferred (Shannon, 1948; Wiener, 1948). When dealing with self-organizing systems, an important characteristic of direct communication is that it does not last in time, and that duration of communication is a negligible parameter. In robotics, such communication can be achieved with light, sound,

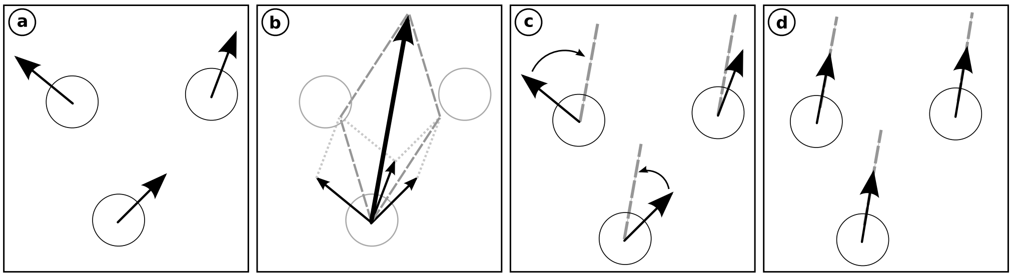


Figure 2.5: Collective motion is commonly achieved with direct information tranfer. **(a)** Consider three robots that have random orientations. **(b)** Each robot perceives the orientation of its neighbors and calculates the local average orientation. **(c)** Now the robots imitate their neighbors by gradually changing their orientation so as to match it with the neighbors' average. **(d)** By repeating several times the two previous steps, robots end up aligned.

radio waves, contact, and more rarely with odors. The properties of the medium utilized to transfer information have consequences on the characteristics of the communication. Communication can be fast or slow, it can be directed in space or broadcast in all directions, sensitive to interferences, *etc*... For instance, radio communication is sensitive to interferences and it is hard to control in which direction it propagates. Interestingly, it is very fast and insensitive to a large number of physical obstacles. On the other hand, visual communication with cameras is not sensitive to interferences (at least not in robotics nowadays), and it is possible to localize quite precisely where a signal comes from. However, cameras are typically slow because they process a large amount of information, and communication is interrupted by almost any kind of physical obstacle.

Collective motion is an example of behavior that relies only on direct information transfer. As shown in Figure 2.5, in the simplest implementation of collective motion, each member of a group of robot uses direct communication to estimate the motion direction of its surrounding neighbors. Practically, robots may perceive the direction of motion of each other with visual cues, or by communicating in a peer to peer manner their orientation with respect to some reference (Turgut *et al.*, 2008). Each robot calculates the motion direction of its neighbors and gradually updates its own direction of motion to match it with the average perceived (Reynolds, 1987). This simple rule eventually leads the whole group to move in a coherent manner.

Indirect information transfer

Moving beyond the classical picture of communication, the study of self-organizing biological systems has shown that indirect information transfer can be at the core of stunning collective behaviors. Indirect information transfer has been first theorized by Grassé (1959, 1984), under the name of "stigmergy" (Bonabeau *et al.*, 1999).

While observing the collective construction behavior of termites, Grassé realized that individuals were not following a blueprint or a plan; rather, their activity was driven by the constructed structure itself. More precisely, Grassé observed that the construction behavior of the termites changed depending on the local density of soil pellets. If the density was low, individuals were just randomly dropping new pellets. But as soon as the pellet density was high enough, termites would build pillars, and if two of these pillars were close enough, termites would bridge them. Grassé coined the word "stigmergy" (from the greek, stimulating work), emphasizing that termites were directed by their work. While this initial vision has inspired a very large number of studies in natural and artificial systems, it would be misleading to consider stigmergy as a different kind of organization. Within the prevailing perspective of interacting components in a self-organizing system, stigmergy can be seen as indirect information transfer.

In nature, indirect information transfer is widespread. Ants, for instance, create pheromone trails from food sources to their nest and interact with each other through deposited pheromone (Hölldobler and Wilson, 1990; Deneubourg and Goss, 1989). Social wasps build their nest incrementally by adding hexagonal cells. New cells are built probabilistically, depending on the local configuration of previously created cells (Theraulaz and Bonabeau, 1995). Humans also rely extensively on indirect communication, for instance by leaving sticky notes in their environment. Lately, internet has become a central communication tool for us. The collective aggregation of information has become the hallmark of this technology. The most used websites all integrate methods to gather and synthesize pieces of information produced by their users. Examples of these websites include Wikipedia for collaborative editing or Amazon and eBay for reputation systems.

In robotics, indirect information transfer has frequently been implemented with a blackboard architecture (Hayes-Roth, 1985; Merino *et al.*, 2006), connecting all the members of a group to a single virtual place where they could interact freely. This technique involves a specific communication infrastructure to centralize information and is therefore not suitable for swarm robotics. Only few works in swarm robotics have relied on indirect information transfer. Two reasons could explain this lack: first, systems that rely on indirect interactions are difficult to design because the collective behavior that results from the interactions is often counter-intuitive and hard to predict without appropriate tools. Second, robots are often not able to modify their environment in order to transfer information, either because they lack appropriate effectors, or because it is not desirable to disrupt the environment. Several solutions to this problem are currently under study. For instance, robots may interact through sensor networks (Batalin *et al.*, 2004), or items in the environment could incorporate RFID tags that can be read and written by any compatible electronic device, including robots (Hahnel *et al.*, 2004).

Characterizing information transfer in self-organizing systems

To summarize and provide a unifying vision of information processing in self-organizing systems, we define information as a pattern imprinted by individuals in the environment. This pattern is propagated in space and/or time, and becomes relevant when it influences the behavior of one or more individuals, or when it is part of the outcome of collective behavior.

Noticeably, the collective behavior of the group and information transfers among members are both spatiotemporal patterns in the environment. Because of this compatibility, interactions and actions may be indistinguishable.

Our definition of information transfer is deliberately general so as to take into account innovative interaction techniques. Nevertheless, information transfers can be characterized in different ways.

- **Direct or indirect**: as described in the previous sections, information can be transferred almost instantaneously, or with a delay.
- **Explicit or implicit**: information transfer is considered implicit when the individual that creates information is not acting on purpose. A robot that perceives another one using his camera and a detection software relies on implicit information transfer. Conversely, if a robot lights up an LED in the dark to allow other robots to localize it, the transfer is explicit.
- **Abstract or situated**: in abstract communication, only the message has a meaning but the physical signal does not have any additional semantic properties (Støy, 2001). Conversely, situated communication means that both the contents of the message and the physical characteristics of the signal contribute to the meaning of the message (Clancey, 1997).
- **Local or global**: information can be available only in a limited area in the neighborhood of the individual or it can be available to all the individuals, whatever their spatial location.
- **Distributed, centralized, or broadcast**: information transfers can occur between two individuals, or from many to one, or from many to many. In fact, information transfers can be characterized both by the number of individuals that contribute to shape the information and by the number of individuals that are using the information.

2.3.2 Collective decision making

When a swarm makes a collective decision, it selects an option that meets desired characteristics. Contrary to the decision making techniques in classical artificial intelligence, a swarm does not have to make a choice internally and then carry out the corresponding action. In swarm systems the collective choice is the product of interactions, and as such it often has to be observed through a resulting collective action. Although there are several different ways to make a collective decision, **imitation** is the most common mechanism employed in swarm intelligence. In addition, **amplification** is an operation that helps focusing the swarm on a single collective choice.

Imitation

Imitation allows groups to synchronize their activity and reach a consensus. Taken in its broad sense, imitation consists of the replication of a particular behavior. From this perspective, it is the underlying fundamental mechanism at work in numerous collective behaviors that involve decision making.

Activity synchronization such as the flashing of fireflies (Buck, 1935; Winfree, 1967; Buck, 1988; Mirollo and Strogatz, 1990) is typically modeled with a very basic imitation mechanism (see Figure 2.6). Take a group of individuals which emit flashes at random times but with a common period. When an individual perceives the flash of a neighbor before its own, it flashes slightly earlier so as to shift it's flashes in time. Constant adjustments made by all the individuals lead to a global synchronization of the flashes. This simple mechanism explains the collective behavior of some species of fireflies in nature, and it is also used in swarm intelligence to synchronize networks or detect faults in groups of robots (Christensen *et al.*, 2009).

Aggregation is one of the most basic examples of self-organization observed in nature (Amé *et al.*, 2004; Jeanson *et al.*, 2004) and also studied in swarm robotics (Garnier *et al.*, 2008). As described in Figure 2.7, it can be seen as the collective outcome of imitation. Consider a group of robots with only two behavioral rules, moving and staying still, that alternate randomly. When a robot perceives a still robot, it imitates it by stopping its motion. Thus, it creates a cluster of two robots which, in turn, acts as a nucleus for a larger and more stable aggregate.

Collective motion, which was mentioned in the previous section, is commonly obtained by imitation of the local neighbors (see Figure 2.5) and can even be studied with tools similar to the ones used for synchronization (Sepulchre *et al.*, 2004; Paley *et al.*, 2007). The same mechanism can also be used to synchronize pulling direction in a task of cooperative transport as reported in Chapter 4.

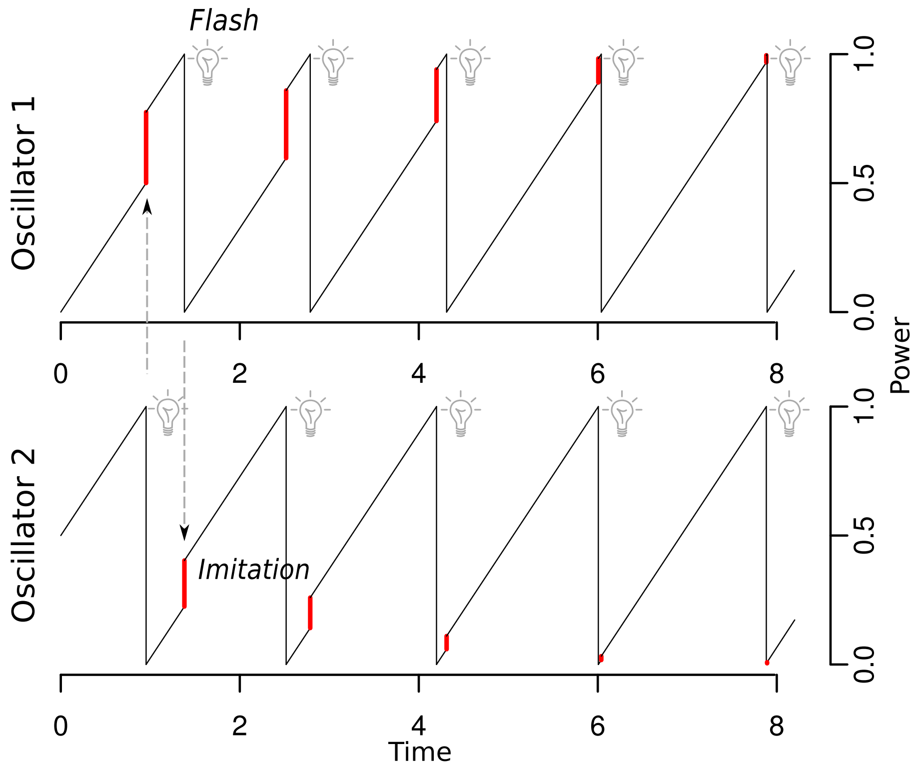


Figure 2.6: The synchronization of two oscillators can be achieved with simple imitation. The activity of two oscillators is represented as a function of time. Oscillators gradually accumulate energy until a certain threshold and when this threshold is reached, energy is suddenly released with a flash of light. When an oscillator perceives a flash from another oscillator, the first one tries to imitate the latter by flashing earlier (*i.e.* shifting its phase). Imitation is stronger, and the phase shift larger, as the difference between the two oscillators is more important.

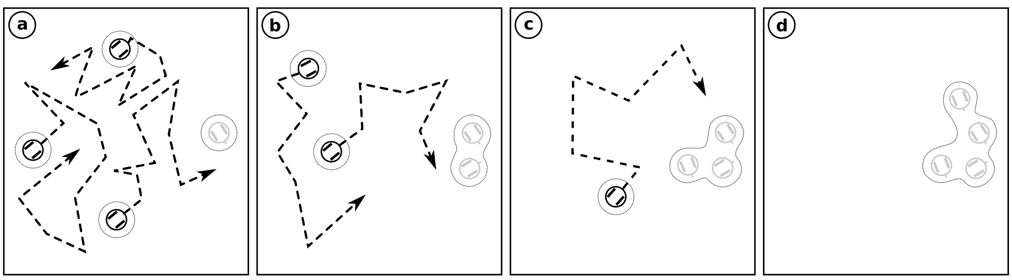


Figure 2.7: Aggregation is a fundamental problem studied in swarm intelligence. **(a)** Consider a group of robots moving randomly in their environment, and one additional robot that has stopped its motion. A first robot finds the stopped conspecific on its trajectory. **(b)** The robot performs imitation and stops its motion. In this way, a cluster of two stopped robot is formed. A second robot hits this cluster by chance. **(c)** Again, the moving robot imitates its stopped conspecifics and joins the cluster. Now the area of the cluster as gone larger and the third moving robot encounters it faster. **(d)** All the robots are aggregated in a compact cluster.

Last but not least, quorum sensing is a type of collective decision making in which a synchronized collective behavior appears when the density of individuals goes beyond a certain threshold. Quorum sensing was first observed in the *Vibrio fischeri* bacteria (Nealson *et al.*, 1970). These cells can synthesize a light emitting enzyme known as luciferase. To make a noticeable effect, luciferase must be mass produced. For this reason, single cells avoid wasting energy on the production of low amounts of this enzyme. It is only when cell density is important enough that the bacterias coordinate with signaling molecules in order to produce luciferase. Moreover, this mechanism has also been used to explain nest site selection in ants and bees (Pratt, 2005; Seeley *et al.*, 2006).

The conditions in which individuals imitate each other have important consequences on the collective decision made by the group. Various flavors of imitation notably include probabilistic imitation as a function of a level of confidence, or as a function of the number of other individuals already in a given state.

Amplification

Amplification (Bonabeau *et al.*, 1999; Camazine *et al.*, 2001) is inherent to imitation mechanisms. Aggregation provides a clear example of this phenomenon (see Figure 2.7). The probability for a single robot to stop its motion near other already stopped robots grows with the numbers of stopped robots. This is because they form a cluster that occupies a growing area, which in turn increases the probability that potential imitators detect the stopped robots and join the cluster. Hence, the time to form an aggregate is not proportional to the size of the aggregate, it rather grows in a sublinear manner.

In some cases, this basic amplification is not sufficient to trigger a collective choice. For instance, when a group must aggregate under one of two identical circular shelters (Garnier *et al.*, 2005; Amé *et al.*, 2006), a naïve implementation of imitation leads the group to split evenly under each shelter. This is because larger groups have also larger chances to lose members. Since individuals have the same probability to enter any shelter, the system stabilizes around an even distribution in the shelters.

This phenomenon can also be seen from the perspective of dynamical systems (Strogatz, 2000). If all the interactions among the individuals are modeled by linear relationships, the system can be described by a set of linear differential equations (of order 1), and it has a single stable state. In the shelter experiment, this stable state is not a collective decision, rather a split of the group. To enforce the cohesion of groups, the rate per second of individuals leaving a group must be a non linear function of the aggregate's size. In that case, the system has at least two stable states in which the group cohesively aggregates under either shelter.

Amplification is therefore a critical part of imitation, and it is often (but not always) non linear in order to trigger cohesive collective decisions.

2.4 Collective decision problems in the foraging task

In this thesis, we focus on the foraging task and use it as a framework to study collective decisions in self-organizing groups of robots. By examining several different situations, we aim at identifying fundamental mechanisms at work when robot groups make collective decisions.

Foraging, the act of searching for resources, is very commonly studied in collective robotics and it has been outlined as a canonical testbed in the review paper of Cao *et al.* (1997). Robot foraging is the generalization of many real world tasks and solving problems related to foraging is synonymous to making advances in practical applications (Winfield, 2009).

In the following, we first provide a general overview of the foraging task, then we proceed by reviewing the literature related to the different specific foraging problems we studied.

2.4.1 The foraging task as a framework

Envisioned applications

The real world tasks that can benefit from advances in robot foraging notably include agriculture, mining, toxic waste cleanup, search and rescue, and exploration of unknown or hostile environments (see Figure 2.8).

- **Agriculture** involves a number of tedious works such as harvesting, weeding, and pruning. These works require a large number of accurate gestures that must be carried out at precise moments to optimize production (Sarig, 1993; Monta *et al.*, 2002; Pedersen *et al.*, 2006).
- **Mining** is a dangerous and exhausting task which is already very mechanized, for instance with load-haul-dump trucks. A further and logical step is the full automatization of this task with groups of autonomous robots (Roberts *et al.*, 2002).
- **Cleaning** of areas contaminated by toxic material such as asbestos could be carried out by robot swarms. Land mines can also be detected and neutralized with robots (Rachkov *et al.*, 2005). Robots are also used for indoor cleaning (Prassler *et al.*, 2000; Endres *et al.*, 2002). Another direction explored is the inspection of in-

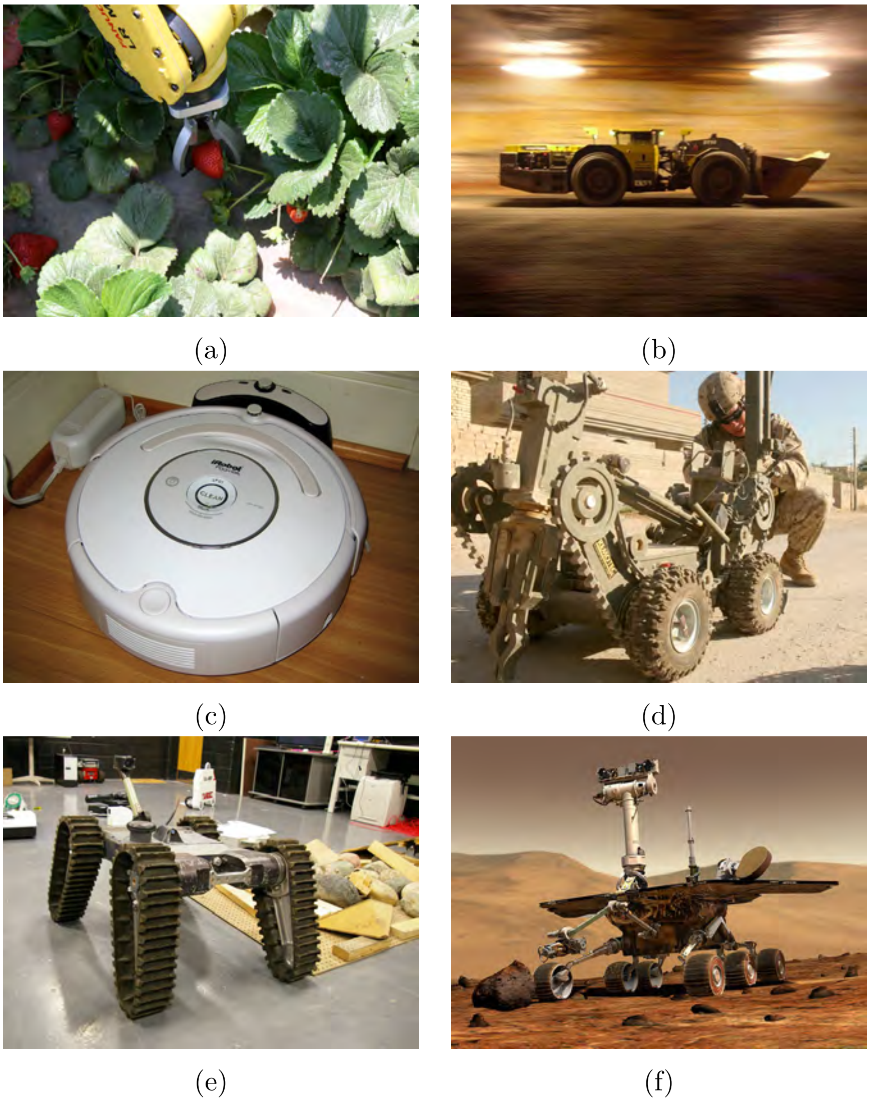


Figure 2.8: Examples of robots that may rely on a foraging behavior to perform real world tasks. **(a)** Fruit picking robot created by Robotic Harvesting LLC. **(b)** Remote controlled Load-Haul-Dump (LHD) truck used in underground mining. **(c)** Vacuum cleaner robot created by iRobot. **(d)** Neutralization of improvised explosive devices (IEDs) with a robot. **(e)** Chaos, a search and rescue robot from the TARDEC of USA. **(f)** Opportunity rover sent to mars by the NASA.

accessible places such as jet turbine engines (Correll and Martinoli, 2006) or old underground mines (Thrun *et al.*, 2003).

- **Rescue** is another task that could benefit from the advances of collective robotics in foraging. Search and rescue after an accident is dangerous for the workers and the disrupted environment makes localization of injured people particularly difficult. Robots may assist rescuers in their tasks, exploring dangerous or unattainable places (Birk and Carpin, 2006).

- **Space exploration** already benefits from the most basic advantage of collective robotics, namely redundancy. Because the Mars Exploration Rover mission of NASA involves two robots rather than a single one, the mission is still ongoing today and the Opportunity rover is still in function while the Spirit rover became silent in march 2010. In future missions, the benefits and lifetime of the mission could be extended by having robots rescue each other, mutually clean their solar cells or repair damaged mechanics.

Categories of foraging

Foraging in collective robotics can be classified in different categories, as a function of the task that the robots must accomplish. In **on-the-spot foraging**, robots must explore their environment to find resources. When a resource is found, they can exploit it immediately, that is, on the spot. An example of this type of foraging is given by autonomous energy management: robots are looking for an electrical plug and when they find it they recharge their batteries immediately (Silverman *et al.*, 2002; Zebrowski and Vaughan, 2005; Wawerla and Vaughan, 2008). In **central place foraging**, the robots can not exploit resources directly on the spot. Rather, they have to retrieve it to a specific location, often referred to as the central place. Examples of such tasks include cooperative transport and waste cleanup (Parker, 1998; Groß *et al.*, 2008). One last type of foraging described here is **multi-foraging**. In this task, the robots can find different types of resources, and they have to decide which are the most interesting to exploit (Balch, 1999; Campo and Dorigo, 2007). These categories cover all the works reported in this thesis, but additional distinctions can be made. A more exhaustive taxonomy of the foraging task is proposed by Winfield (2009).

Fundamental underlying problems

The foraging problem is not only useful for industrial applications in the medium term; it is also a very meaningful case study for fundamental problems at the core of collective and swarm robotics. The problems that can be studied via a foraging task with a robot swarm include navigation and exploration, decision making, and task allocation.

As a general rule, foraging can be decomposed into several subtasks. First, robots must explore their environment to localize resources. Second, they must select the resources and decide which ones to exploit. In an optional third step, the robots may process the resource, for instance by transporting it to a central place. Fourth and last, the robots can exploit the harvested products.

Navigation, and exploration strategies are necessary to perform the subtask of resource localization. In the case of central place foraging, robots may have to travel repeatedly to the resources. Techniques such as odometry (Borenstein and Feng, 1996; Fox *et al.*, 2000) or path creation (Werger and Matarić, 1996; Nouyan *et al.*, 2006, 2008) may be useful to help robots navigate effectively in the environment. Collective decision making allows robot swarms to select which resources should be exploited, and which ones should be neglected (Garnier *et al.*, 2007). Finally, task allocation can be implemented to use only the necessary and sufficient number of robots and avoid overcrowding, which would reduce performance (Goldberg and Matarić, 1997; Labella *et al.*, 2004; Frison *et al.*, 2010).

2.4.2 Practical foraging problems considered

Resource transport

In some cases, resources are not exploited on the spot; rather, they are constituted of objects that have to be transported somewhere. A classical instance of the cooperative transport problem is central place foraging, in which robots must gather harvested objects in a central place. When one of these objects is too large or heavy, a single robot is not enough to accomplish the transport. Several robots must then cooperate to transport it. A number of problems need to be solved to perform this task successfully. The coordination of the movement of the robots is one of them. This problem has been investigated by Groß *et al.* (2006b), in situations in which either all or some of the robots are able to perceive the central place. In fact, it is quite rare that robots perceive directly the place they must reach. This is because the distance between a resource and the central place is often large, or because the sensors of the robots are limited. When all the robots have completely lost sight of the central place, it is not possible anymore to rely on first hand information. Nevertheless, robots are not totally ignorant, and it is safe to assume that they have partial knowledge of the goal direction. For instance, they may have perceived the central place earlier and kept track of its direction by means of odometry (Fox *et al.*, 2000).

Odometry is achieved using internal, proprioceptive information (Borenstein and Feng, 1996) (*e.g.*, by measuring the rotation of the wheels of a robot). The information on the movement of a robot is integrated, thus the error made on localization increases with the distance covered. In our case, this leads to an erroneous indication about the direction of the nest. If several robots attempt to transport a heavy object in different directions they may fail to move the object at all. Therefore, a mechanism of coordination is required so that robots negotiate in which direction to pull the object. This problem is of particular interest to us because it requires robots to make a collective decision about the direction in which they transport the object.

Models of collective motion have long been studied by biologists to understand how fish or birds move together, avoiding collisions among each other and deciding in which direction to go (Aoki, 1982; Huth and Wissel, 1992; Couzin *et al.*, 2002). Physicists have also proposed models of collective motion to study the transition from disorder to order, when individuals in groups share the same orientation (Vicsek *et al.*, 1995). The models available in the literature are usually composed of three behaviors: an attraction behavior that makes the individuals stick together, a repulsion behavior that prevents collisions among individuals, and an orientation behavior that coordinates the individuals' motion (Reynolds, 1987).

However, the orientation behavior is the most challenging to implement in robots because it necessitates a specific type of sensor. This behavior is commonly implemented in a robot as the imitation of the surrounding neighbors' orientations. Imitation can only take place if robots are able to sense each others' positions and orientations. Sensors that provide robots with this information are called range and bearing sensors (we propose two different implementations with cameras and infrared sensors in Chapter 3). Several studies in robotics have reported implementations of flocking using a simulated range and bearing sensor (Hayes and Dorminiani-Tabatabaei, 2002; Regmi *et al.*, 2005). These works demonstrated the potential of this type of sensors, and at the same time the lack of any prototype underlined the difficulty to implement them in physical robots.

Physical implementations of range and bearing sensors have been reported by Kelly and Keating (1996) and Kelly and Martinoli (2004), but in these works the sensor is not used to produce the orientation behavior. Turgut *et al.* (2008) have demonstrated flocking on real robots using a range and bearing sensor implemented with infrared sensors and a compass to get a common reference frame for all the robots. These results show that the orientation behavior, part of the flocking collective behavior, may be transposed to robot swarms so as to provide them with a valuable decision mechanism for cooperative transport.

Resource localization

In central place foraging, robots must go back and forth between resources and the central place. When these locations are distant, or when the perception capabilities of the robots are limited, robots may not use directly observable information to navigate from one place to the other. In this context, robots must solve a localization problem so as to find out where they are in the environment and carry out their task. Several different solutions to the localization problem have been implemented.

For a single robot, localization has been commonly achieved with probabilistic methods that use data from odometry and map-like representations of the environment (Simmons and Koenig, 1995; Cassandra *et al.*, 1996; Burgard *et al.*, 1996, 1998; Gutmann *et al.*, 1999, 2001; Chong and Kleeman, 1997; Wang, 1988). Most of these approaches are based on probabilistic methods which make use of dead-reckoning and absolute or relative measurements. The key idea is that each robot maintains an estimate of its position which will be updated according to its odometry calculations and to its measurements of the environment. The most used probabilistic method has been the Kalman filter (Kalman, 1960; Larsen *et al.*, 1998; Martinelli and Siegwart, 2003). Although the Kalman filter is an efficient recursive filter, it requires external information that models the environment. Moreover it is computationally costly. In addition, map-like approaches (Dudek and Mackenzie, 1993) do not scale well for large groups of robots, and are also costly in computational terms. Last but not least, odometry (Borenstein and Feng, 1996; Fox *et al.*, 2000) is probably the most used mechanism as it provides easy and cheap real time position information by the integration of incremental motion information over time. Unfortunately, this integration causes an accumulation of errors during the movement of the robot.

In collective robotics, localization is typically achieved by taking inspiration from social insects. However, most of the implementations of localization in swarm robotics rely on the use of static robots which are not allowed to move, thus reducing the effectiveness of the group. In Kurazume *et al.* (1996), part of the robots remain stationary while the rest are in motion. The moving group stops after several steps and becomes a landmark for the others that take on the role of moving robots. In Grabowski *et al.* (2000), only one robot is allowed to move while the others act as immobile landmarks, while in Rekleitis *et al.* (2001) just one robot remains stationary while the rest of the group navigates. These approaches either require synchronous communications between all members of the team or a central processing unit taking care of the synchronization. In Vaughan *et al.* (2002), each robot shares with a central computer its best-known path to the goal based on landmarks. The group has to be permanently in contact with the data center which forwards the information provided by each robot to the other members of the group. Therefore, the environment must be properly configured before setting up the experiment. In Rekleitis *et al.* (2003) the robots explore the environment in teams of two. Each team of two robots take turns moving so that at any time one is stationary and acts as a fixed reference point to the robot which moves. Each robot is equipped with a robot tracker sensor that reports the relative position of the other robot. These position measurements are used to update the positions and uncertainties of a multi-robot system. In Szymanski *et al.* (2006) the authors implemented a distributed algorithm that finds the shortest path between two goal areas within a labyrinth. The robots need to create a virtual chain and be static until the collective decision emerges. In Nouyan *et al.* (2008), a chain between two specific areas is created and the group can follow it to the goals. The robots in the chain are not able to move; consequently a direct relationship exists between the number of available robots and the maximum distance between the goals.

The aforementioned localization or navigation strategies have a number of different limitations: they are power consuming in terms of computation because of the Kalman filters and use of maps, some robots are not allowed to move while others are tracking distance between them, robots must maintain visual contact at all times with the rest of the group, and in some cases robots have to communicate with a central device to update or download maps, synchronize movements, or update positions. One way to solve these problems is to take advantage of the redundant information available in groups of robots. Indeed, each individual has its own estimates of the foraging locations. By communicating these estimates, the robots may localize resources more efficiently and, in turn, improve their foraging capabilities.

Resource selection

A number of works relied on path creation to keep track of the locations of the resources and of the central place (Werger and Matarić, 1996; Nouyan *et al.*, 2006, 2008). When creating a path, the robots literally form a chain from one place to another marking in this way a path. The main limitation of path creation lies in the number of robots used to mark the path and that can therefore not be used for other tasks. When there are several resources in the environment, the number of robots used on the different paths can be reduced by carrying out a path selection.

Selecting which path to maintain to reach a specific resource means allocating less robots to marking, as the robots used in the no longer maintained paths are free to perform different tasks. For instance, the exploitation of a resource may require a minimum number of robots to be successful. This is the case, for example, when retrieval of items involves cooperative transport (Groß and Dorigo, 2009; Nouyan *et al.*, 2009b); if robots try to exploit several resources at the same time, they may end up in a deadlock situation in which there are too few robots assigned to each resource and none of them can be exploited.

Hence, when robots use paths to navigate in their environment, it is interesting to implement the selection of the path to the most profitable resource, with the possibility for unused robots to stop participating in path maintenance and switch to another task. This sub-problem of foraging requires that robots make a collective decision about which resource to exploit. Robots should choose the path to the most profitable resource, and when offered several identical resources, we expect robots to arbitrarily focus on a single resource.

A number of strategies that improve foraging by implementing path selection have been applied with remarkable success. Payton *et al.* (2001) proposed a bio-inspired mechanism making use of virtual pheromone to form a distributed computing mesh made of robots. This mechanism creates gradients of virtual pheromone from the central place to the resource and vice-versa. To do so, robots that are part of the mesh relay hop-count messages to each other. The robot at the central place emits a message with a given hop-count number. Upon reception of a message, the hop-count number is decreased and the message is relayed to neighboring robots. The robot at the resource also emits messages to create a second pheromone gradient from the resource to the central place. In this way, robots that are not part of the mesh are able to move to the two locations following the shortest path. However, robots relaying messages are unable to distinguish if they are useful to mark out the shortest path or not. Therefore, they cannot switch to another task if necessary and are doomed to stay in place. Moreover, when there are two identical resources in the environment the mechanism does not allow a collective choice to take place. Most likely the robots would go indifferently to one or the other resource.

Szymanski *et al.* (2006) proposed a novel mechanism to let the robots know when they contribute to mark out the shortest path. The robots at the central place or at the resource emit messages that propagate through the swarm as waves. The messages are composed of two elements: the number of hops done and the estimated number of hops left to reach the destination. Upon reception of messages, robots update their smallest hop distance to each location. After a while, robots at the central place and at the resource have received at least one message that travelled through the shortest path. Therefore, they send messages with the minimal number of hops to reach the destination. Robots of the swarm also know their hop distance to each location and can compare it with the information of the messages relayed to determine if they are on the shortest path or not. This distributed behavior allows the robots to find the shortest path from one place to the other. More importantly, the robots know when they are on the shortest path and they can switch task and participate in foraging instead of remaining along a useless path. The main limitation of this mechanism is that it cannot select among two identical resources. In addition, it seems very sensitive to transmission errors. An error in a single message is enough to prevent the whole swarm from finding the shortest path.

Schmickl and Crailsheim (2008) introduce a foraging mechanism inspired by the trophallactic behavior of bees. Robots forage in a closed arena, with one resource available and a central place. When they reach the resource, robots receive a load of virtual nectar that they gradually distribute to encountered neighbors. The peer-to-peer transmission of nectar acts as a diffusion process which creates a gradient of virtual nectar inside the population of robots, between the resource and the central place. When foraging, robots simply orient themselves using this gradient. The results of the algorithm are quite promising, but its domain of application differs from chain based systems as it may only work with a closed arena and a number of robots large enough to cover the surface of the arena in order to establish a nectar gradient that can be followed by the robots. Furthermore, the system has been tested in simulation exclusively and is based on assumptions such as bi-directional infrared communication during robot motion with spatially directed sensors. Therefore, a real robot implementation seems necessary to fully understand the benefits of this approach.

Garnier *et al.* (2007) took a completely different direction. Instead of creating another variation of the Bellman-Ford algorithm (Bellman, 1958), they simply transposed the pheromone laying behavior of ants on Alice robots (Caprari *et al.*, 2002). As it is particularly difficult to implement pheromone sensors and actuators on robots, the authors substituted pheromone with light projected on the ground thanks to a video projector as proposed by Sugawara *et al.* (2004). The mechanism proves to be successful and the robots collectively select a single path to the resource. Because of the strong similarity with ants' behavior, we can safely assume that robots could also select the shortest path to the most profitable resource, for instance the closest one. However, using this mechanism outside laboratories seems difficult for the moment as there is no cheap and miniaturized technology that would allow robots to deposit and perceive pheromone reliably (Russell and Norvig, 1995; Stella *et al.*, 1995; Russell, 1999; Svennebring and Koenig, 2003; Purnamadjaja and Russell, 2004).

Resource discrimination

Foraging robot swarms which are able to select between several resources and perform collective decisions are often designed with the successful examples of ant colony optimization or particle swarm optimization in mind (Dorigo *et al.*, 1996; Dorigo and Şahin, 2004; Poli *et al.*, 2007). For instance, systems inspired by the pheromone laying behavior of ants are typically used to identify the shortest route in a network (Di Caro and Dorigo, 1998; Sugawara *et al.*, 2004; Garnier *et al.*, 2007). Bee inspired systems try to locate the best resource available in the environment (Pham *et al.*, 2006). In the literature, a basic imitation mechanism is frequently pointed out as the key factor that triggers collective decision making in groups (Camazine *et al.*, 2001; Garnier *et al.*, 2007). With this mechanism, the probability for an individual to select a given option or resource increases superlinearly with the number of conspecifics that select the same option. This creates a positive feedback as more and more individuals tend to make the same choice, and eventually leads to a consensus between members of a group. Such systems are often bound to choose an option that minimizes or maximizes a certain characteristic.

However, more is not always better. If the robots are not alone in their environment, large and rich resources are more likely to attract competitors, adding an extra cost for the protection of the resource (Nagamitsu and Inoue, 1997; Holway and Case, 2001; Lichtenberg *et al.*, 2010). Moreover, robots may be forced to spread over a larger space to occupy the whole resource, hence impairing intra-group cooperation or reducing the benefits of working in group (Krause and Ruxton, 2002; Sumpter, 2010). In all these situations, it is more advantageous for groups to select resources that correspond closely to their needs, and to avoid oversized ones. But this task requires to evaluate the overall needs of the group in addition to the capacity of the available resources. This may be particularly difficult to achieve with a large population, or if individuals have low cognitive abilities (Seeley, 2002).

In this context, collective discrimination can be seen as a first step towards more complex collective decisions mechanisms. Using collective discrimination, a group may be able to select a resource based on a desired characteristic. This mechanism may be useful in a number of situations. For example, when a group of robots retrieves an object, they may benefit from transporting larger objects that yield more reward than smaller ones, but the group should also avoid large objects that cannot be transported by the whole group. Another example is the one of two independent groups that are searching for a resource. If they both try to find the best resource, competition and overcrowding may hamper their ability to exploit the resource (Adams, 1990; Nagamitsu and Inoue, 1997; Matarić, 1997; Goldberg and Matarić, 1997; Holway and Case, 2001; Smith *et al.*, 2008; Lichtenberg *et al.*, 2010).

To the best of our knowledge, collective discrimination has not yet been demonstrated in swarm robotics. Valuable foundations for the study of collective discrimination are provided by the model of Amé *et al.* (2006). This model is used to explain the collective choice behavior of cockroaches when they select one shelter out of several identical ones. Amé *et al.*'s model is based on the assumption that the probability for a cockroach to leave a shelter decreases with the density of individuals in the shelter. This behavior produces aggregation at the collective level, and imitation is the underlying process at work. The model predicts that, when each shelter is sufficiently large to house all the cockroaches, the group will aggregate in only one of them. If shelters are too small, the model predicts that the group will use two or more shelters equally.

In their mathematical model, the authors made the hypothesis that cockroaches have access to an estimate of the density of conspecifics present under a shelter. While this hypothesis is only a detail in that modelization work, any attempt to use the model in swarm robotics prompts us to find a practical implementation for robots. As a matter of fact, evaluating the local density of robots at a resource may not be a simple task. A naïve method to estimate the density of robots in a region would consist in calculating the surface of the region and counting the number of individuals present in the region. This method is not feasible in the context of swarm robotics because the region may be too large to be entirely explored, and because counting robots requires to find them all and draw up a list to avoid multiple detections. A recent study of emigrating ants *Temnothorax albipennis* suggests an original way to estimate the local density of conspecifics (Pratt, 2005). These ants rely on the rate of encounters with other ants to evaluate the density of individuals in a cavity: the more contacts they have with other ants, the greater their estimated value of the density. This mechanism may also be used by robots to estimate local densities of conspecifics.

These works provide the basis for a new experiment in swarm robotics. Shelters can be seen as resources for cockroaches and the shelters' surfaces correspond to the capacities of the resources. The total surface of the cockroaches' bodies corresponds to the group needs. Building on this equivalence, it is possible to transpose the study to foraging robot swarms. More precisely, this defines a task of foraging in which robots exploit resources on-the-spot, and must make a collective decision about which resource to exploit.

Chapter 3

Material and methods

3.1 Robotic Platforms

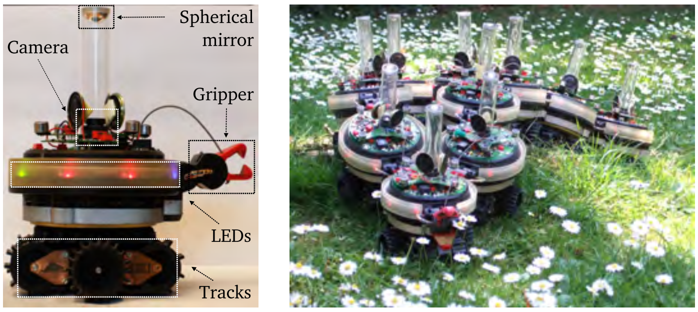
In our robotics experiments, we have used two different types of robots, the s-bot and the e-puck robots.

Demonstrating our results with physical robots constitutes the ultimate validation step of our studies. There are many disadvantages to work with physical robots: we have to recharge them, or manually change the batteries. Robots, and especially our research prototypes, may break or wear out with time. Also, sensors and actuators may behave differently as time goes on. Finally, robots typically run much slower than any computerized model.

Nevertheless, demonstrating our results with real robots is an important step because models are always a simplification of the real world and we can only have a limited confidence about their correctness. So-called realistic models attempt to incorporate as much details of the reality as possible. However, as J. L. Borges underlined in his poem "Del rigor en la ciencia", attempting to make a model that perfectly matches reality in all its details is a waste of time because models never capture all the subtleties of the physical reality. Adding more details and features to models only brings them closer to reality, but at the same time it also introduces the risk of making wrong assumptions, and of spending valuable time on the design of the model rather than on the study of the phenomenon.

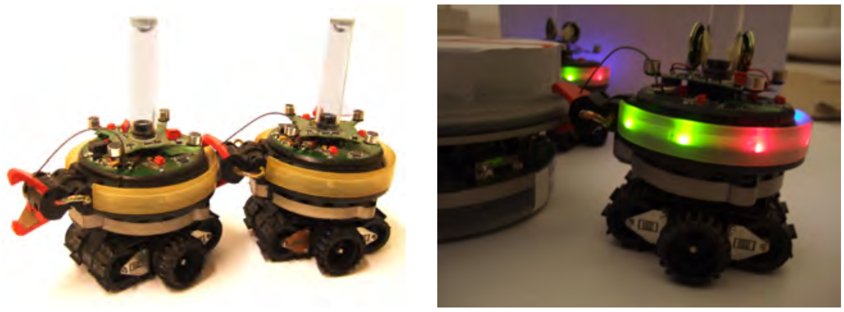
A model is a tool designed to study specific aspects of a phenomenon. This lack of generality is an intrinsic limitation but it is also the strength of models. Because they are simplified, they allow to focus the study and understand mechanisms. However, there is no guarantee that the assumptions and simplifications made still hold in reality. The only way to know is to make experiments with real robots.

To give practical examples of wrong assumptions that are commonly made in models, robots sensors are typically modeled by adding noise to the measures. In reality,



(a) (b)

Figure 3.1: (a) The s-bot robot. **(b)** A swarm-bot staged outdoor.



(a) (b)

Figure 3.2: The gripper of the s-bot. **(a)** Robot to robot connection. **(b)** Physical connection with an object to transport.

when a robots remains still, the sensors can return a non-noisy, constant output. Another example is the motion of wheeled robots, which is typically modeled with kinematic equations that do not take into account possible slippage of the wheels. The consequence of this simplification is that odometry can be very effective in simulations and produce poor results in reality.

Hence, our methodology consists in using robots to demonstrate the main results of our experiments and simulations to extrapolate further results on the properties of the mechanisms studied.

3.1.1 Swarm-bots

The s-bot (see Figure 3.1) is a robot of 12 cm of diameter, designed and built within the context of the SWARM-BOTS project (Mondada *et al.*, 2004; Dorigo *et al.*, 2004; Dorigo and Şahin, 2004). This robot is notably equipped with a gripper that allows the study of self-assembly and associated problems with a super entity called the swarm-bot and made of assembled s-bots. The s-bot comes with a 400 MHz XScale CPU and runs a micro Linux operating system.

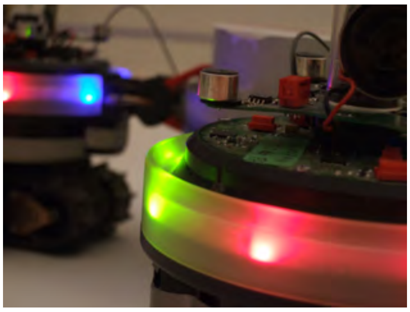


Figure 3.3: The LED ring of the s-bot, made of 8 groups of red, green, and blue diodes.

The s-bot moves using a combination of two wheels and two tracks, which are called "treels". This system provides differential drive motion to the robot, and allows the robot to navigate effectively on rough terrains. The treels are connected to the chassis, on which a cylindrical turret is mounted by means of a motorized joint. The turret can rotate with respect to the chassis, allowing to the robot to grip objects in directions unrelated to its motion. This particular feature is essential to collective transport tasks and is an important feature of the robot. The gripper is mounted on the turret and can be used to perform physical connections (see Figures 3.2a and b). The T-shape of the gripper matches the shape of the ring placed around the s-bot body, so as to allow robot to robot connections. Any other object equipped with the appropriate handle can be gripped by the s-bot. The gripper opens and closes, and can tilt to carry out tiny adjustments or slightly lift the gripped item. Inside the claws of the gripper, an optical barrier composed of two leds and a light sensor allows the detection of gripped objects.

With 15 infrared sensors distributed around its body, the s-bot can detect obstacles and navigate safely. Four proximity sensors (also called ground sensors) are placed under the chassis and can be used for the perception of holes or markers on the ground. In addition, the s-bot has an omni-directional camera mounted on the turret. The camera can be used to perceive colored objects up to a distance of approximately 50 cm. Using 8 groups of red, green, and blue LEDs evenly distributed around its body the s-bot can display colored patterns that can be perceived by other robots with their camera (see Figure 3.3).

The s-bot robot is rather complex and comes with a large set of actuators and sensors. The above description is not exhaustive but sufficient for the purposes of this thesis (the interested reader may consult Mondada *et al.*, 2004).

3.1.2 E-pucks

E-pucks (see Figure 3.4) are modular, robust, and inexpensive robots designed by Michael Bonani and Francesco Mondada (Mondada *et al.*, 2009). The primary purpose of these robots is to be used in research and educational projects. They are wheeled cylindrical

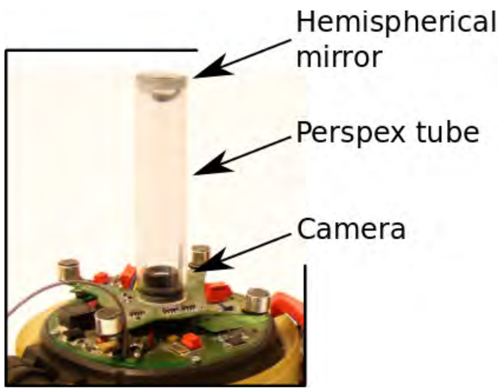
(a) (b)

Figure 3.4: (a) The e-puck robot. **(b)** A group of 10 e-puck robots.

robots, 7 cm of diameter, equipped with a variety of sensors, and whose mobility is ensured by a differential drive system. The central processing unit of the e-puck is a dsPIC 30F6014A produced by Microchip.

On the actuators side, the e-puck can use its wheels to perform mechanical actions, and LEDs, infrared emitters, a speaker and a bluetooth antenna to send signals. The wheels are set in motion by step motors that take 1000 steps to perform one wheel revolution. The internal measure of displacement (odometry) on the e-puck is therefore rather accurate, especially compared to the s-bot. The e-puck has 8 red LEDs evenly disposed around its body and it can light them up independently. There are two more LEDs, a green one inside the transparent body and an orange one on the front of the robot, but their purpose is rather debugging than communication. Robots can emit infrared signals using eight emitters disposed approximately evenly around its body. Lastly, a bluetooth antenna allows the robot to establish wireless radio communication with up to any 8 other bluetooth devices at the same time. On the sensors side, the e-puck is endowed with a frontal color camera, infrared sensors, the bluetooth antenna, an accelerometer and 3 microphones. We have mostly used the infrared sensors to detect obstacles and receive encoded messages from other robots.

There are several advantages in using the e-puck platform. First, the robot is extremely reliable, robust, tested and cheap ($\approx$ 500 euros). Given the very restricted market (less than 5000 e-pucks produced up to now) and the limited founding available for theoretical studies in robotics, customizing or repairing a robot can be very costly. Because the e-puck is open source, it is more effective to do this internally, without resorting to the services of third-party companies. In particular, the e-puck robot is extensible and accepts extension boards that are integrated via an UART or I2C bus. This allowed us to develop a range and bearing communication system for the e-puck, which is used in several works reported in this thesis. We could also replicate the ground sensor extension and successfully use it in our experiments.



(a)

(b)

Figure 3.5: The omnidirectional vision of the s-bot. **(a)** Close-up on the three components of the system **(b)** During an experiment, surrounded by several other robots at various distances and with red LEDs turned on.

3.2 Communication systems

3.2.1 Visual range and bearing

We used the omnidirectional camera and the LEDs of the s-bot to develop a visual range and bearing system. Robots are able to detect each other and estimate their position by analyzing the pictures captured with the omnidirectional camera. Robots advertise their position by emitting colored light with their LEDs. These lighted up robots appear as colored blobs in the pictures. After a series of image processing operations and geometric transformations, the neighboring robots are located. Furthermore, robots are able to estimate the orientation of their neighbors if these ones light up their LEDs according to a specific pattern. We empirically found that a triangle made of three colors was the most robust pattern to achieve reliable detection. Pattern recognition was achieved with an original software designed for speed and accuracy.

Hardware

The s-bots' omnidirectional camera (see Figure 3.5a) is composed of a USB camera pointing upwards and inserted inside a perspex tube that is 85 mm long. At the other extremity of the perspex tube, there is a hemispherical mirror which allows the camera to capture panoramic images. The camera of the robot is capable of capturing up to 30 pictures per second. Captured pictures have a resolution of 640 by 480 pixels in 24-bit Truecolor. The camera is connected to the motherboard of the robot via a USB bus, and controlled by the ov511 device driver for Linux. The output of the camera are standard jpeg pictures.

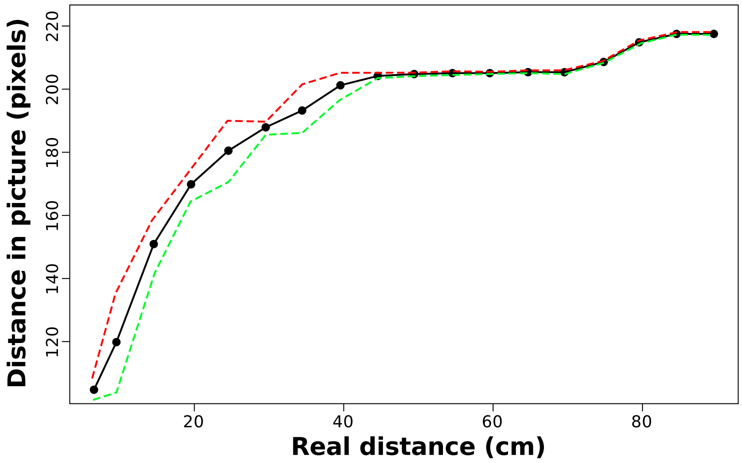


Figure 3.6: Relationship between the physical distance (in cm) of a red object and its distance to the center of the image (in pixels) captured by the omnidirectional camera. The solid line represents the average distance, surrounded by dashed lines that show the minimum and maximum measures.

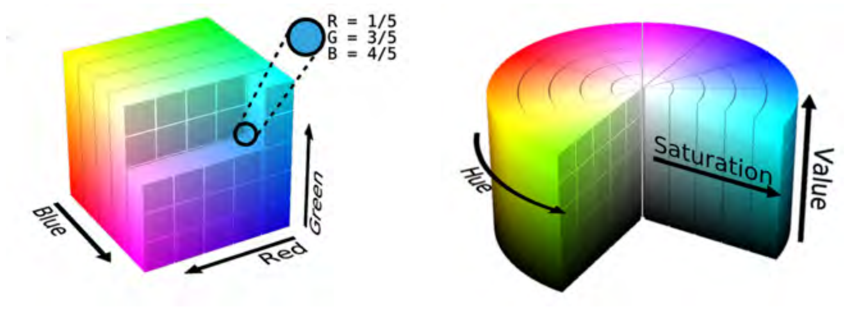
An example of such picture can be seen in Figure 3.5b. With respect to the human vision, this type of representation differs notably by the fact that it spans all around the robot, and that the physical distance of an object is directly related to its distance to the center of the panoramic picture (see Figure 3.6). The shape of the mirror makes distant objects smaller in the picture. For this reason, the detection of objects distant 50 cm or more is unreliable.

In Figure 3.6, we report the perception quality as a function of the distance of a red object in our experimental room, the accuracy of the camera decreases quickly with the distance.

Software

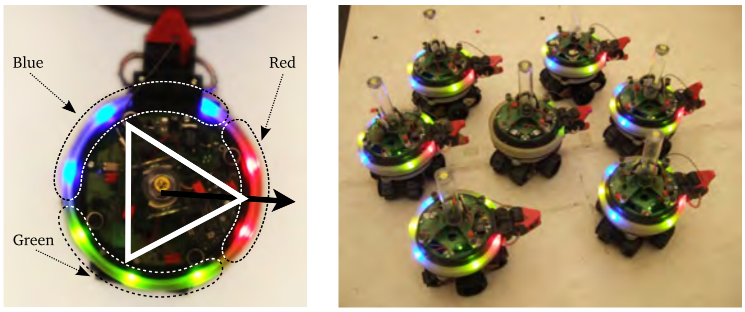
The software part of the visual range and bearing system works on the images captured by the omnidirectional camera. To simplify visual perception, we only use the colors produced by robots LEDs in the pictures. To do so, we start by converting the pictures from the RGB to the HSV color space (Smith, 1978). In HSV space, the color of a pixel is represented with a single dimension called hue (see Figure 3.7). This transformation is a classical step in image processing that facilitates color based manipulations of pixels (Foley *et al.*, 1994). Once the pixels are converted, we apply a simple color segmentation filter to identify the regions of the image that contain colors emitted by the robots. We call these colored regions "blobs".

At that point, a robot perceives a list of blobs of different colors, in different places. There are two main obstacles to detect neighboring robots and their orientation with this data. First, it is not possible to differentiate a single robot from two robots emitting



(a) (b)

Figure 3.7: Representation of the color spaces we used. **(a)** The RGB color space, which does not allow to characterize the color of a pixel with a single value. **(b)** The HSV color space makes the hue, lightness, and saturation linearly separable, allowing to identify the color of a pixel with a single value.



(a) (b)

Figure 3.8: The colored triangle pattern is used to let robots advertise their position and orientation to their neighbors. **(a)** The pattern on a single robot. **(b)** Setup with one robot in the center, perceiving its surrounding neighbors.

the same color and being side by side. This is because the two robots are perceived as a single large region of a same color. Second, it is not possible to detect the orientation of the robots if they are detected as a single blob.

To solve this problem, we propose to use a visual colored pattern that robots can display with their LEDs. In this way, neighbors are perceived as a set of blobs with a particular spatial arrangement. Notice that the back side of neighbors is occluded in the pictures of the camera, which implies that the visual pattern is never fully detected. During preliminary experiments, we found that the best pattern is a triangle made of three different colors (see Figure 3.8a). Patterns with two colors suffered from the back side occlusion, while the detection of patterns with four colors or more was too computationally expensive.

The last part of our software is concerned with the extraction of robots positions and orientations from the list of detected blobs. In Figure 3.9, we show an image of the robots along with graphs reporting the blobs detected by the central robot. The main

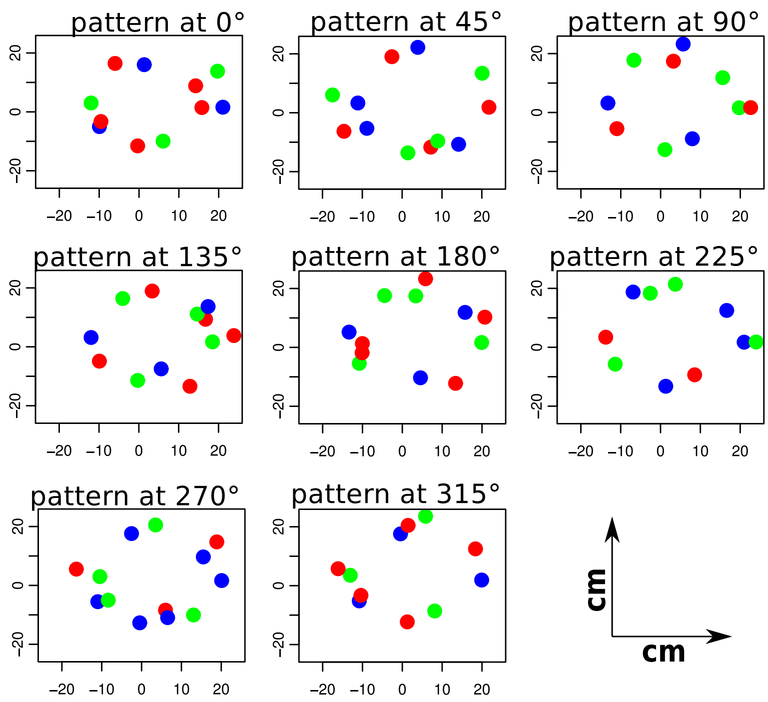


Figure 3.9: The colored blobs detected by a robot when surrounded by four neighbors. The perception of the central robot is shown for eight different pattern directions, with all the neighbors pointing in the same direction.

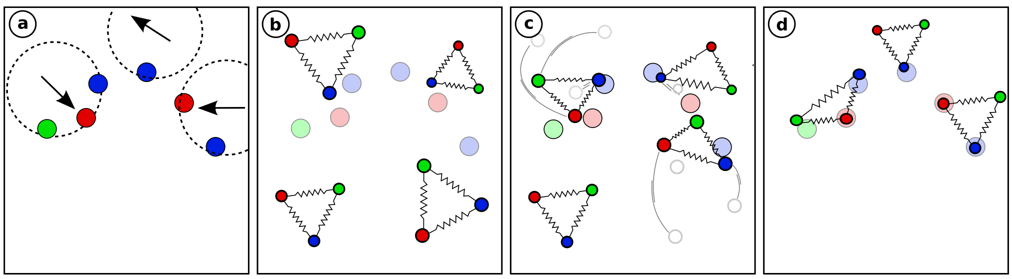


Figure 3.10: The pattern recognition algorithm. **(a)** The robot has perceived a list of colored blobs around itself. Dotted circles and arrow represent the physical robots in the picture. **(b)** Elastic triangular structures are created with random positions. **(c)** The simulation runs: the structures stretch and move, attracted by the blobs. **(d)** At the end of the run, the structures fit the blobs, and give estimates of the robot locations and orientations.

difficulty here is to recognize the triangular pattern of robots using only these blobs, at real time speed.

A classical approach consists of the evaluation of all the possible pattern configurations in order to select the one that maximizes a given criterion. This approach is not feasible because of the large number of possibilities. Instead, we devised an algorithm inspired by Kohonen maps (Kohonen, 1982, 2002). In a nutshell, the algorithm simulates a space in which we initialize a number of elastic triangles at random positions

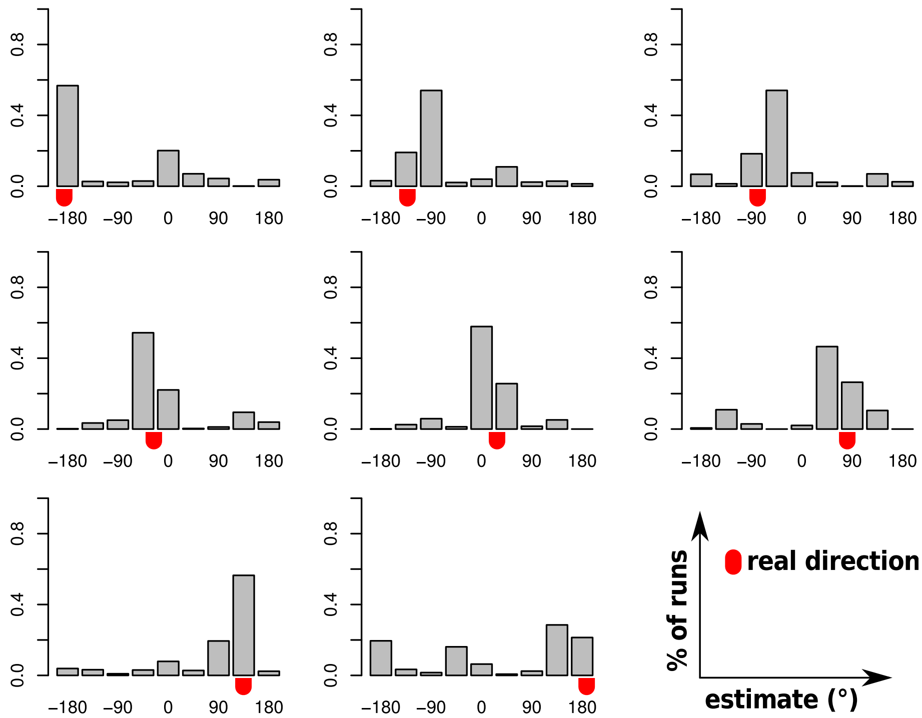
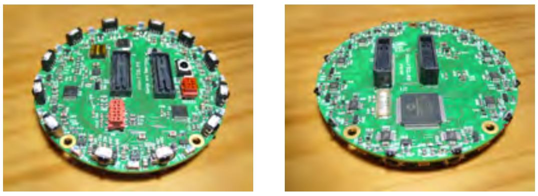
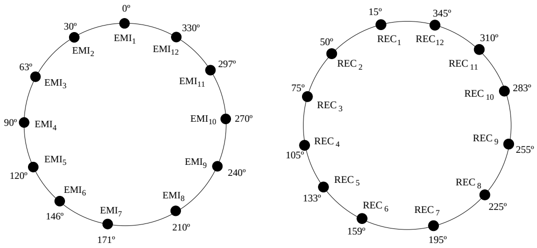


Figure 3.11: A central robot perceives the orientation advertised by four neighbors. The output of the pattern recognition software is reported for eight different directions (n=3500 runs for each plot), with a red mark indicating the real orientation advertised.



(a) (b)

Figure 3.12: E-puck range and bearing board **(a)** Top view **(b)** Bottom view.



(a) (b)

Figure 3.13: E-puck range and bearing board **(a)** emitters location **(b)** receivers location.

and orientations (see Figure 3.10). The summits of these triangles are colored. In addition, we introduce in the simulated space the color blobs which act as attractors for the summits of the triangles. Moreover, triangles can not superimpose and repel each other at close range. We simply let the simulation run for a short time and eliminate the triangles that do not stabilize their summits over blobs. The detected robots and their orientations are simply the remaining triangles. The simulation is simple and uses very limited computational resources, allowing to process up to four images per seconds on the s-bot.

Although a number of mistakes can happen in the pattern recognition process, the average performance of the system is fair. In Figure 3.11, we report the average direction detected by a robot surrounded by four other neighbors. We find that 75% of the runs produce an estimated orientation near the real orientation of the neighbors, in a range of $\pm 22.5°$.

3.2.2 Infrared range and bearing communication system

We have developed an extension board named the range and bearing communication system for the e-puck robot (see Figure 3.12 and Gutiérrez *et al.*, 2008; 2009a). The board allows the robots to communicate locally, obtaining at the same time both the range and the bearing of the emitter without the need for any centralized control or external reference. The communication system is therefore scalable and robots are fully autonomous. The range and bearing system allows robots to send and receive messages up to 6 m and in 12 different directions thanks to the 12 sensor/actuator pairs disposed around the body of the robot. Robots are capable of sending the same message in all directions, or specifically in one direction by using a single sensor/actuator pair. The power used to emit infrared signals can be adjusted in real time, which allows robots to decide at which maximum distance their messages can be perceived. We provide this board under an open hardware/software license which allows the robotics community to replicate, change and adapt the board to their needs (all material freely available at http://www.e-puck.org).

Hardware

The range and bearing board is controlled by its own processor, which is a dsPIC 33FJ256 produced by Microchip. In this way, the main processor of the e-puck does not lose computational resources to handle infrared communication. On the perimeter of the board there are 12 identical transmission modules. The modules are nearly equally distributed on the perimeter of the board so that the distance between them is of approximately $30°$ (see Figure 3.13). Each of these modules is composed of one infrared emitting diode, one infrared modulated receiver and one infrared photodiode. A Manchester code is implemented so that when a message is sent from a certain distance, it is received with the same intensity whatever the contents of the message. The implementation of the Manchester code allows a maximum data rate of 5 kbps. The range of communication can be software controlled from 0 cm to 80 cm.

The board has an isolated power supply dedicated to the transmission modules. This supply is built with a linear regulator which allows voltage variation. The power variation permits the modules to modify the emission range by changing the polarization of the emitting diodes. The original electronics on the board allows ranges of transmission of up to 6 m, but range has been limited to 80 cm in agreement with the robot size and the requirements of our experiments. Ranges from 0 cm to 80 cm are software controlled by the robot's main board via power supply adjustments.

Software

For any infrared message received, the board also calculates the range and the bearing of the emitter. To obtain the values of the range and bearing we start with the calculations of the bearing. A vector sum is implemented for the bearing calculations following Equation 3.1:

$$\widetilde{\phi} = arctan\left(\frac{\sum_{i=1}^{12} \widehat{\nu}_i cos\left(\beta_i\right)}{\sum_{i=1}^{12} \widehat{\nu}_i sin\left(\beta_i\right)}\right) \tag{3.1}$$

where $\widetilde{\phi}$ is the estimated bearing with respect to the robot's heading, β_i is the angular distance between sensor i and the board's heading and $\widehat{\nu}_i$ is the signal intensity perceived on sensor i.

Once the bearing is calculated we proceed to calculate the estimated distance to the source. The bearing calculated previously will determine if the emitter is facing perfectly a receiving sensor or if it is in between two sensors. In both cases, following the calculations of Pugh and Martinoli (2006), we obtain the correction of the received power value $\widetilde{\nu}$ as follows in Equation 3.2:

$$\widetilde{\nu} = \left(\left(\frac{\widehat{\nu}_l}{\sqrt[2]{\cos\theta_r}}\right)^4 + \left(\frac{\widehat{\nu}_r}{\sqrt[2]{\cos\theta_l}}\right)^4\right)^{\frac{1}{4}}, \tag{3.2}$$

where $\widehat{\nu}_l$ and $\widehat{\nu}_r$ are the values received on the left and right sensor from the estimated angle respectively and θ_r and θ_l are the angular distances between the estimated angle $\widetilde{\phi}$ and its left and right nearest sensor respectively.

Finally an interpolation for obtaining the distance in centimeters is obtained using a lookup table created empirically. We performed extensive communication tests with the board and found that the average bearing error across all angles and distances is 4.32 ° and 12.32 ° in the worst case. The average range error across all angles and distances is 2.39 cm and the worst case is 6.87 cm.

Validation

We validated the capabilities of the board with a simple but fundamental task of alignment. In this task, robots must adopt similar directions by exchanging information about their current orientation. The experimental setup consists of a group of homogeneous e-pucks that are positioned at a distance of 15 cm from each other, with randomly generated initial orientations, as depicted in Figure 3.14. Each agent can only change its orientation through rotational movements (*i.e.* they only turn on the spot).

Robots do not have any common global reference frame so they can only communicate the orientation as a relative measure to each other. In this case the common el-

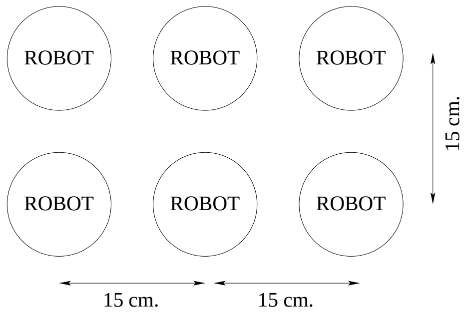


Figure 3.14: Physical arrangement of a group of 6 robots for the tests.

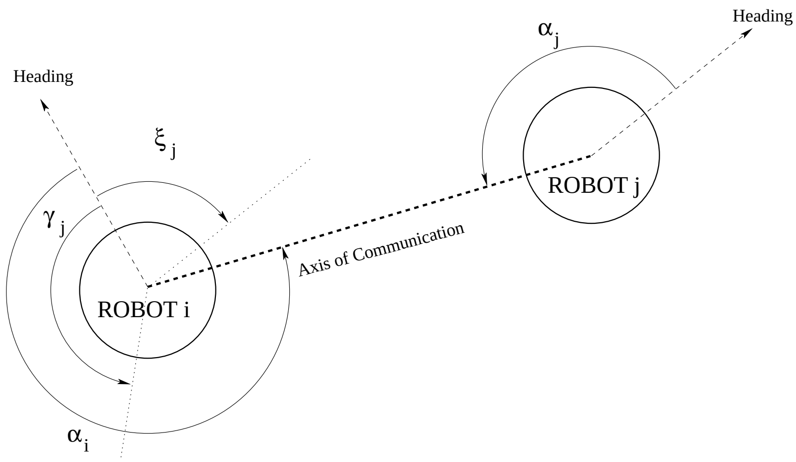


Figure 3.15: Robots sharing information about their relative orientations.

ement is the axis of communication (see Figure 3.15). In a first step, robot i transmits a broadcast frame. Subsequently, robot j understands that there is a neighbor at angle α_j. Robot j communicates its relative orientation α_j. In a second step, robot i transforms the received data ($\gamma_j = \alpha_j$) into its own coordinates system. It calculates the direction pointed by robot j as $\xi_j = \gamma_j + \alpha_i - \pi$. Following this communication, the robots both move gradually towards the same direction. The robots keep communicating regularly to achieve synchronization of their orientations.

To calculate the degree of alignment of all the robots, we use a specific measure of polarization. Polarization $P(G)$ of a group of robots G is defined using the angular nearest neighbor. For a robot r, the corresponding angular nearest neighbor c is defined such that θ^{rc}, the relative orientation of c with respect to r is the smallest possible : $\theta^{rc} < \theta^{ri}, \forall i \in G \backslash \{c\}$. We denote $\theta_{ann}(r)$ the relative orientation of the angular nearest neighbor of the robot r. The formal definition of polarization is as follows :

$$P(G) = \sum_{i \in G} \theta_{ann}(i). \tag{3.3}$$

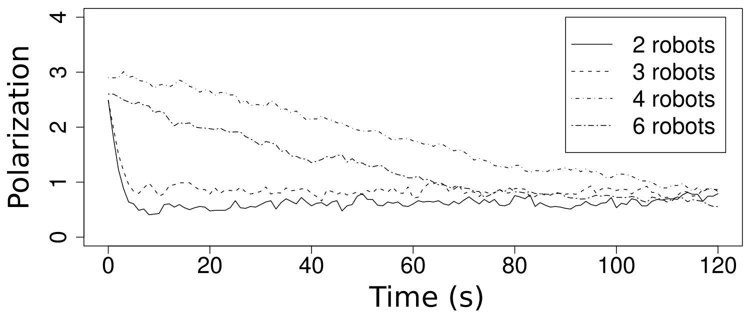


Figure 3.16: Mean polarization for 30 repeated experiments for all group sizes tested (2, 3, 4, and 6 e-pucks). Standard errors are not shown for the sake of clarity.

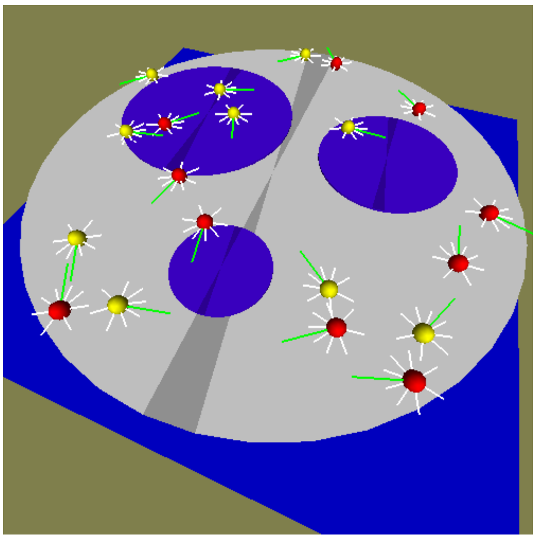


Figure 3.17: A snapshot of the simulator's graphical output.

If all robots are aligned, then $P(G) = 0$. Conversely, if headings are evenly distributed, $P(G) = 2\pi$. Lastly, if headings are random, *i.e.* drawn from a uniform distribution, then $P(G) = \pi$ in average. This measure makes possible comparisons among different group sizes since the average value of $P(G)$ is not affected by the number of robots in G.

We tested the algorithm with groups of 2, 3, 4, and 6 e-puck robots. In Figure 3.16, we report the average polarization of the robots across 30 repeated experiments. We observe that groups are successfully making use of the range and bearing communication system; they converge and maintain their alignment till the end of the experiments.

3.3 Simulator

Most of the experiments reported in this thesis have been first ran using a simulator. The use of simulations is a mandatory step when designing robotics experiment. The simulator allows to test ideas and make mistakes at low cost. On the opposite, real world robots, and especially our research prototypes, are prone to heavy failure when we make a mistake. Robots can wear and break fast if experiments are not carried out with special care. Robots are also time consuming: to make an experiment or test a controller, one has to upload a new program and change or recharge the batteries. Since we work with swarms of robots, this set-up can take ten minutes if we use 10 robots, and this has to be repeated after every experiment replication…In addition, a number of studies are only feasible in simulation, because they require too many robots, too much space, or too much time. We typically design our experiments in simulation. We then demonstrate the validity of our work with real robots, and finally we look for further results using additional simulations which are then backed up by the validation process.

Our simulation platform is a fast multi-robot simulator coded in C++ and forked from the *Twodeepuck* simulator, a simple and effective simulator implementing 2D kinematics, initially designed by Anders L. Christensen and Laurent Bury at the IRIDIA laboratory (Christensen, 2005; Bury, 2007). We have implemented a quadtree to speed up subroutines that handle the detection of collisions and the detection of local communication. Motion of objects is handled by a custom rigid body physics engine, specialized to simulate only the dynamics in environments containing flat terrain and walls. This restriction allows simplifications of the physics computations and thereby reduces the computational resources necessary for running simulations.

In order to accurately reproduce real world experiments, we have systematically sampled the data output of all the robot's sensors we used. For instance, with the infrared sensors we gathered the signal intensity perceived when the robot was presented another robot or a wall. To get a complete picture of the sensor's output, we tested an exhaustive set of distances and angles. With cameras, we recorded pictures and extracted the useful color information of objects with image processing tools. With this data at hand, we have been able to create effective models of the sensors output. The data fed to the controllers in simulation corresponds closely to what happens in reality.

The performance of our simulator varies in function of the complexity of the controllers and the amount of interactions to simulate. To give an approximate idea of the simulator speed, we used an Intel CPU (T7250 at 2 GHz) to make tests with 10 robots, running our most complex controller at hand, exchanging messages locally and avoiding obstacles. On average, an experiment of one hour is simulated in 1.27 seconds. With 100 robots, the same one hour experiment is simulated in 8.14 seconds on average. The

100 robots are simulated proportionally faster than the 10 robots because the amount of local interactions does not grow proportionally with the group size.

Chapter 4

Cooperative transport with negotiation of the pulling direction

In this chapter, we investigate collective decision making with a group of robots that have identical knowledge of the situation. More precisely, robots only have an approximate knowledge to make their decision, but all the robots share the same level of uncertainty and none of them is better informed than the others. The objective of this research is to demonstrate a mechanism that allows the robots to take advantage of redundant information present at the level of the group. Eventually, robots should collectively produce a better knowledge and take the appropriate collective decision.

This study is implemented with an experiment of cooperative transport in which the robots do not have precise knowledge of the delivery location. Initially, the robots receive a noisy information about which direction they should follow. The robots are then using local communication to exchange information and reach a consensus about the transport direction.

This experiment is performed with swarm-bots and relies on the visual range and bearing system. The information exchanged among robots bears only on opinions about the transport direction. At the core of the robots behavior, there is a simple instance of imitation behavior, directly inspired from the alignment behavior found in biological models of flocking and schooling. Each robot advertises its opinion with a system of LEDs, and uses the opinion advertised by the neighbors to update its own opinion. This opinion update takes place gradually, allowing in this way the convergence of all the opinions, and thereby leading to a collective choice about the transport direction.

This chapter is organized as follows. In Section 4.1.1 we present the task and the experimental setup. In Section 4.1.2 we present the negotiation mechanism we used to synthesize information inside the swarm of robots and reach a collective decision. In Section 4.1.2, we detail four different transport strategies that we tested with the robots. Section 4.2 present experiments conducted with real robots. There, we assess quantitatively the performance of the negotiation mechanism implemented with respect to different levels of noise and the different control strategies. Lastly, in Section 4.3 closes the chapter with discussion and conclusions.

4.1 Methods

4.1.1 The task and the experimental setup

The task is the cooperative transport of a heavy object towards a central place by a group of four real s-bots. The robots are physically connected to the object using their grippers. The central place is out of sight and the robots have no means to perceive it. The initial knowledge of each individual about the goal direction is provided with a given amount of noise.

The robots can share knowledge using visual communication in order to collectively improve their estimate of the goal direction and transport the object as fast and as accurately as possible towards the goal.

The mass of the object is 1.5 kilograms and it is chosen such that a single robot can not transport it. At least three robots are necessary to move the object. A high degree of coordination of the robots' motion is required to apply enough force to the object to move it. If the robots lack coordination, that is, if they pull in different directions, they can not move the object at all.

The experiments take place in an open space. Initially, four robots are connected to the object in a regular arrangement, thus forming a cross pattern as shown in Figure 4.1(c). We test four levels of noise on the robots' initial estimate of the goal direction: *no noise* (*0*), *low noise* (*L*), *medium noise* (*M*) and *high noise* (*H*). In the case of no noise, the initial direction of the robots is the same and points towards the central place.

The initial imprecise knowledge of the robots about the direction of the central place is modeled by a random number drawn from a von Mises distribution, which is the equivalent of the Gaussian in circular statistics (Jammalamadaka and SenGupta, 2001), and well suited for directional data. This distribution is characterized by two parameters μ and κ. The direction to the central place is indicated by μ, the mean of the distribution. The level of noise is indicated by κ. The smaller κ, the more the distribution resembles a uniform distribution in $[-\pi, \pi]$. When κ is large, the distribution resembles a Gaussian of mean μ and standard deviation σ, when $\kappa \to \infty$ the relationship $\sigma^2 = 1/\kappa$ holds. The three levels of noise L, M, H correspond to $\kappa = 3, 2, 1$, as displayed in Figure 4.2(a).

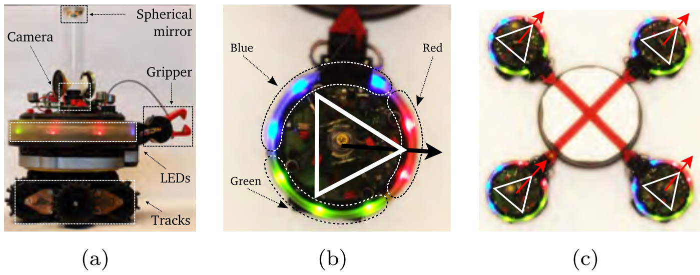
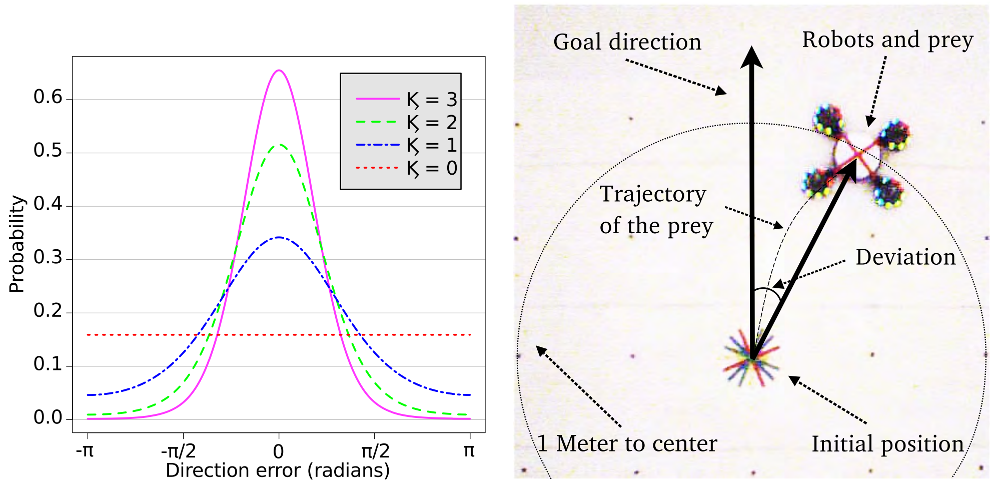


Figure 4.1: (a) The s-bot. **(b)** An s-bot displaying a direction using a triangular LED pattern. **(c)** Star-like formation of four s-bots around the object as used in the experiment.

After each trial, the robots are randomly permuted, so that the possible physical differences among the robots are averaged out and can be neglected in this study. We tested 4 possible goal directions of $0°, 22.5°, 45°$ and $67.5°$. Any direction above $90°$ is redundant as the pattern of connected robots (a cross) is symmetrical on the two perpendicular axes, and the robots are permuted at each trial. Finally, we have tested 4 possible strategies for the robots to transport the object towards the goal (see next section for more details). In total, we performed 256 replications: we tested 4 goal directions, 4 levels of noise and 4 distinct strategies for transport. Each combination of the aforementioned parameters was tested 4 times.

To extract the results, we used a camera placed above the initial position of the object to record videos of each trial. The experiment is stopped either when the object has been transported to a distance of 1 meter from its initial position or after 60 seconds (an average transport takes approximately 20 seconds). A trial can also be stopped if we judge that the robots are stuck in a situation that is potentially harmful to their hardware. Indeed, if the robots do not manage to coordinate their movements, they may pull in opposite directions and thus induce a high torque to their grippers. One gripper was broken during the experiments reported here, and we wished to avoid as much as possible any further damage. Any experiment stopped without the object being transported for more than 1 meter of distance from the initial position is considered as a transport failure.

For each trial, we have extracted the position of the object at each time step (5 pictures per seconds) using a simple tracking software. Using these data, we have categorized the trials in transport failure or success, and measured the duration of all trials. Furthermore, we measured the angular difference between the direction in which the object has been moved and the goal direction, as shown in Figure 4.2(b). Later on, we also use the term deviation to refer to this angular difference.



(a) (b)

Figure 4.2: (a) The effect of parameter κ on a von Mises distribution. **(b)** A snapshot describing the final situation of a successful transport. Note how the deviation of the transport direction from goal direction is measured.

4.1.2 Robot's controller

Negotiation mechanism

The negotiation mechanism is bio-inspired and implemented in a straightforward manner, following closely the rules used to model the orientation behavior of fish schools or bird flocks (Couzin *et al.*, 2002). Let n be the total number of robots. For each robot $i \in [1, n]$, let $\mathcal{N}_i(t)$ be the set of robots in the visual range of robot i at time t. This defines the topology of the communication network. Let $d_i(t) \in [-\pi, \pi]$ be the goal direction estimated by robot i at time t. Let $D^i_j(t) = d_j(t) + \epsilon^i_j(t)$ with $j \in \mathcal{N}_i(t)$ the direction of robot j perceived by robot i assuming noise $\epsilon^i_j(t)$.

If robot i communicates and exchanges information with its neighbors, it will calculate what we call a desired direction $\overline{d}_i$ by using Equation 4.1 that basically computes a mean direction. To do so, we use the sum of unit vectors, which is a classical method in circular statistics (Jammalamadaka and SenGupta, 2001):

$$\overline{d}_i(t) \quad = \quad \arctan\left(\frac{\sin\left(d_i\left(t\right)\right) + \sum_{j\in\mathcal{N}(i)}\left(\sin(D^i_j(t))\right)}{\cos\left(d_i\left(t\right)\right) + \sum_{j\in\mathcal{N}(i)}\left(\cos(D^i_j(t))\right)}\right). \tag{4.1}$$

The estimate of the goal direction of a particular robot is not updated directly. Indeed, the noise present in perception might induce oscillations if the update of the robots' estimates is done too fast. Therefore, we use a damping factor δ to stabilize the system (we chose $\delta = 0.05$ for our experiments). The update of the estimate of the goal direction for robot i is described by Equation 4.2:

$$d_i(t + \Delta t) \quad = \quad (1 - \delta) \cdot d_i(t) + \delta \cdot \overline{d}_i(t). \tag{4.2}$$

The motion control of each robot is implemented by a simple algorithm (Groß *et al.*, 2006b) that sets the speed of the robot's *treels* and the orientation of its turret to pull the object in the estimated direction d of the central place.

Control strategies

We have defined and implemented four distinct strategies. To refer to the strategies, we employ a notation in which **T** means transport, **N** means negotiation, and **:** marks the end of an optional and preliminary negotiation phase. If this preliminary phase takes place, it lasts 30 seconds. The second phase always involves transport and lasts 60 seconds.

- **Transport directly (T)**: a naive strategy that we use as a yardstick to show the improvement brought by the negotiation mechanism. The robots move along their initial direction. No communication and no update of the estimated direction is done.
- **Negotiate then transport (N:T)**: robots first negotiate their estimate of the direction of the goal for 30 seconds without moving. Afterwards, they all start moving without either communicating or updating their estimates.
- **Negotiate then transport and negotiate (N:NT)**: robots start by negotiating the direction of the goal for 30 seconds without moving. After this preliminary negotiation, they all start moving and at the same time they keep on negotiating.
- **Negotiate and transport (NT)**: from the very beginning of the experiment, the robots start both moving and negotiating.

At the beginning of the experiments, robots have each a rough estimate of the direction of the goal, but they never perceive directly the goal. The three last strategies may appear identical to the reader, but in fact two important aspects, namely time and noise in communication should be considered. On one hand, the duration of the negotiation process affects the degree of synchronization of the robots. On the other hand, visual communication is imperfect. When robots do not move, errors in visual communication are persistent and may have a strong impact on the outcome of the negotiation process.

When robots move, they modify slightly their relative locations and this results in a reduction of the errors in visual communication.

4.2 Results

We report here the experimental results of the task of cooperative transport for all the strategies and levels of noise tested. We examine three different aspects of the system: the ability of the system to succeed in transporting the object for a certain distance, the duration of transport and the accuracy in direction of transport.

4.2.1 Success in transporting

We first study the ability of the robots to transport the object. If the robots are not able to move the object over a distance of at least 1 meter from the initial position within 60 seconds, we consider the trial as a transport failure. Figure 4.3 presents the performances in transport of the four strategies for the different levels of noise.

First, we observe that in absence of noise (level *0*), the robots manage very well to transport the object without negotiating the direction. Therefore, negotiation is not necessary and it is desirable that strategies employing the negotiation mechanism do not perform worse. The strategy *N:T* yields only 75% of successful transports when there is no noise in the initial direction of the central place. When this strategy is employed, it is possible that negotiation is stopped while robots are not perfectly coordinated and no further correction can be done on the direction of the robots. The two other strategies *N:NT* and *NT* do not decrease the capability of the group of robots to transport the object with respect to strategy *T*. We have observed that during motion, the formation of robots can alter slightly, mainly due to slippage of the grippers on the object. Strategies using negotiation during transport allowed robots to quickly correct their direction and remain coordinated. Conversely, the strategies *T* and *N:T* were very sensitive to small errors.

When noise is present, the performance of the group of robots using strategy *T* decreases. For medium and high noise it is close to 10%. This result was expected as robots are not able to coordinate their motion at all and are initialized with different initial directions. We also notice that, although noise has a non negligible impact on the transport capability, the performances stay quite similar for different levels *L*, *M* and *H* of noise considering the strategies *N:T*, *N:NT* and *NT*. All strategies relying on the negotiation mechanism achieve better performances, and especially strategy *NT* is much less sensitive to noise than the others.

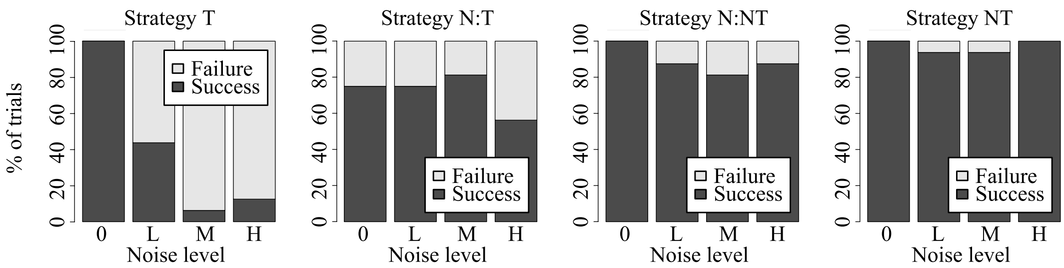


Figure 4.3: The percentage of successful and failed transports grouped by strategy and by level of noise.

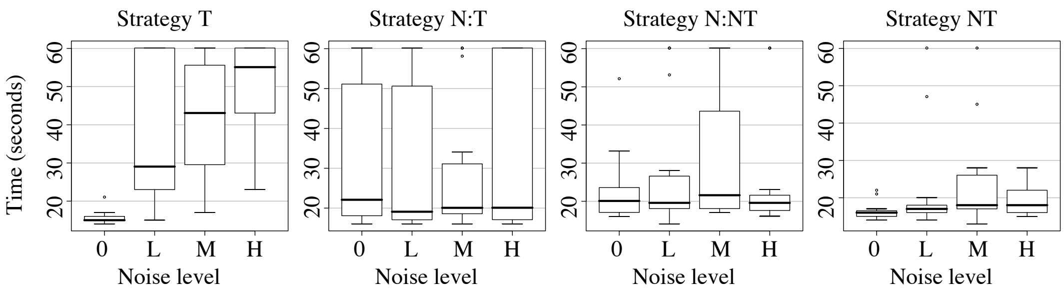


Figure 4.4: Box-and-whisker plot (Chambers *et al.*, 1983) showing the duration of transport of the object (in seconds), taking into account successful and failed transports. The distributions are grouped by strategy and by level of noise.

4.2.2 Duration of transport

We focus now on the duration of the transport. For all the trials, we consider whether or not transport is successful. Figure 4.4 shows for all strategies and all levels of noise boxplots of the duration of the 16 transport tasks. Note that we do not take into account the preliminary negotiation period that lasts 30 seconds when strategies *N:T* or *N:NT* are employed.

Once again, the performance of strategy *T* in absence of noise is the best with respect to any other pair of strategy and level of noise. Only strategy *NT* reaches a comparable performance.

When the level of noise increases, the duration of transport of the strategy *T* increases too, in a quasi linear manner. Strategies that rely on the negotiation mechanism are much less sensitive to noise. The duration of transport using those strategies is very similar for the different levels of noise *L*, *M* and *H*, but strategy *N:T* has produced more failures. Because robots can not correct their coordination with this strategy, they easily rotate while transporting the object. This constant error produces round or even circular trajectories and prevents the robots to quickly move the object away from its initial position. Strategies can be clearly ranked: the slowest (*N:T*), the average (*N:NT*) and the fastest (*NT*).

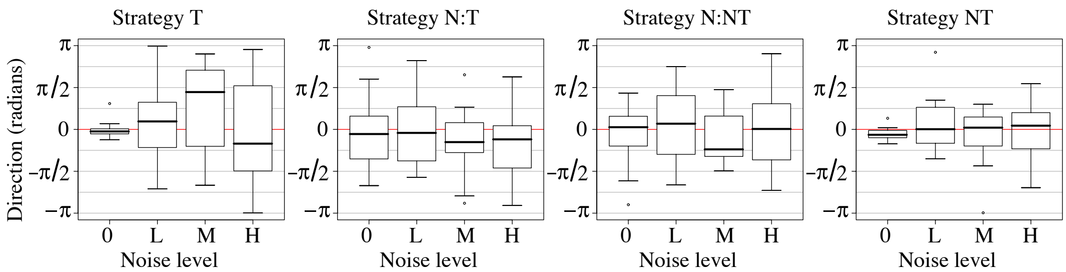


Figure 4.5: Box-and-whisker plot (Chambers *et al.*, 1983) showing the average deviation (in radians) between direction of transport and direction of the central place. Both successful and failed transports are taken into account. The distributions are grouped by strategy and by level of noise.

4.2.3 Deviation from the direction of the central place

The last measure we study is the deviation of the direction of transport with respect to the direction of the central place. Again, we take into account all trials. The study of deviation from the direction of the central place confirms all previous observations (see Figure 4.5).

In absence of noise, the naive strategy *T* performs very well, and the only other strategy with a comparable result is strategy *NT*. When noise is introduced, the performance of strategy *T* decreases. The strategies that make use of negotiation perform better, and show only small differences for the different levels of noise tested. Among these strategies, the best is *NT*.

With strategy *NT*, the median transport direction is centered on the goal direction and the experimental deviation is limited to $\pm 61°$ in 80% of the trials when there is a high level of noise on the initial knwledge given to the robots. This result is obtained with robots that can advertise at most 8 different directions only.

We compared the initial knowledge of the robots to the average deviation from the goal direction after transport using the strategy *NT*. To do so, we fitted von Mises distributions to the deviations measured. Then, we estimated the κ parameter of these distributions which corresponds to the error of the transport direction. The robots start with an initial knowledge affected by a noise that corresponds to an individual error of respectively $33.1°$, $40.5°$ and $57.3°$. After the application of the strategy *NT*, the error of transport direction for the levels of noise *L*, *M*, *H* are respectively $42.8° \pm 76.2°$, $42.3° \pm 75.5°$ and $42.5° \pm 75.7°$ (degrees $\pm$ standard error). These values are not significantly different. Moreover, it is observed that for the level of noise *H*, the strategy *NT* improves the robots' estimate of the direction of the central place.

4.3 Discussion and Conclusions

In this chapter, we provided a simple negotiation mechanism that allows robots to share knowledge and reach consensus on a transport direction. The comparison of the transport strategies has shown that negotiation during transport of an object improves the coordination of motion. It has been observed that the mechanism is neutral: if negotiation is not mandatory to achieve efficient transport, making use of negotiation does not alter the transport performances with respect to naive transport with no negotiation. Hence, it is not necessary to choose which strategy to employ depending on the level of noise affecting robots' knowledge of the direction of the central place, negotiation can be used at any time.

Besides the coordination of motion, our experimental results have also shown that the group of robots could improve their knowledge of the direction of the central place by means of negotiation. The improvement of the accuracy of the transport direction with respect to the robots' initial knowledge is most striking when the level of noise is high.

Negotiation of transport direction contains a collective decision mechanism that could be adapted to a large number of situations. The core mechanism that underpins negotiation can be seen as a synthesizing function of the robots knowledge. By exchanging locally their opinions and imitating each other, robots converge to a global average opinion. In this way, not only do robots reach a consensus, but they also improve their individual knowledge of the situation.

In the following chapter, we perform a localization experiment in which robots must share knowledge to better navigate to foraging areas, and at the same time decide which areas to exploit. In the experiment, some robots have better information than others. In that case, the negotiation mechanism presented here is not able to take advantage of the extra information. The simple modification that we will investigate consists in giving more weight to the opinion of the better informed individuals.

Chapter 5

Collective localization with social odometry

In this chapter we examine collective decision making when some individuals are better informed than others. We have previously shown that our negotiation mechanism, which consists in averaging local opinions, works well with individuals that have statistically identical knowledge. Here, the average function would naturally ruin the privileged information available inside the group. In the following experiments, informed individuals are not fixed, rather every individual can acquire knowledge at any moment, but this knowledge degrades with time. Therefore we want the robots to take advantage of these individual differences and exploit information of better quality when possible.

To perform this study, we have devised a central place foraging experiment that requires collective localization. Robots must repetitively navigate between resources and a central place. However, robots are unable to locate precisely neither the resources, nor the central place. In fact, to estimate the location of these different areas, robots rely on odometry. Odometry is a localization technique that consists in measuring the number of wheel revolutions done so far to estimate the distance traveled and the direction of motion. The knowledge obtained by means of odometry is imprecise because any error due to slippage on the ground is accumulated and amplified as time and distance traveled increase. Therefore, the knowledge of the robots degrades as they move away from a known location. To cope with this degradation, robots may encounter conspecifics that have better localization knowledge and exchange information with them in order to improve their estimates.

This experiment is performed with e-puck robots equipped with the range and bearing communication system that we have designed. Two robots exchange information in a peer to peer manner when they come across each other. The information transmitted by each robot is its estimate of the location of the last place visited and a measure of the quality of that information. Robots update their own localization estimates with the communicated estimates, taking into account the confidence levels related to each estimate.

We initially used this information processing method to improve localization in presence of a single resource and we further tested the same mechanism with two resources. We found that robots propagate information of better quality through the swarm, and as a side effect they also favor situations in which information quality is higher, that is, they prefer to navigate along the shortest path to the closest resource because localization estimates are more accurate on shorter distances.

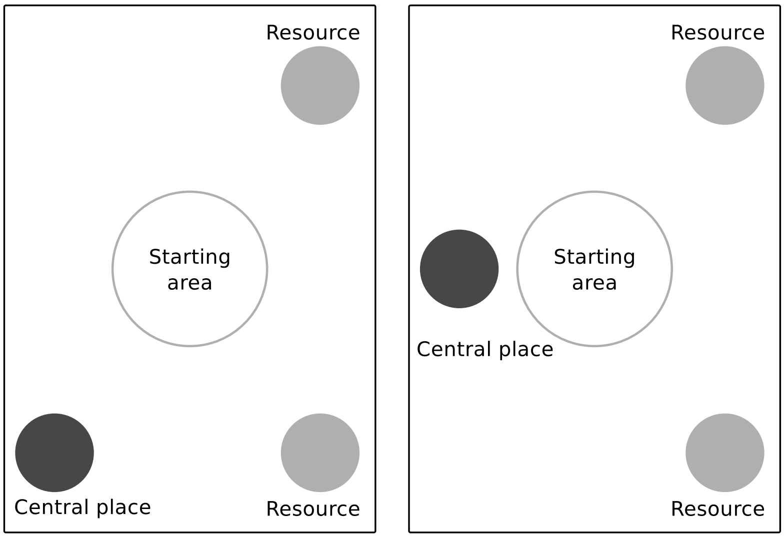
This chapter is organized as follows. After detailing our experiment in Section 5.1.1, we present the controller designed to perform collective localization using social odometry in Section 5.1.2. We describe the communication method, as well as the mechanism used to fuse information based on confidence levels. In Section 5.2 we present the results of experiments conducted in simulation and with real robots that demonstrate the ability of the swarm to make a collective decision. Lastly, in Section 5.3, we discuss the contribution of this chapter.

5.1 Methods

5.1.1 The task and the experimental setup

The foraging task presented in this chapter is carried out in a rectangular bounded arena that contains a central place and one or two resources. The central place and the resources are represented by circular areas on the ground. Robots have to find a resource and virtually transport items back to the central place. Robots have no knowledge about the dimensions of the arena or the areas locations. They are initially located in the middle of the arena with random position and orientation, where they can perceive neither the central place nor a resource. Once the experiment starts, robots search for the central place and a resource. When both areas have been located, they try to forage from one area to the other endlessly. Note that the random initialization outside the central place has been chosen because of two main reasons: (i) the physical robots could not be all initialized at the very same time in the central place, given the small central place radius and (ii) we wanted the robots to start their foraging without any knowledge about the areas in the arena. Therefore, although based on a foraging strategy, the experiments presented could be easily extended to, for example, a collective cleaning or collective rescue experiment.

Two different experimental setups (ESs) have been chosen to compare and study the convergence of the algorithms. Figure 5.1a shows an *Asymmetric ES* (AES), where the two resource are at different distances from the central place. Figure 5.1b shows a



(a) (b)

Figure 5.1: **(a)** Asymmetric experimental setup: one resource is closer to the central place than the other. **(b)** Symmetric experimental setup: the two resource are at the same distance from the central place.

Symmetric ES (SES), where the two resource are at the same distance from the central place. Depending on the number of robots involved in our experiments, we have used arenas of different sizes. In total, we have used three different arenas whose dimensions and number of robots vary as shown in Table 5.1. Moreover, all our experiments are repeated 30 times so as to allow statistical analysis of the results.

For the experiments we have used e-puck robots. We have equipped each robot with a range and bearing local communication board. To perform the odometry movement calculations, robots rely on their internal clock. Moreover, the robots make use of the infrared sensors to detect the presence of obstacles or any neighbors with whom the robots can communicate. Lastly, robots rely on ground sensors to differentiate the different foraging areas in the environment. To this end, we have colored the central place in black, the resources in grey, and left the rest of the environment in white.

5.1.2 Robots' controller

Robots are controlled by a finite state machine depicted in Figure 5.2. In the following we summarize the possible behavioral states. The controller is initialized in the *Explore* state.

- **Explore**: the robot carries out a random walk (*i.e.* moves straight and randomly changes direction; see Reif, 1965) with obstacle avoidance. If the robot encounters another robot, it switches to the *Communicate* state. If a central place or resource

Table 5.1: Parameters describing the three arenas used in the experiments.

Parameter	Arena 10	Arena 50	Arena 100
Number of robots	10	50	100
Number of resources	2	2	2
Arena dimensions (m^2)	1.2 x 1.7	3.0 x 4.25	6.0 x 8.5
Experiment duration (s)	1800	7200	7200
Starting area radius (m)	0.2	0.5	0.5
Goal areas radius (m)	0.2	0.1	0.1
AES: central place location (x,y) (m)	(0.45,-0.75)	(1.1,-1.7)	(2.2,-3.4)
AES: resource 1 location (x,y) (m)	(-0.45,0.75)	(-1.1,1.7)	(-2.2,3.4)
AES: resource 2 location (x,y) (m)	(-0.45,-0.75)	(-1.1,-1.7)	(-2.2,-3.4)
SES: central place location (x,y) (m)	(-0.45,0.0)	(-1.1,0.0)	(-2.2,0.0)
SES: resource 1 location (x,y) (m)	(0.45,0.75)	(1.1,1.7)	(2.2,3.4)
SES: resource 2 location (x,y) (m)	(0.45,-0.75)	(1.1,-1.7)	(2.2,-3.4)

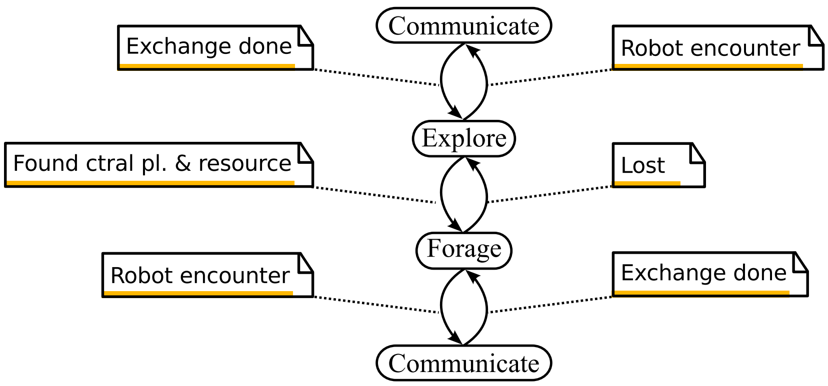


Figure 5.2: The robots' controller. Three different states are present in the finite state machine. In this representation, we duplicated the *Communication* state to make clear that robots resume their previous activity after a communication. Arcs denote transitions between the states.

area is found, the robot stores its position in its goal locations memory. Once the robot has discovered the central place and at least one resource, it switches to the *Forage* state.

- **Forage**: the robot has information about both the central place and the resource locations. It moves from the central place to a resource and back following the shortest path, and avoiding any obstacle on the way. If the robot arrives at one of the two areas, detected by the ground sensors, it stores the new estimated position and goes towards the other area. If the robot arrives at a place where the area was supposed to be but is not, it resets its memory of goal locations. As a consequence, the robot switches to the *Explore* state. If the robot encounters another one, it switches to the *Communicate* state.

- **Communicate:** The robot receives positional information from the encountered robot. The robot also sends its information about the location of the last detected area to the encountered robot. Robots use relative coordinates and locally establish a common reference frame to exchange information (see the next subsection). After these operations, the robot switches back to its previous state.

The robots are initially located at random positions in the middle of the arena. There, robots perceive neither the central place nor the resources. Once a robot finds the central place or a resource, it stores its position and continues with a random walk until it finds the other area. When both areas have been located, the robots try to go from one area to the other endlessly. Because of the movement integration errors, robots might arrive at some coordinates where they estimate the area should be located but it is not. If this happens, a robot considers itself lost, resets its estimated location and starts a random walk until it finds both areas again. On the other hand, if a robot correctly arrives at one of the areas, it updates the area position coordinates.

Establishing a common reference frame to communicate locations

When two robots meet, they exchange their estimates about the locations of the areas. However, robots do not share a global coordinates system, so they rely on their communication axis to transform the information transmitted by their neighbor into their own frame. This information can be locally transmitted thanks to the range and bearing board. Figure 5.3 shows an example of how information about the estimated location of area A, previously visited by robot i, is transmitted from robot i to robot j. In a first step, robot i transmits its estimate of the distance dy^i and direction ϕ^i of area A to robot j. For the direction, the value transmitted is the angle α, obtained from ϕ^i using the communication beam as reference axis: $\alpha = \phi^i - \gamma^i$, where γ^i is the bearing provided by the range and bearing board. In a second step, robot j transforms the received data

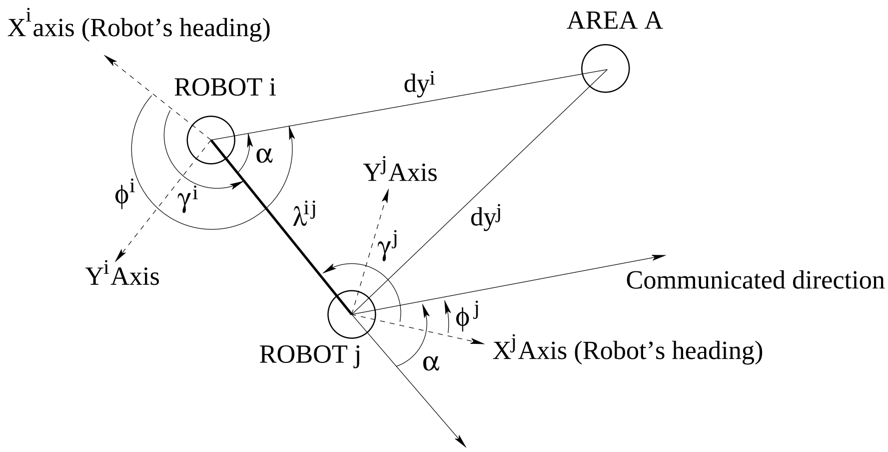


Figure 5.3: Robots sharing information about the estimated location of area A. Robot i has previously visited area A and communicates its estimates (dy^i and ϕ^i) to robot j.

into its own coordinates system. First, it calculates the direction provided by robot i as $\phi^j = \gamma^j + \alpha - \pi$, followed by the calculation of the location $(\widetilde{x}^j, \widetilde{y}^j)$ of area A:

$$\begin{aligned} \widetilde{x}^j &= \lambda^{ij} cos\left(\gamma^j\right) + dy^i cos\left(\phi^j\right) \\ \widetilde{y}^j &= \lambda^{ij} sin\left(\gamma^j\right) + dy^i sin\left(\phi^j\right) \end{aligned} \tag{5.1}$$

where λ^{ij} is the range provided by the range and bearing board.

The social generalized induced fermi filter

The Social Generalized Induced Fermi Filter (SGIFF) is a filter inspired by the Kalman Filter (KF) equations induced by the spectral norm of the error covariance matrix. Its main idea is to obtain a scalar value which represents the uncertainty a robot has about its estimates (confidence level). The SGIFF allows the robots to evaluate the information they have with respect to the information provided by their neighbors by means of a confidence level parameter proportional to the distance traveled. Based on this confidence level, the robots are able to update their estimate state vector with the information provided by other robots without the use of any movement error model. Therefore, depending on the confidence level ratio, the robots will adopt or ignore the information offered by their neighbors.

The SGIFF can be explained as follows. Let the state vector of the robot i at time k be

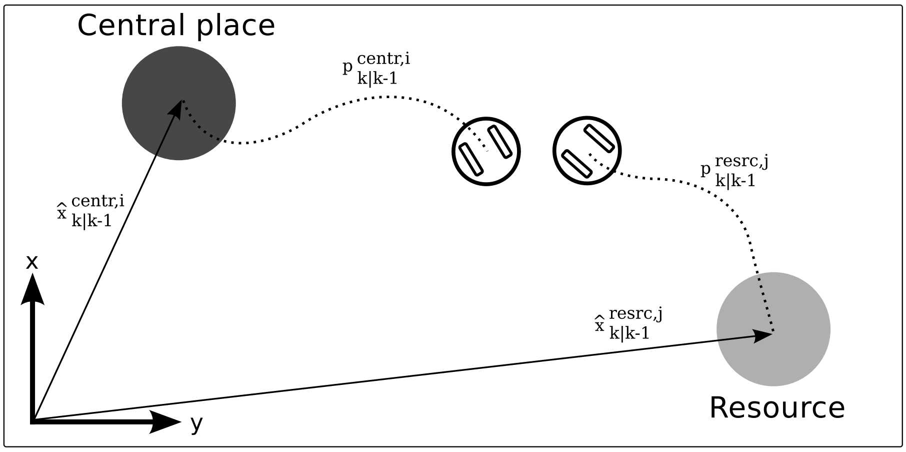


Figure 5.4: Robots information about the central place and resource areas and the inverse of their confidence levels. $\hat{\boldsymbol{x}}^{centr,i}_{k|k-1}$ represents the *a priori* estimated central place position of robot i at time k and $\hat{\boldsymbol{x}}^{resrc,j}_{k|k-1}$ represents the *a priori* estimated resource position of robot j at time k. The robots keep track of the distance traveled as the inverse of the confidence levels ($p^{centr,i}_{k|k-1}$ and $p^{resrc,j}_{k|k-1}$).

$$\boldsymbol{x}^i_k = \begin{bmatrix} x^i_k & y^i_k & \theta^i_k \end{bmatrix}^T \tag{5.2}$$

where x^i_k and y^i_k are the robot's Cartesian coordinates and θ^i_k its orientation.

Robots use dead-reckoning to estimate and reach the central place and resource locations. When robot i finds the central place or the resource, it stores its *a priori* estimated location information (its actual position and orientation) as $\hat{\boldsymbol{x}}^{centr,i}_{k|k-1}$ and $\hat{\boldsymbol{x}}^{resrc,i}_{k|k-1}$ respectively. Additionally, the robot keeps track of the distance traveled since it left the central place or the resource denoted by $p^{centr,i}_{k|k-1}$ and $p^{resrc,i}_{k|k-1}$ respectively, which represents the inverse of the *a priori* confidence level the robot has about its estimated information (see Figure 5.4). While the robot is moving, the uncertainty about its location grows (Smith and Cheeseman, 1987; Feng *et al.*, 1994). Therefore, its confidence level decreases.

At each time step, robot i checks whether there is another robot to communicate with. If there is not, it updates its *a posteriori* estimated goal locations and the inverse of the confidence levels as:

$$\begin{aligned}
\widehat{\boldsymbol{x}}_{k|k}^{centr,i} &= \widehat{\boldsymbol{x}}_{k|k-1}^{centr,i} \\
p_{k|k}^{centr,i} &= p_{k|k-1}^{centr,i} \\
\widehat{\boldsymbol{x}}_{k|k}^{resrc,i} &= \widehat{\boldsymbol{x}}_{k|k-1}^{resrc,i} \\
p_{k|k}^{resrc,i} &= p_{k|k-1}^{resrc,i}
\end{aligned} \tag{5.3}$$

In the next step ($k + 1$), the *a priori* estimated goal locations are updated with the robot's movement in the time step duration ($\Delta\widehat{\boldsymbol{x}}_{k+1}^{i}$), and the inverse of the *a priori* confidence levels are updated with the distance traveled (Δd_{k+1}^{i}) in the time step duration. Therefore, $\Delta d_{k+1}^{i} = \xi_{k+1}^{i} \cdot \rho/2$, where ρ is the distance between the wheels (53 mm for the e-puck) and ξ_{k+1}^{i} is the angular displacement made in the time step duration:

$$\begin{aligned}
\widehat{\boldsymbol{x}}_{k+1|k}^{centr,i} &= \widehat{\boldsymbol{x}}_{k|k}^{centr,i} + \Delta\widehat{\boldsymbol{x}}_{k+1}^{i} \\
p_{k+1|k}^{centr,i} &= p_{k|k}^{centr,i} + \Delta d_{k+1}^{i} \\
\widehat{\boldsymbol{x}}_{k+1|k}^{resrc,i} &= \widehat{\boldsymbol{x}}_{k|k}^{resrc,i} + \Delta\widehat{\boldsymbol{x}}_{k+1}^{i} \\
p_{k+1|k}^{resrc,i} &= p_{k|k}^{resrc,i} + \Delta d_{k+1}^{i}
\end{aligned} \tag{5.4}$$

Therefore, if there is no encounter between the robots, the confidence level continues to decrease until the robot arrives at the central place or a resource or until it gets lost.

If two robots meet, they communicate and update their estimates. In what follows we show all the different goal location exchange options, where goal represents either the central place or a resource:

- **None of the two robots know the goal locations:** Robots do not exchange any information.
- **Only one robot knows a goal location:** Let us assume that robot i is the one which has previously visited the goal. Its *a priori* estimated location is $\widehat{\boldsymbol{x}}_{k|k-1}^{goal,i}$ and the inverse of its corresponding *a priori* confidence level is $p_{k|k-1}^{goal,i}$. Robot j replaces its information with the one provided by robot i:

$$\begin{aligned}
\widehat{\boldsymbol{x}}_{k|k}^{goal,j} &= \widehat{\boldsymbol{x}}_{k|k-1}^{goal,i} \\
p_{k|k}^{goal,j} &= p_{k|k-1}^{goal,i}
\end{aligned} \tag{5.5}$$

- **Both robots know the goal location:** In this case, the two robots exchange data and update their information using the SGIFF (explicited with the Equations 5.9 and 5.10 reported below):

$$\begin{aligned} \widehat{\boldsymbol{x}}_{k|k}^{goal,i} &= SGIFF\left(\widehat{\boldsymbol{x}}_{k|k-1}^{goal,i}, \widehat{\boldsymbol{x}}_{k|k-1}^{goal,j}, p_{k|k-1}^{goal,i}, p_{k|k-1}^{goal,j}\right) \\ p_{k|k}^{goal,i} &= SGIFF\left(p_{k|k-1}^{goal,i}, p_{k|k-1}^{goal,j}\right) \end{aligned} \tag{5.6}$$

and

$$\begin{aligned} \widehat{\boldsymbol{x}}_{k|k}^{goal,j} &= SGIFF\left(\widehat{\boldsymbol{x}}_{k|k-1}^{goal,j}, \widehat{\boldsymbol{x}}_{k|k-1}^{goal,i}, p_{k|k-1}^{goal,j}, p_{k|k-1}^{goal,i}\right) \\ p_{k|k}^{goal,j} &= SGIFF\left(p_{k|k-1}^{goal,j}, p_{k|k-1}^{goal,i}\right) \end{aligned} \tag{5.7}$$

Note that $p^{goal,i}$ values are not initialized until the robot finds each area or a neighbor reports about its location. Once the robot has found one goal area, $p^{goal,i}$ is set to 0. On the other hand, a robot, to which a neighbor has communicated the area position, updates $p^{goal,i}$ with the value offered by its neighbor according to the SGIFF equations.

In order to produce an *a posteriori* guess location, each robot takes into account all information available, but weighs its sources in a different way. To calculate $\widehat{\boldsymbol{x}}_{k|k}^{goal}$, we adopt the so called pairwise comparison rule (Santos *et al.*, 2006; Traulsen *et al.*, 2006, 2007) often adopted in evolutionary/social dynamics studies, to code the social learning dynamics, which makes use of the Fermi distribution (Reif, 1965; see also Figure 5.5):

$$g_k^{goal,i} = \frac{1}{1 + e^{-\beta\left(\Delta p_{k|k-1}^{goal,ij}\right)}} \tag{5.8}$$

where $\Delta p_{k|k-1}^{goal,ij} = p_{k|k-1}^{goal,i} - p_{k|k-1}^{goal,j}$ and β measures the importance of the relative confidence levels in the decision making. For low values of β, the decision making proceeds by averaging the confidence levels whereas for high values of β, we obtain a pure imitation dynamics commonly used in cultural evolution (Hammerstein, 2003) defined by a sharp step function. In the first case, the confidence level works as a small perturbation to a simple average between the two estimates, while in the latter, each robot is ready to completely ignore the estimate which has a smaller relative confidence level.

Using the Fermi function, we use a weighted average to obtain the new location $\widehat{\boldsymbol{x}}_{k|k}^{goal,i}$ and inverse confidence level $p_{k|k}^{goal,i}$. Hence, the SGIFF is defined by the following equations:

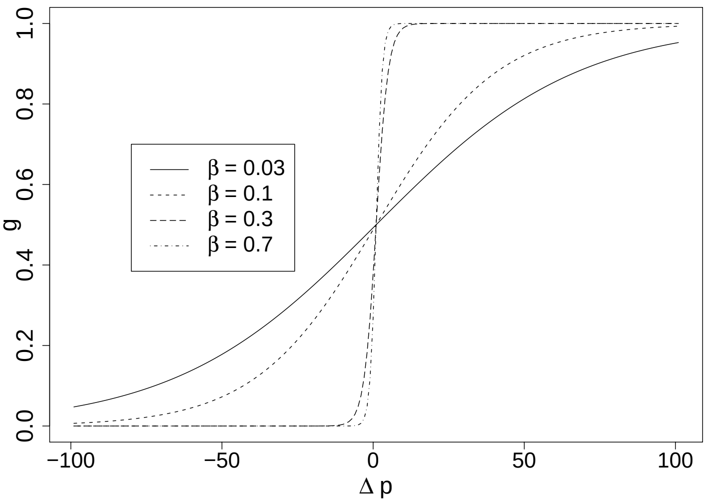


Figure 5.5: The Fermi function that allows robots to decide between their own estimate and the information provided by the others.

$$\widehat{\boldsymbol{x}}_{k|k}^{goal,i} = \left(1 - g_k^{goal,i}\right) \widehat{\boldsymbol{x}}_{k|k-1}^{goal,i} + g_k^{goal,i} \left(\widehat{\boldsymbol{x}}_{k|k-1}^{goal,j} + \boldsymbol{x}_k^{ij}\right) \tag{5.9}$$

$$p_{k|k}^{goal,i} = \left(1 - g_k^{goal,i}\right) p_{k|k-1}^{goal,i} + g_k^{goal,i} p_{k|k-1}^{goal,j} \tag{5.10}$$

Therefore, we observe that the SGIFF fuses the robot estimations based on their confidence levels. A better confidence level implies its associated measure will have a stronger weight on the filter performance. As we will see in Section 5.2, this simple mechanism triggers a collective decision in favor of the closer goal area. This is because a closer goal gives shorter paths which have less uncertainty, and therefore they are chosen by the individuals. Typically, when $\Delta p_{k|k-1}^{goal} \sim 0$ robots average their estimated information about the location of the same goal area. On the other hand, there is a high probability that robots communicate information about different goal areas when $\Delta p_{k|k-1}^{goal}$ is in one of its ends. In that case, a robot adopts ($\Delta p_{k|k-1}^{goal} \to \infty$) or ignores ($\Delta p_{k|k-1}^{goal} \to -\infty$) its neighbor estimates. When the two resources are located at different distances from the central place, robots coming from the nearest one (*resource A*) will have a better confidence level than those coming from the farthest one (*resource B*). The difference between confidence levels allows the robots coming from *resource B* to update their estimated location of *resource B* with that of *resource A*, favoring in this way the exploitation of the closest resource.

5.2 Results

In the following, we report results of simulated and real experiments and analyze the collective behavior of the robots.

5.2.1 Experiments in simulation

In our simulations, IR proximity sensors have a range of 5 cm, while the range and bearing module used for the communication has a range of 15 cm. Furthermore, we added uniformly distributed noise to the samples in order to simulate effectively the different sensors. $\pm 20\,\%$ noise is added to the infrared sensors and $\pm 30\,\%$ to the ground sensors. In the range and bearing sensor, noise is added to the range (± 2.5 cm) and bearing ($\pm 20\,°$) values. Moreover, each message emitted can be lost with a probability that varies linearly from 1 % when the sender-receiver distance is less than 1 cm, to 50 % when the two robots are 15 cm from each other. A differential drive system made up of two wheels is fixed to the body of the simulated robot. Errors have also been introduced into the encoder sensors chosen uniformly random in $\pm 20\%$ of the maximum movement at each time step for each wheel. The robot speed has been limited to 6 cm/s when moving straight and 3 cm/s when turning. These values are also used with the real robots so as to allow the comparison of simulated and real experiments.

General performance

The overall behavior of our controllers is a function of the parameter β. As explained in Section 5.1.2, the value of this parameter determines the rate at which the robots adopt or ignore their neighbor's information. When β is small, robots average their personal information with the neighbor's information. When β is large, robots adopt neighbor's information if they have a lower confidence than the neighbor in their estimates. To assess the general impact of this value we have conducted a parameter study. We examined eight values defined by $10 \cdot {}^{x}$, with $x \in \{-5, -4, ..., 2\}$. The general performance was evaluated using a retrieval task, that is, a task in which robots have to transport items from a resource to the central place. Each time a robot completes a run from the central place to the resource and comes back to the nest, we consider the robot has succeeded in its task and we count one more round trip. Figure 5.6 shows the number of round trips made by all the robots in the experiment for the two ESs in the three different arenas for each β value.

The figures show a similar shape for the asymmetric and symmetric experimental setup and the different arenas. For $\beta \in \{10^{-5}, 10^{-4}, 10^{-3}\}$ robots perform poorly in the retrieval process. This is because the robots simply average the available information.

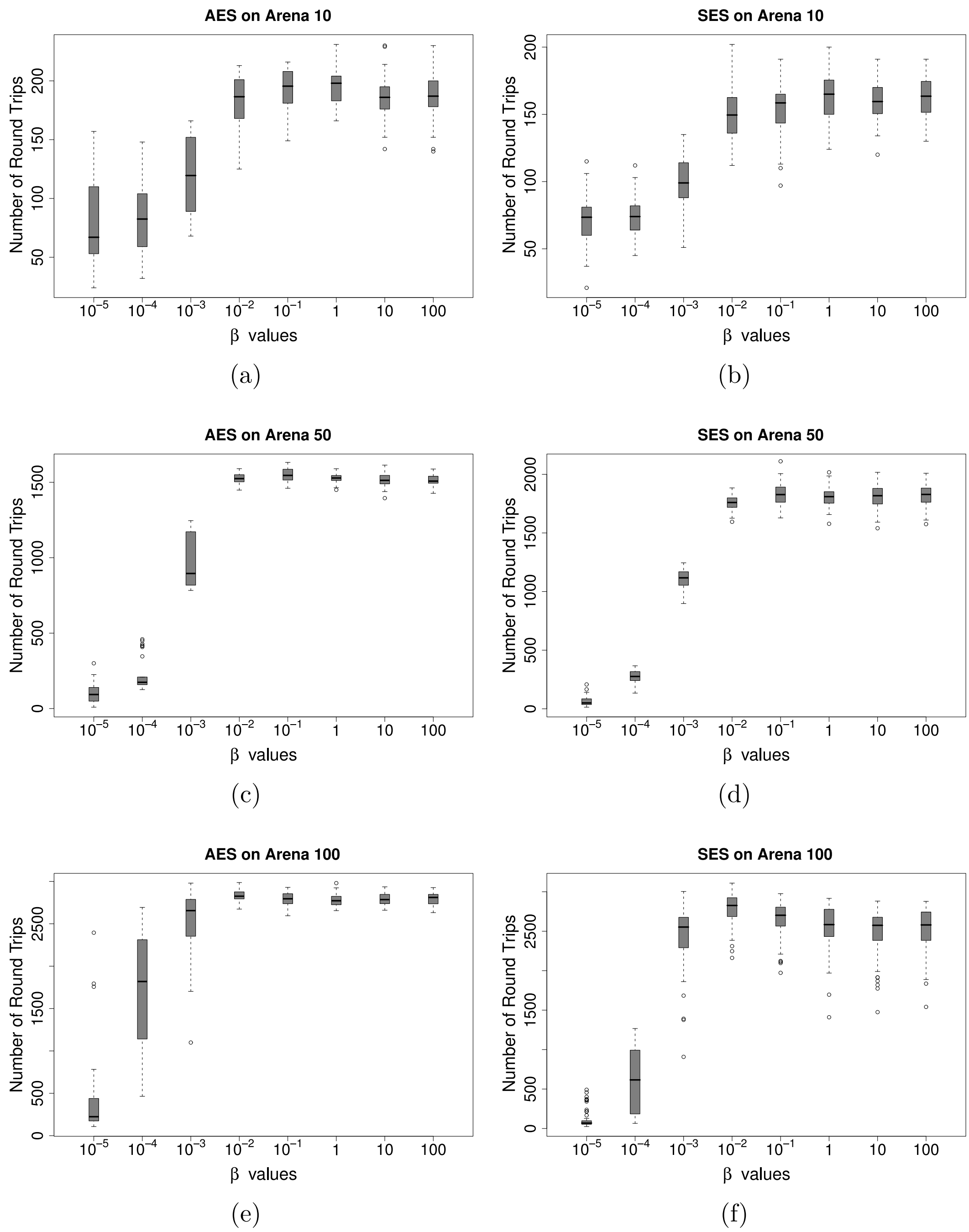
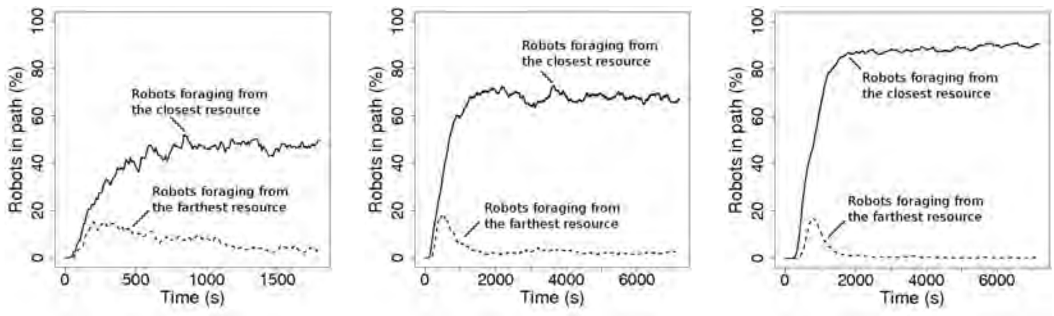


Figure 5.6: Retrieval results for the different ESs and arenas tested (30 replications for each boxplot). Each box comprises observations ranging from the first to the third quartile. The median is indicated by a horizontal bar, dividing the box into the upper and lower part. The whiskers extend to the farthest data points that are within 1.5 times the interquartile range. Outliers are shown as circles.



(a) (b) (c)

Figure 5.7: Percentage of simulated robots foraging in the two paths for the asymmetric experimental setup (AES) and different β values: **(a)** Arena 10 (β=1), **(b)** Arena 50 ($\beta = 10^{-1}$) and **(c)** Arena 100 ($\beta = 10^{-2}$) (30 replications). The shortest path is represented by the solid lines. In its steady-state we observe 50 % of the population in the shortest path in Arena 10, 69 % in Arena 50 and 90 % in Arena 100.

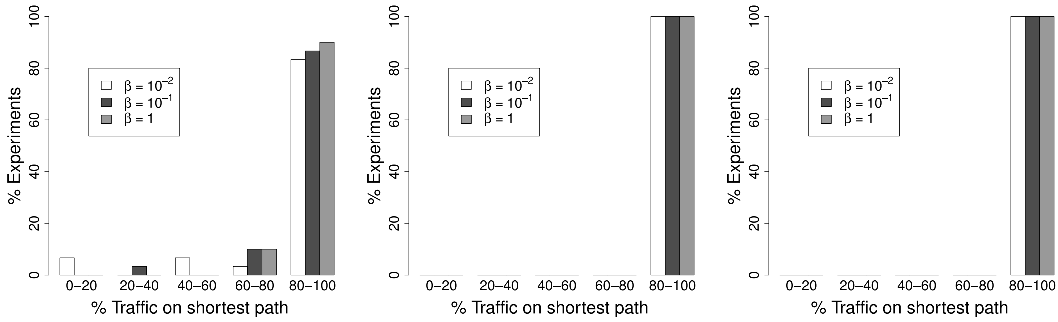
This solution is suboptimal as it gives too much weight to the estimates with low confidence levels. For $\beta \geqslant 10^{-2}$ we observe a great improvement in the retrieval outcome. However, the optimal values are not those which imply pure imitation, but a combination of both robots' information. This is because every robot has useful information about the areas' location. Useful information means that even an estimate with low confidence can be of value, especially if the confidence levels of the two estimates are close to each other or if the estimate of higher confidence is defective. Therefore, better solutions are those which take into account the estimates of all the robots, adjusting their importance in function of their associated confidence level.

In the following, we analyze the collective behavior of the robots, considering three different values for β ($\beta \in \{10^{-2}, 10^{-1}, 1\}$) in the three arenas for the two ESs. For the sake of clarity, we only show data obtained with the best β in the graphics representing proportion of robots as a function of time.

Resource selection with the asymmetric setup

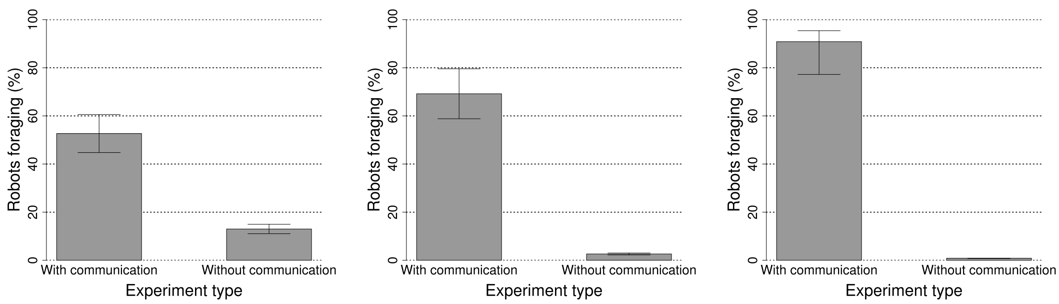
In this test, simulated robots are initialized in the AES. The robots start their random walk in the middle of the arena. Once the central place and one resource are located, the robots try to forage from central place to resource endlessly. If a robot finds an obstacle (robot or wall) while foraging, it tries to communicate with it. Then, if the obstacle is still present, the robot moves away of the obstacle until it is not detected anymore. In this way, the obstacle is avoided and the robot can continue with its foraging behavior.

Figure 5.7 shows the percentage of robots in the population that forage from one resource or the other as a function of time for different β values. A robot is considered



(a) (b) (c)

Figure 5.8: Distribution of the percentage of traffic on the shortest path for the asymmetric experimental setup (AES) in **(a)** Arena 10, **(b)** Arena 50 and **(c)** Arena 100 (30 replications). Lost robots are not considered in the dataset. For the experiments in Arena 50 and Arena 100, at the end of each experiment all the robots not considered lost are foraging using the shortest path. For Arena 10, more than 80 % of the experiments end up with most of the robots (80 %-100 %) foraging using the shortest path.



(a) (b) (c)

Figure 5.9: Percentage of robots foraging in the two paths for the asymmetric experimental setup (AES) in **(a)** Arena 10, **(b)** Arena 50 and **(c)** Arena 100 for the experiments with and without communication (30 replications). For the three arenas, the experiments in which the robots do not communicate obtain a poor foraging performance.

to belong to a path if it forages between the central place and one resource without getting lost. Remember, as shown in Section 5.1.2, that a robot is considered lost if it arrives at some coordinates where it estimates the goal area should be located but it is not.

In the first few minutes of the experiments, the robots choose any of the two paths with equal probability, depending on the first resource found. However, the shortest path starts recruiting more and more robots after some robots have already found both paths. On the longest path we observe a reduction in the number of robots foraging on it, arriving at zero in some of the trials. The recruitment on the shortest path, in any of the different configurations, increases rapidly and non-linearly. Note that the percentage of robots in the path is lower for the 10 robots in Arena 10 compared to the percentage of robots in the other arenas. The reason is that there are not enough robots in the arena to communicate in short periods of time.

Once the foraging has reached its equilibrium, we count the number of robots on the shortest path for each experiment. Figure 5.8 shows the percentage of experiments in which the robots select the shortest path for all the replications and all the β values tested in each arena. Note that this percentage of traffic is not taken from the total population but from the number of robots not considered lost. In this situation, we observe that the swarm chooses the shortest path in most of the experiments (100 % for the experiments in Arena 50 and Arena 100 and 90 % for the experiment in Arena 10).

The robots collectively choose the closest resource because when two robots coming from different resources meet (typically in the vicinity of the central place), they exchange their information according to the SGIFF. A robot which uses the shortest path, statistically has a better confidence level than the one using the longest one, because it has traveled a shorter distance since it left the resource. This confidence level difference makes the robots that use the shortest path ignore the other's information, and those using the longest path adopt the information from the robots using the shortest one. Encounters along the same path allow the robots to correct their location estimate errors according to the better information provided by robots that have a slightly better confidence level.

To study the improvement obtained through the robots' communication, we run the same experiments with a setup in which the robots do not communicate. Figure 5.9 compares the percentage of robots foraging in the two paths for the experiments with and without communication. Note that in the experiments without communication the robots are not able to create a path because the odometry errors are not collectively corrected. Therefore, the robots obtain a poor foraging performance.

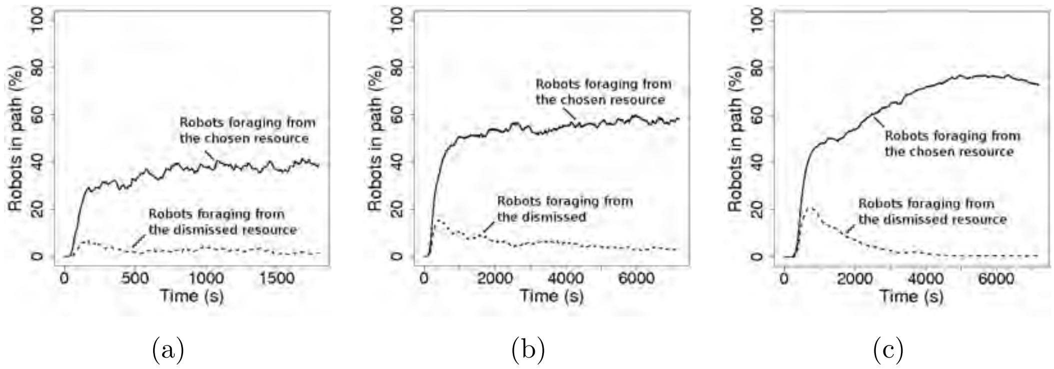


Figure 5.10: Percentage of simulated robots foraging in the two paths for the symmetric experimental setup (SES) and different β values: **(a)** Arena 10 ($\beta = 1$), **(b)** Arena 50 ($\beta = 10^{-1}$) and **(c)** Arena 100 ($\beta = 10^{-2}$) (30 replications). The chosen path is represented by the solid lines. In its steady-state we observe 40 % of the swarm in the chosen path in Arena 10, 58 % in Arena 50 and 75 % in Arena 100.

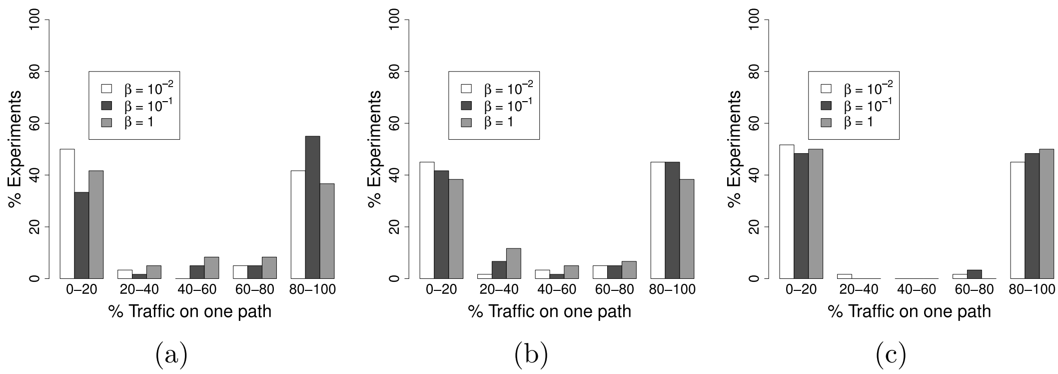
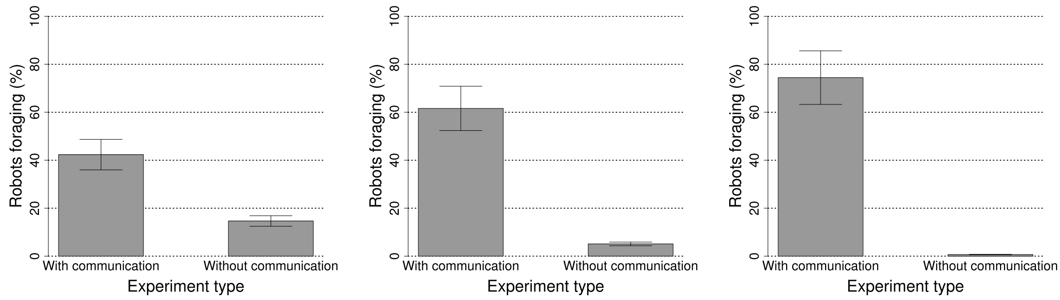


Figure 5.11: Distribution of the percentage of traffic on the selected path for the symmetric experimental setup (SES) in **(a)** Arena 10, **(b)** Arena 50 and **(c)** Arena 100 (30 replications). Lost robots are not considered in the dataset. For the three arenas robots choose with equal probability one resource area or the other.



(a) (b) (c)

Figure 5.12: Percentage of robots foraging in the two paths for the symmetric experimental setup (SES) in **(a)** Arena 10, **(b)** Arena 50 and **(c)** Arena 100 for the experiments with and without communication (30 replications). For the three arenas, the experiments in which the robots do not communicate obtain a poor foraging performance.

Resource selection with the symmetric setup

In this test robots are initialized in the SES, where the two resource are at the same distance from the central place. Because of this symmetry, the robots cannot collectively choose one single preferred resource based on the distance. In fact, when two robots arriving at the central place from different resources exchange their estimated resource information, the controller fuses the two robots' data sending the robot to an erroneous location at some point in between the two areas. In an ideal example, two robots which arrive at the central place from different resource areas which are at the same distance from the central place, will have the same confidence level value ($p^{resrc,i}_{k|k-1} = p^{resrc,j}_{k|k-1}$). Therefore, $\Delta p^{resrc,ij}_{k|k-1} = 0$ and $g^{resrc,i}_{k} = g^{resrc,j}_{k} = 0.5$; hence $\hat{\boldsymbol{x}}^{resrc,i}_{k|k} = \hat{\boldsymbol{x}}^{resrc,j}_{k|k} = \frac{1}{2}\left(\hat{\boldsymbol{x}}^{resrc,i}_{k|k-1} + \hat{\boldsymbol{x}}^{resrc,j}_{k|k-1} + \boldsymbol{x}^{ij}_{k}\right)$.

However, results in Figure 5.10 show that the robots succeed in choosing one of the resource. The reason is that if after the first meeting a robot meets other robots, the filter will correct its estimates and hence will guide the robot to the correct goal. Nonetheless, the way robots weigh their estimates and the equal distance between the central place and the resource make the swarm take more time to choose its dominant path collectively and the number of lost robots increases, as observed if comparing Figure 5.7 with Figure 5.10. Figure 5.11 displays the percentage of experiments in which the swarm choose one of the paths for each arena. We can observe that the robots choose one path or the other with the same probability.

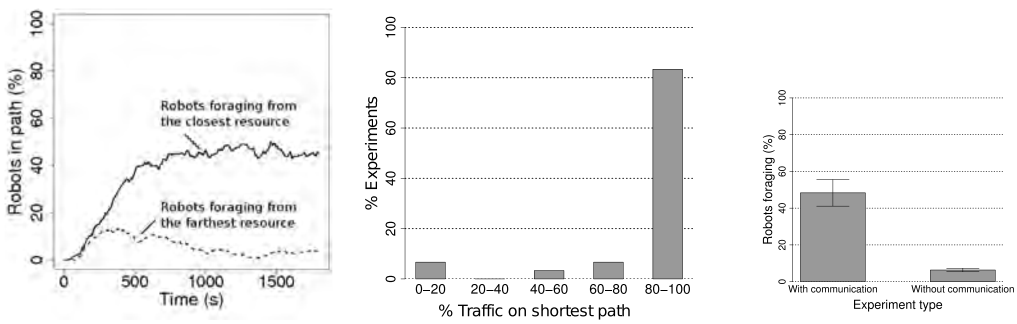
As in the asymmetric experiment, we run the same experiments with a setup in which the robots do not communicate. Figure 5.12 compares the percentage of robots foraging in any of the two paths in the experiments with and without communication. As in the AES, in the experiments in which the robots do not communicate, the swarm is not able to create a path because the odometry errors are not corrected.

5.2.2 Experiments with real robots

Experiments with real robots were carried out with 10 e-pucks in an arena of dimensions $1.2 \times 1.7\,\mathrm{m}^2$ and lasted 1800 s. Robots start in in the middle of the arena, scattered in a circle of 0.2 m radius with random position and orientation. Data collection is managed through the Bluetooth connection. Because of Bluetooth limitations, two different computers are used to communicate with 5 different robots each. Robots are initialized simultaneously and keep track of a timer which is initialized at the beginning. This timer allows the robots to have the same time reference axis, which is not used by the robots to perform the task but it is used by us for evaluation purposes. When a robot arrives at one of the two goal areas (i.e., central place or resource) or it gets lost, the robot sends a Bluetooth command to the computer, indicating the state in which it finds itself (i.e., central place, resource or lost) and the time at which the event takes place. After 1,800 s the controller stops and robots are randomly re-initialized. The 8 IR proximity sensors are used as input to the obstacle avoidance and communication behaviors. The central place and resource areas are detected with the ground sensors. The communication range of the range and bearing board has been limited to 15 cm, as in the simulation experiments, to avoid the IR signals spread over the arena.

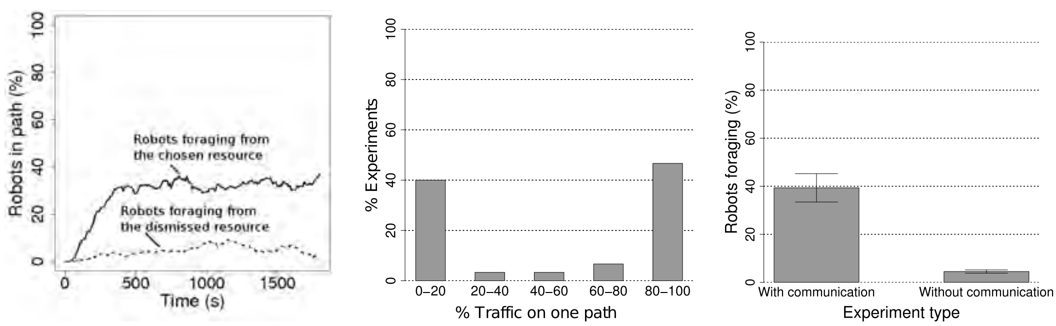
We selected $\beta = 1$ from the simulation results and ported the controller to the real robot. Figure 5.13a shows results of the exploration of the robots as a function of time for the AES. We observe that robots are able to collectively choose to forage from the closest resource, where more than 80 % of the robots not considered lost end up foraging using the shortest path (see Figure 5.13b). Note that in 7 % of the experiments, less than 20 % of the foraging robots were using the shortest path. This situation was observed in experiments in which a small percentage of robots (fewer than three robots) were foraging in the arena while the rest of the group was lost. If we look at Figure 5.13a and Figure 5.13b simultaneously, we observe that on average 4-5 robots are lost, 4-5 robots are foraging using the shortest path and 1 robot is foraging using the longest path. Finally, a comparison between the experiments with and without communication is shown in Figure 5.13c. We observe that the robots obtain a poor foraging performance in the experiments without communication as already observed in the simulation experiments. Figure 5.14 shows results of the experiments in the SES. As already observed in the experiments with simulated robots, the swarm is able to collectively choose one of the two resource offered.

Notice that there are quantitative differences between the real and the simulated experiments. These differences are mainly due to the imperfect communication model of the range and bearing sensor implemented in simulation. Another problem is given by the reflections caused by the borders of the arena that distort some of the bearing measurements when robots are near the borders. Moreover, we have observed some interferences between the range and bearing sensor and the IR proximity sensors. These



(a) (b) (c)

Figure 5.13: Results for the experiments with real robots in the asymmetric experimental setup (AES) and $\beta = 1$ (30 replications). **(a)** Percentage of e-puck robots foraging in the two paths. In its steady-state we observe 45 % of the population using the shortest path. **(b)** Distribution of the percentage of traffic on the shortest path. Lost robots are not considered in the dataset. More than 80 % of the robots that are not lost are foraging using the shortest path at the end of the experiment. **(c)** Comparison of the percentage of robots foraging on the two paths for the experiments with and without communication. The robots obtain a poor foraging performance in the experiments without communication.



(a) (b) (c)

Figure 5.14: Results for the experiments with real robots in the symmetric experimental setup (SES) and $\beta = 1$ (30 replications). **(a)** Percentage of e-puck robots foraging in the two paths. In its steady-state we observe 35 % of the population using the chosen path. **(b)** Distribution of the percentage of traffic on the selected path. Lost robots are not considered in the dataset. Robots choose with equal probability one resource area or the other. Once collectively chosen, less than 5 % of the robots forage using the other path. **(c)** Comparison of the percentage of robots foraging in the two paths for the experiments with and without communication. The robots obtain a poor foraging performance in the experiments without communication.

alterations in the range and bearing produce extra errors not modeled in the simulation, which make the robots miscalculate the information given by other neighbors. However, we observe the same behavior in all the qualitative measures and only a decrease of less than 10 % in the number of robots foraging in the chosen or shortest path.

5.3 Discussion and conclusions

In this chapter, we presented a mechanism of collective localization which uses local communication to let the robots adopt or ignore information given by their neighbors. We conducted a central place foraging experiment in which robots rely on collective localization to repeatedly travel between the resources and the central place.

We first investigated the impact of the β parameter, which determines the importance of the relative confidence levels in the localization process. We found that the β parameter gives best results when it gives moderate imitation. On the one hand, pure imitation should be avoided because it allows an erroneous estimate to rapidly propagate if a robot is mistaken. On the other hand, averaging information is also detrimental because it discards valuable information. In addition, when we introduced two resources, robots could sometimes produce useless information by averaging estimates about two different locations. This problem had limited impact on the collective behavior because a sufficient level of imitation makes averaging less likely. The negotiation mechanism investigated in the previous chapter can be seen as a special instance of the social fermi filter with β set to a very small value, so that individuals average all their opinions. Hence, our results demonstrate that negotiation is not suited for problems in which some individuals have better information than the others, in particular the collective localization problem.

Social odometry was originally designed to allow robots to better localize themselves in their environment, but we also found that it triggers a collective choice with the robots exploiting the closest resource. This is because social odometry contains a feedback that favors the closest resource. Indeed, robots that forage at that resource travel less and therefore they have on average better confidence levels in their estimates. As a consequence, their estimates have higher chances to spread in the population. In addition, as more robots exploit a resource, the probability for a new robot to meet and join them increases too. The combination of these two phenomena leads to a collective choice in favor of the nearest resource.

The decision making mechanism reported in this chapter demonstrates how a group can take advantage of relevant information carried by informed individuals, here the robots that travel less. The confidence level associated to the opinions allows the group to selectively propagate the most relevant information.

In the next chapter, we perform another resource selection experiment in which individuals have no direct knowledge that can be used to make the collective decision. Instead, the decision is gradually shaped by a stigmergic process simulated directly inside the whole group of robots.

Chapter 6

Resource selection with artificial pheromone

The previous chapters focused on mechanisms based on direct information transfer. Here, we investigate a method that allows robot swarms to make collective decisions using indirect information transfer.

Indirect information transfer — also known as stigmergy — is achieved by modifications of the environment. This type of communication is very common in nature and human societies. For instance, wasps rely on stigmergy to build their nest (Theraulaz and Bonabeau, 1995); ants deposit pheromone on the ground, which is a chemical signal that can be used to select a resource (Deneubourg and Goss, 1989). On the internet, we create collaborative content and make collective decisions by aggregating tiny pieces of information such as ratings, comments or text edits.

Despite the potential of indirect information transfer, it is almost never employed in swarm robotics. This is mostly due to the difficulty and impracticality of modifying the environment: it would be rather boring if robots had to deposit stones here and there to navigate, just as humans do when they hike in the mountains. Less intrusive signals such as pheromone would be unreliable in human environments.

We envision two possible solutions to this problem. The first one is to rely on sensors spread in the environment. This solution becomes more and more realistic as RFID chips invade our daily life (Finkenzeller, 2010). If robots are able to communicate indirectly through these chips, they can implement stigmergy in their collective behaviors. However, this solution has limited interest as long as sensors are not deployed in the environment in which robots operate. Also, if robots are placed in a new, unprepared environment, this approach will not work.

The second solution, which is the one we explore in this chapter, consists in using the robots themselves to form a network, a mesh that covers the environment and inside which indirect information transfer can occur.

In this chapter, we investigate a decision making situation in which some robots have key information, while the other robots are only creating a network inside the environment. No robot has the necessary information to directly make the decision. Rather, the collective decision is gradually elaborated through indirect information transfer inside the network of robots.

The foraging experiment involves two resources that are situated at different distances from the central place. We consider that robots are already forming physical chains between the central place and the resources. These chains may be the product of the chain controller (detailed in Nouyan *et al.*, 2008). Single robots only know where their immediate neighbors are, and they can detect resources or the central place in close range. None of the robot knows what is its distance to the central place, or which resource is the most interesting to exploit. The collective decision of the robots is the product of the activity of virtual ants injected inside the network of robots. These virtual ants travel from the resources to the central place, back and forth, laying artificial pheromone on the nodes of the network. The level of artificial pheromones on the nodes therefore reflects the collective choice of the virtual ants. Using this process, the robots can determine if they contribute to mark out the path to the most profitable resource.

The rest of this chapter is organized as follows. In Section 6.1.1 we expose our experiment in detail along with the setup used. In Section 6.1.2 we present the controller used to simulate vitals ants and artificial pheromone inside the network formed by the swarm. An analytical model of the collective behavior is described in Section 6.1.3. We then proceed with results in Section 6.2. We start with a study of the analytical model and provide a large picture of how the system responds to various parameters. Then, we investigate the selection of the path to the most profitable resource with simulated and real robots. The remaining sections are devoted to the study of properties of the system, including plasticity and robustness to communication errors. Finally, Section 6.3 discusses results, perspectives and concludes the chapter.

6.1 Methods

6.1.1 The task and the experimental setup

We have devised a simple task in which robots are offered two resources. We focus on the path selection process and assume that robots have already established chains between the central place and each resource using, for example, a mechanism such as the one presented by Nouyan *et al.* (2006, 2008). Once chains are formed, robots do not move anymore and simply exchange messages.

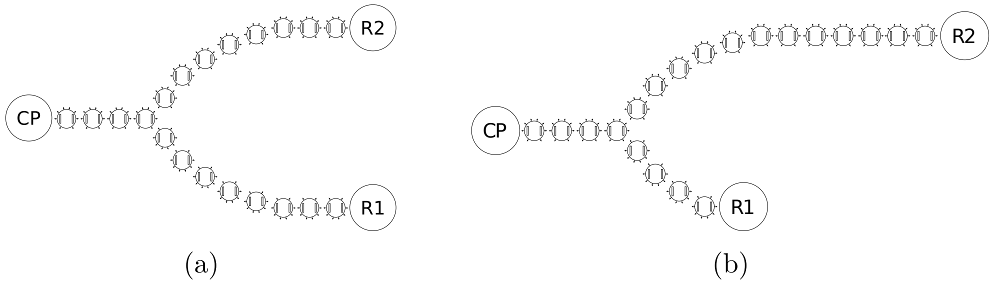


Figure 6.1: **(a)** Setup with resources at identical distances. **(b)** Setup with resources at different distances. CP denotes the central place, R1 and R2 denote resources 1 and 2 respectively.

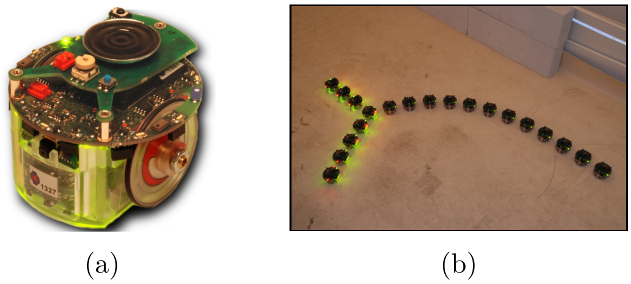


Figure 6.2: **(a)** The e-puck robot. **(b)** A swarm of 20 e-pucks arranged in a long and a short path for the setup with resources at different distances.

Figure 6.1 shows the two different setups employed. In the first setup, the two resources are situated at equal distance from the central place, while in the second setup, one resource is closer to the central place than the other. We always use a total of 20 robots that are positioned at 10 cm intervals to form chains.

The task allows us to test whether the robots are able to (i) perform a collective choice when offered two resources at identical distance, (ii) modulate their choice depending on the distances of the available resources from the central place, and (iii) exhibit plasticity after sudden environmental changes.

We have used e-puck robots equipped with the range and bearing board (Gutiérrez *et al.*, 2008, 2009a). The board (described in Chapter 3) allows the robots to communicate locally, obtaining at the same time both the range and the bearing of the emitter without the need for any centralized control or external reference. In this study, we have limited the emission range to 13 cm and used 8 sensor/actuator pairs in order to limit the size of our experimental setup, to reduce the probability of message collisions and to make sure that a message is specifically sent to a given robot and cannot be perceived by another robot in the neighborhood.

In the simulator, all the simulated sensors and actuators are modeled after measures taken on real robots. In order to simulate effectively the sensory input of real robots, we added noise to the simulated sensory data. In the range and bearing communication system, noise is added to the range (± 2.5 cm) and bearing ($\pm 20\,°$) values. By default, messages transmitted with the range and bearing system were delivered with no errors. For the study of robustness to communication errors (see Section 6.2.6), we implemented a probability that a message became corrupted. Corrupted messages could be either lost or their content could be altered into any other possible content.

6.1.2 Robots' controller

The robots' controller is designed to produce a selection mechanism at the collective level. The main part of the selection mechanism consists of virtual ants traveling along the chains of robots from the central place to the resource, and vice-versa. Virtual ants are implemented as messages transmitted from robot to robot (called VA_{CP} when going towards the central place and VA_{R} when going towards a resource).

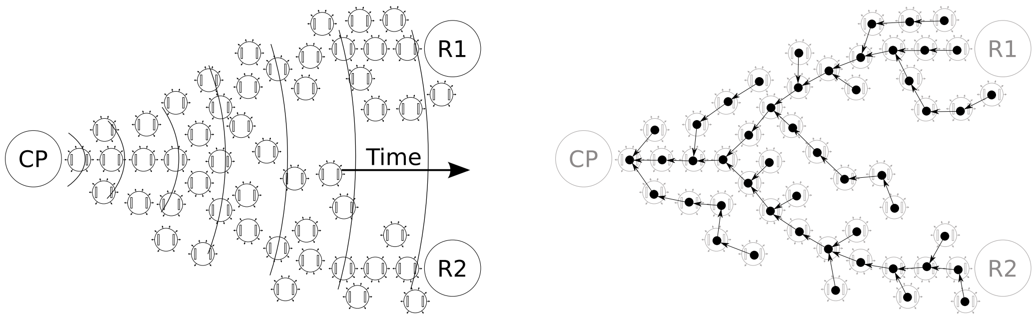
The virtual ants deposit artificial pheromone on the robots forming the chain and indicating the path. Because virtual ants take less time to traverse a shorter path, in a given amount of time they will make more trips on short paths than on long ones. Shorter paths will therefore receive larger amounts of pheromone than longer paths. The amplification process will increase further this difference: relay robots preferentially send virtual ants in directions that have already higher amounts of pheromone. Amplification determines a collective choice even when resources are at identical distances. In fact, even though on average these paths receive the same amount of pheromone, a slight and random difference in pheromone concentrations can be amplified and lead to a final collective choice (Pasteels *et al.*, 1987; Deneubourg and Goss, 1989; Deneubourg *et al.*, 1990).

Only robots that perceive either the central place or a resource have the possibility to emit new virtual ants. These ants are then relayed by any robot that perceives them until they reach their destination. There is no fixed population of virtual ants: they are created at a rate determined by the emitters' concentration of pheromone and disappear if they are not relayed, typically when reaching the extremity of a chain.

Establishing the communication network

Before emitting or relaying virtual ants, robots need to know in which direction to send them. In a first stage, the robots organize themselves in a network with a shortest path tree topology that allows ants emitted from a resource to travel straight to the central place along the shortest path.

To do so, the robot perceiving the central place probabilistically emits wavefront signals. These signals propagate through the swarm and inform each robot of the swarm about which of its immediate neighbors is closer to the central place (see Figure 6.3a).



(a) (b)

Figure 6.3: The robots organize themselves in a network with a shortest path tree topology. **(a)** The robot perceiving the central place emits a wavefront signal that is in turn retransmitted by every robot perceiving it, thus propagating the signal through the entire swarm. **(b)** Each robot remembers the direction from which the wavefront was perceived first, creating the desired topology at the collective level.

In practice, wavefront signals are implemented by a particular message (called SPT) which is first emitted at the central place. Upon detection of an SPT message, a robot interrupts any current action and repeats the message, sending it in all directions. To ensure that the messages are propagated only forward, robots that have just sent an SPT message enter an inhibited state of $h = 0.35$ seconds during which they do not receive or send any message.

As described in Figure 6.3b, this method creates a communication network with a shortest path tree topology and is general enough to work with any arrangement of the robots, as long as the swarm forms a single connected network. We have also implemented a regular update of the communication network because it is the most sensitive element of the system. If a communication error happens during the construction of the shortest path tree, or if a robot moves, our assumption that virtual ants are able to travel between the central place and the resources may not be verified anymore. To cope with these unlikely but possible events, the communication network is updated by sending new wavefront signals at a rate of $u = 0.01$ updates per second.

The motion of virtual ants within the network

Since virtual ants are messages in the system, the emission and relay of these messages within the communication network is achieved by the robots themselves.

Figure 6.4 provides a detailed view of a robot relaying a virtual ant. Robots have a total of 8 pairs of sensors/actuators, positioned around their body, that receive or send virtual ants. The physical location of the sensors/actuators allows the robots to infer the incoming direction of virtual ants and also to send them in specific directions.

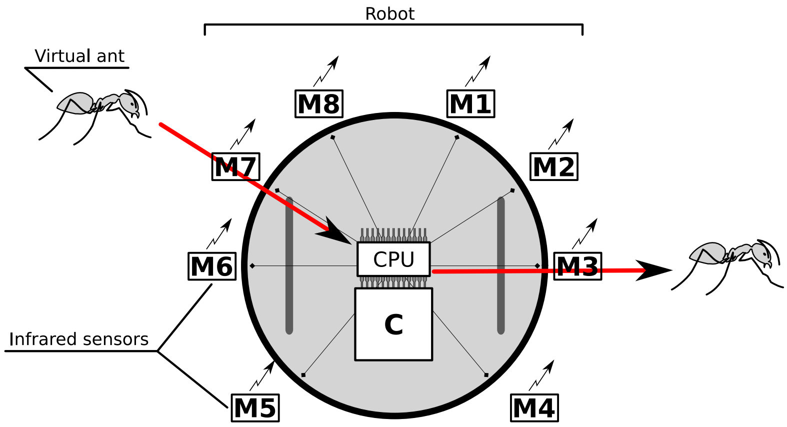


Figure 6.4: Detailed schema of a robot relaying a virtual ant. During this process, the robot receives the virtual ant on sensor/actuator $j = 7$, updates the memory register M_j associated to sensor/actuator j as well as the concentration of pheromone C, and finally transmits the virtual ant in a new direction through another sensor/actuator $i = 3$.

Because of infrared reflections on the ground or on the body of other robots, a robot might perceive an ant he just relayed ahead of himself. To avoid that a robot relays several times the same ant by mistake, we use two different messages for ants going to the central place (VA$_{CP}$) or ants going to the resources (VA$_R$). For instance, a virtual ant VA$_{CP}$ perceived by a robot in the direction of the central place will not be relayed again by this same robot because it is already heading towards its destination.

There are 8 memory registers $M_1, \ldots, M_8$ associated to the eight pairs of infrared sensors/actuators. When a virtual ant is perceived by the sensor/actuator i, memory M_i is updated by adding an amount $l \in]0, 1]$ of pheromone:

$$M_i = M_i + l. \tag{6.1}$$

Therefore, Equation 6.1 implements the same fundamental principle as described by Goss *et al.* (1989) to explain path selection in real ants. In addition, the relaying robot updates its concentration of pheromone C according to Equation 6.2:

$$C = C + (1 - C)l. \tag{6.2}$$

When updated, the concentration of pheromone C converges asymptotically to 1. Pheromone concentration is used to tell, after convergence, whether a robot is on the selected path: this is the case if a robot's C value exceeds a given threshold. C is also the rate per second at which robots may emit new ants. Note that only robots perceiving the central place or the resources may emit new ants (that is create them), other robots are only relaying these ants.

When a relaying robot perceives with sensor/actuator j an ant going to a resource, it has to decide in which direction to send it. The robot calculates the probability $P(i,j)$ to send this ant with actuator i using Equation 6.3:

$$P(i,j) = \frac{{M_i}^s}{\sum_{k \neq j} {M_k}^s}, \tag{6.3}$$

where $s > 1$ is a factor that introduces non-linearity in the system by amplifying small differences among possible directions. For instance, if we consider that $s = 2$ and there are only two sensors/actuators and one of them has received $10\,\%$ more pheromones than the other, the probability to relay a virtual ant with that sensor/actuator is higher by $21\,\%$: $110^2/(110^2 + 100^2) = 1.21 \cdot 100^2/(110^2 + 100^2)$.

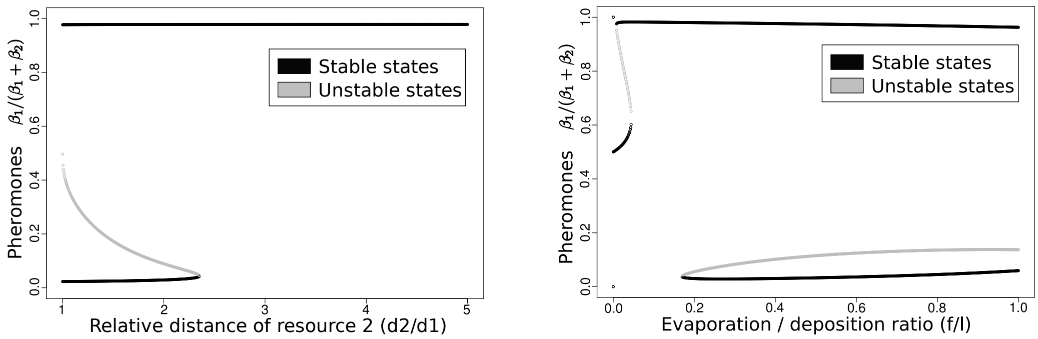
Similarly to its natural counterpart, artificial pheromone evaporates at a constant rate f per second:

$$C(t + \Delta t) = C(t) - \Delta t \cdot C(t) \cdot f, \tag{6.4}$$

$$M_i(t + \Delta t) = M_i(t) - \Delta t \cdot M_i(t) \cdot f \quad \forall i \in [1, 8], \tag{6.5}$$

where t is the time and Δt is the controller time step. This evaporation ensures that pheromone concentration decreases on a path that is not selected. The selection of a path is maintained only if virtual ants keep on traveling along the path and depositing artificial pheromone. If a sudden change in the environment makes a path unavailable, the swarm slowly forgets it. However, ignored paths may need to be reactivated. For this purpose, we define a minimum concentration of pheromone ϵ for every robot that guarantees a minimal traffic of virtual ants.

An experiment typically unfolds with robots waiting for an initial wavefront signal that will create the first shortest path tree. Once the wavefront signal is received, robots at the resources emit virtual ants towards the central place. When receiving virtual ants, the robot at the central place starts sending virtual ants as well. The virtual ants are emitted at a rate that is equal to the pheromone concentration of the emitting robot. Initially, the pheromone concentration on all the robots is set to 0.5. We consider that the selection process is completed when the pheromone concentration of the non selected path has decreased below a predefined threshold.



(a) (b)

Figure 6.5: Equilibria of the analytical model when pheromone concentrations are stable. **(a)** Impact of distances ratio d_2/d_1. **(b)** Impact of evaporation/deposition ratio f/l.

6.1.3 Analytical Model

The previous definitions of the robots' controller and task involve several parameters that may change, such as the distances d_1 and d_2 of the two resources to the central place, as well as the evaporation rate of pheromone f (forgetting process) and the amount of pheromone l deposited by a virtual ant on a robot during relaying (learning process).

To understand the impact of these parameters on the collective behavior of the robots, we devise a mathematical model and study its steady states along with the probabilities to reach them. Our approach is not aimed at finding the best parameters for a specific case. Rather, we try to obtain a better understanding of the collective behavior and to extract general rules in order to parameterize effectively the robots in the subsequent experiments.

We model the concentrations of pheromone at the central place (CP) and at the resources (R1 and R2) over time with the variables α, β_1, and β_2 respectively (see Table 6.1). As said in Section 6.1.2, pheromone concentration is used to set the rate of emission of virtual ants (ants/s). Hence, the number of ants emitted at R_i over a time interval T is given by $m_i = \beta_i T$. We assume that the transmission of an ant lasts exactly h seconds, so the number m_i of ants that can be emitted at R_i over a period T must satisfy the constraint $m_i h \leqslant T$.

When a robot is relaying a virtual ant, it cannot receive another ant at the same time because of hardware limitations. If another ant is sent to the robot during transmission, we say that a *collision* happens. The first ant is transmitted while the other one is lost. When $n > 1$ resources are present in the environment, the possible paths to the central place intersect. Robots at intersections may receive ants coming from each resource and are likely to experience collisions more frequently than others. For this reason, we include collisions at intersections in our model to take their effect into account. We call

x_i the number of ants per second emitted at R_i not experiencing a collision. At any moment, the probability p_i that an ant emitted at R_i collides at the intersection point with another ant emitted at R_j ($j \neq i$) is given by:

$$p_i = \frac{(x_j T)h}{T} = x_j h.$$

Hence, the rate of ants emitted at R_i and not experiencing collisions is: $x_i = \beta_i(1 - p_i) = \beta_i(1 - x_j h)$. We obtain a system of equations that can be solved using a simple Gaussian elimination:

$$\begin{cases} x_i = & \beta_i(1 - x_j h) \\ x_j = & \beta_j(1 - x_i h) \end{cases} \iff \begin{cases} x_i = & \dfrac{\beta_i - \beta_i \beta_j h_j}{\beta_i \beta_j h_i h_j - 1} \\ x_j = & \dfrac{\beta_j - \beta_i \beta_j h_i}{\beta_i \beta_j h_i h_j - 1}. \end{cases}$$

Following the description of the robots' controller given in Section 6.1.2, the dynamics of α and β_i are modeled by the following differential equations:

$$\begin{cases} \dfrac{d\alpha}{dt} = -f(\alpha - \epsilon) + l(1 - \alpha) \sum_i \varphi_i, \\ \dfrac{d\beta_i}{dt} = -f(\beta_i - \epsilon) + l(1 - \beta_i) \alpha \dfrac{\varphi_i^s}{\sum_k \varphi_k^s}. \end{cases}$$

where $\varphi_i = x_i v / d_i$ is the number of ants traveling along the path from R_i to the central place.

6.2 Results

In the following, we evaluate how the parameters of the system influence the collective behavior using the analytical model. We then report results obtained from experiments with real and simulated robots. For each setup with resources at identical or different distances, we perform 20 repetitions. Experiments concern the collective choice of paths, the plasticity of the selection process, and the robustness to communication errors.

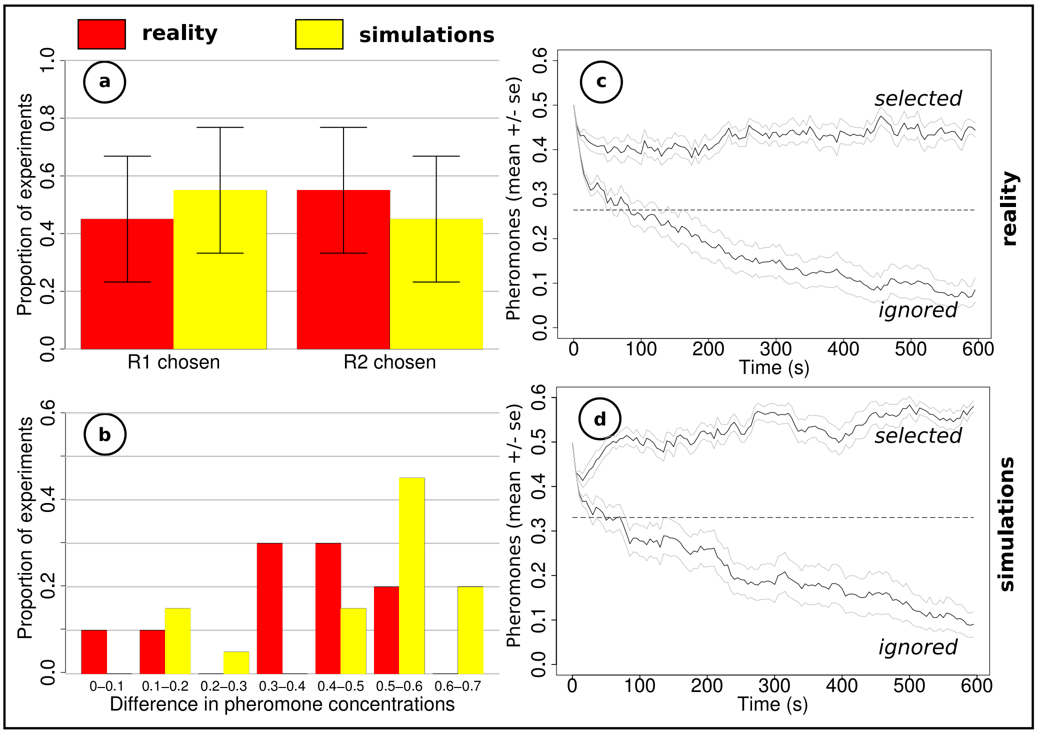


Figure 6.6: Results with real and simulated robots for setups with resources at identical distances. **(a)** Proportion of repetitions that ended with the choice of R1 or R2. **(b)** Distribution of differences in pheromone concentrations ($|\beta_1 - \beta_2|$), accounting for the discrimination of the swarm among the paths. **(c, d)** Average pheromone concentrations over time at the selected and ignored paths.

6.2.1 Parameter Study

We study the stable states of the analytical model when pheromone concentrations reach a steady state. We focus on the influence of the relative distances of resources, as well as the deposition and evaporation coefficients. The values of parameters h and v in Table 6.1 are set to the maximum allowed by the hardware communication system at the time of the study. The amplification factor s is set to 2 (it must be strictly higher than 1 in order to introduce non-linearities and trigger amplification). Table 6.1 gives the default values of the parameters.

We define the relative concentration of pheromone $k = \beta_1/(\beta_1 + \beta_2)$, a coefficient to study path selection. If $k \approx 0$, the concentration of pheromone at R1 is negligible while it is high at R2. In this case, we say that the path to R2 is selected. Conversely, if $k \approx 1$, the path to R1 is selected. If $k \approx 0.5$, the swarm is unable to select a single path.

Figure 6.5a shows coefficient k when the relative distance to R2 increases. When resources are at the same distance, we observe two stable states with one unstable state at $k = 0.5$. The swarm selects indifferently one of the two paths. As soon as R2 is

Variable	Description		Initial
α	rate of ants emitted at central place		0.5 ants/s
β_i	rate of ants emitted at R_i		0.5 ants/s
Parameter	**Description**		**Default**
n	number of resources in the experiment		2
s	amplification factor		2
l	pheromone deposited (learn)		0.88
f	evaporation rate (forget)		0.029
d_1	distance of R1 to central place	(identical distances)	1.2 m
		(different distances)	0.8 m
d_2	distance of R2 to central place	(identical distances)	1.2 m
		(different distances)	1.6 m
h	duration of inhibition period		0.35 s
v	speed of a virtual ant		5 m/s
ϵ	minimum pheromone concentration		0.016 ants/s
u	update rate of the network topology		0.01 /s

Table 6.1: Summary of variables and parameters of the experiment.

farther than R1 the stable state $k \approx 1$ in which the path to R1 is selected has a larger basin of attraction and the probability to converge there is increased. When the ratio of distances $d_2/d_1 > 2.5$, the swarm always selects the path to R1.

Figure 6.5b shows the impact of the evaporation and deposition ratio (f/l) on the model's stable states when the ratio of distances $d_2/d_1 = 2$. We have set the deposition coefficient to $l = 0.88$ in order to achieve a relatively fast collective behavior. We observe that low values of f generate a third stable state in which $k = 0.5$ and the swarm is unable to make a selection. When $f/l < 0.2$, the model may have up to three stable states, and depending on noise and initial conditions, it may not be able to select a path. As soon as $f/l > 0.2$, the model has two stable states in which the swarm selects the path to either R1 or R2. The unstable state is very close to 0 and we observe a larger basin of attraction of stable state $k = 1$ in which the path to R1 is selected.

Based on the previous observations, we have set parameters d_1, d_2, l, f with the values reported in Table 6.1. Unless stated otherwise, these settings are used to perform the experiments reported in the following section.

6.2.2 Resources at identical distances

When resources are located at the same distance from the central place the swarm arbitrarily selects one of the paths (binomial test, $p > 0.35$; see Figure 6.6). The swarm's collective choices in reality and in simulations do not differ statistically (proportions test, $p > 0.75$). At the beginning of the experiments, the concentrations of pheromone

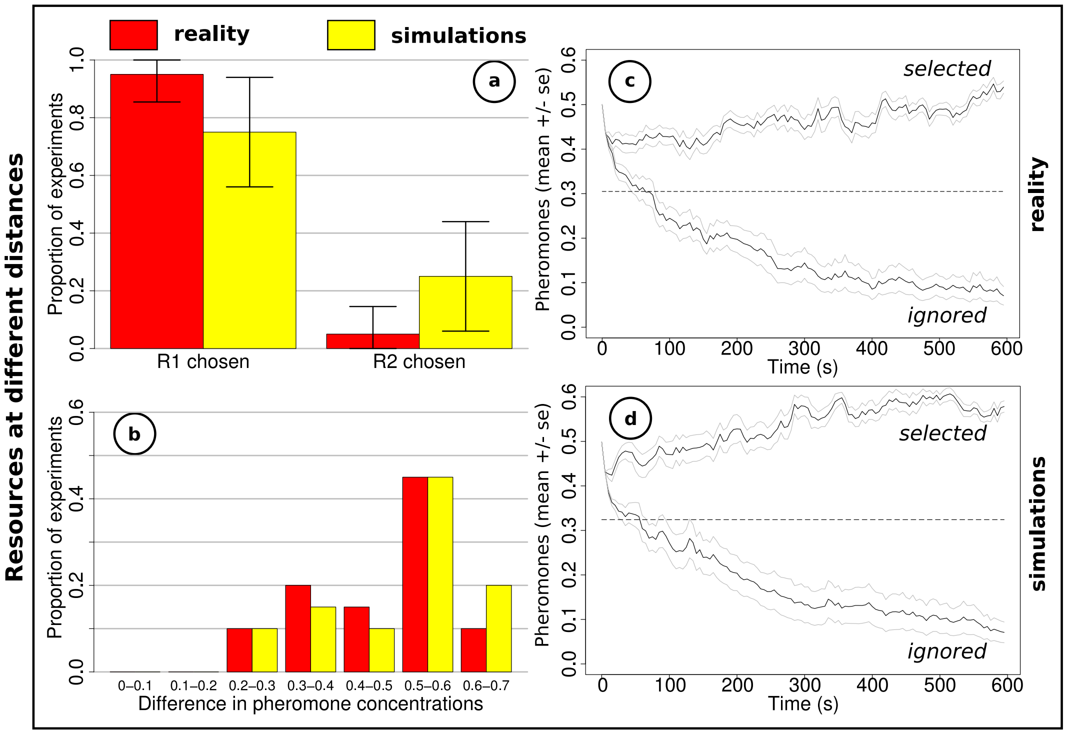


Figure 6.7: Results with real and simulated robots for setups with resources at different distances. **(a)** Proportion of repetitions that ended with the choice of R1 or R2. **(b)** Distribution of differences in pheromone concentrations ($|\beta_1 - \beta_2|$), accounting for the discrimination of the swarm among the paths. **(c, d)** Average pheromone concentrations over time at the selected and ignored paths.

on the two paths are similar and virtual ants are routed towards either resource. The amplification of random fluctuations in the emission of ants triggers the swarm's collective choice. One of the two paths receives more and more virtual ants laying artificial pheromone, while the pheromone concentration on the other path decreases quickly to the allowed minimum. The selection achieved by the swarm is stable, both in reality and in simulations. In simulations, most of the experiments ($85\,\%$) end with a clear selection, where the difference in pheromone concentrations is high enough to define a threshold (> 0.25) for discriminating among paths. With real robots, we observe that $80\,\%$ of the experiments end with a clear discrimination of the paths. The remaining experiments exhibit differences that are too small to allow discrimination. This slight difference between simulations and reality probably arises because of two related factors that are reducing the strength of the amplification process: the non-modeled collisions of messages traveling in opposite directions and the vanishing of virtual ants.

6.2.3 Resources at different distances

When resources are located at different distances the swarm selects the path to the closest resource more often (binomial test, $p < 0.001$). As shown in Figure 6.7, that path is chosen in $95\,\%$ of the cases with real robots. The concentration of pheromone is maintained for the chosen path, while it quickly drops for the other one. We observe a good agreement between reality and simulations, both for the evolution of pheromone concentrations in time and the differences in concentrations. However, with real robots concentrations of pheromone are not as high as expected. Again, this behavior can be explained by the non-modeled collisions of virtual ants. This phenomenon is also responsible for the better capability of the swarm to choose the path to the closest resource in reality, although this difference is not statistically significant (proportions test, $p > 0.18$). Collisions of virtual ants are more likely to occur on a longer path because virtual ants spend more time traveling from one end to the other. Hence, the path to the farthest resource is more affected by the negative feedback of collisions, which results in the swarm discriminating more accurately among the two resources.

6.2.4 Resources of different qualities

The system proposed is not bound to select paths only based on their length. It is also possible for the robots to modulate their collective choice as a function of the quality of the resources detected. We make the assumption that robots can evaluate the quality of the resources when they detect them and that they can translate quality into a number $Q \in [0, 1]$. For this specific experiment, we modify the behavior of the robots so that if they detect a resource, they emit virtual ants at the rate $r = C \cdot Q$ where C is the pheromone concentration on the robot, and Q is the quality of the resource detected.

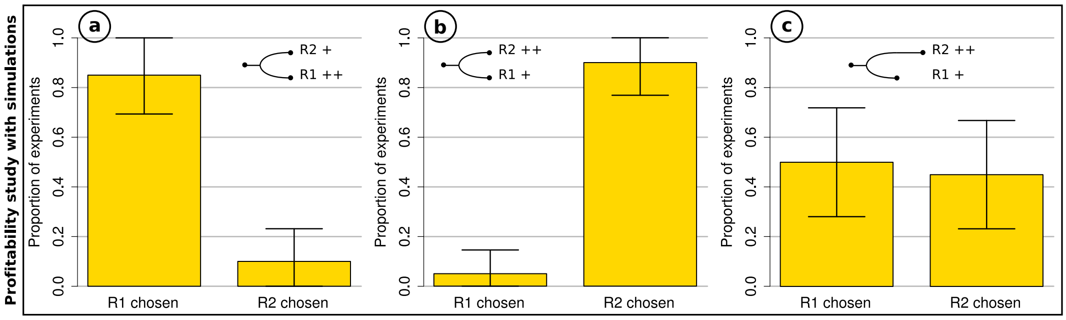


Figure 6.8: Selection of the resources based on their quality. We report the proportion of trials that end up with the selection of R1 or R2 in three different sets of simulation experiments. **(a, b)** Resources are situated at identical distances from the central place. **(a)** R1 has a higher quality than R2, **(b)** R1 has a lower quality than R2. Robots choose the resource with best quality. **(c)** Resources are situated at different distances and their respective qualities are tuned to compensate for the effects of distance. The choice of the robots is randomly in favor of R1 that is closer but with a lower quality, or R2 that is farther but with a higher quality.

We run two simulation experiments using resources at identical distances. In one experiment, resource R1 has a quality set to $Q_{R1} = 1$, while resource R2 has a quality $Q_{R2} = 0.8$. In the second experiment, the qualities of the resources are inverted. Figure 6.8a and b summarize the collective choice of the robots in these two experiments. When R1 has a better quality than R2, it is selected in $85\,\%$ of the trials. In the symmetrical situation where R2 has a better quality than R1, the robots select R2 in $85\,\%$ of the cases. This contrasts with the previous experiments in which resource quality is not defined and only distance is taken into account by the robots, and shows that the collective choice of the robots can be steered by the quality of the resources detected.

It is worth mentioning that the collective choice of the robots depends on the resources profitabilities which are function of their distances and qualities. To show this duality, we run simulation experiments with resources at different distances in which the quality of the resources ($Q_{R1} = 0.92$; $Q_{R2} = 1.00$) are specifically tuned to compensate for the effects of distance ($d_1 = 0.8$ m; $d_2 = 1.6$ m). In Figure 6.8c, we observe that the robots do not distinguish between the two offered resources and they select randomly R1 or R2 (binomial test, $p > 0.74$). This is because R1 is closer but with a lower quality, while R2 is farther with a higher quality.

6.2.5 Plasticity of the selection process

To test the response of the swarm when there is a sudden environmental change, that is, when a selected resource becomes unavailable or when a new resource appears in the environment, we run three sets of experiments with real robots. In all these experiments, we use the setup with resources located at different distances.

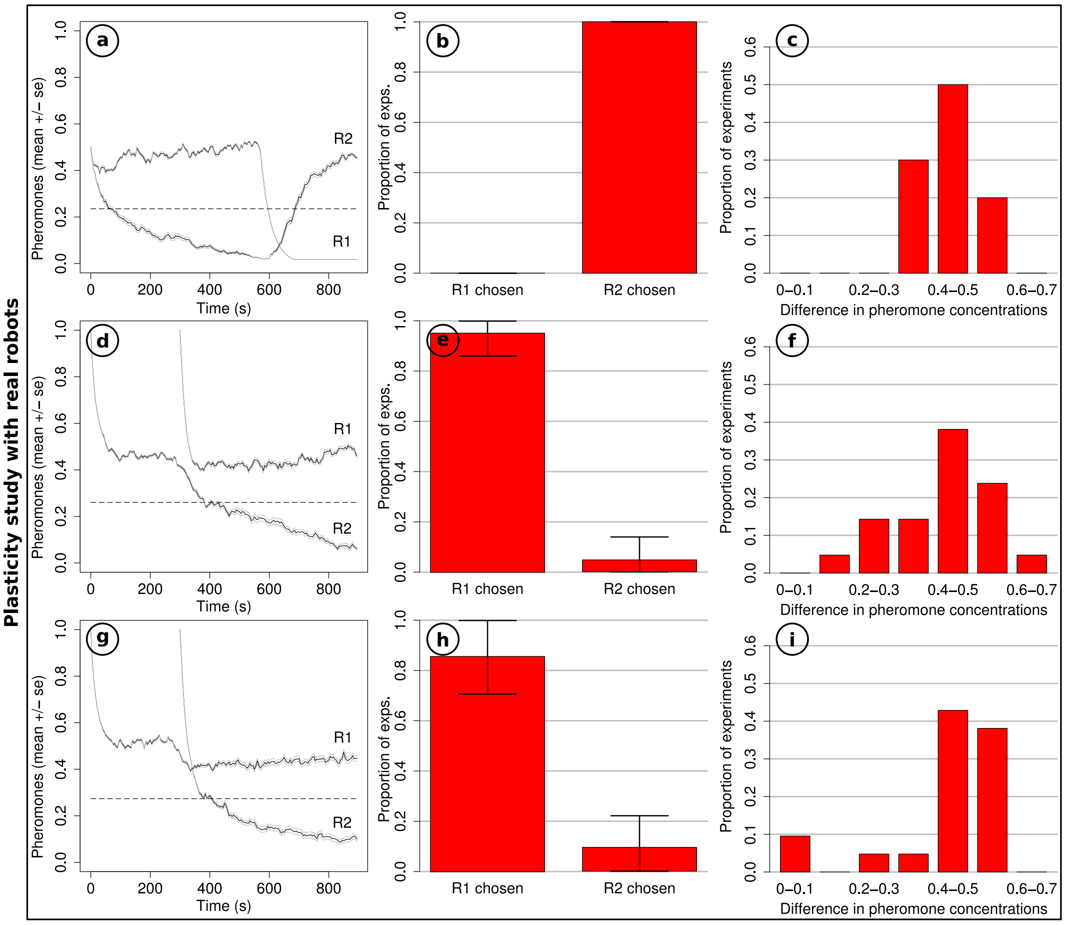


Figure 6.9: Plasticity of the swarm selection in experiments with real robots. Three different experiments are carried out using a setup with resources at different distances. First column: average pheromone concentrations over time on the paths. Second column: proportions of trials in which R1 or R2 are selected at the end of experiments. Third column: difference in pheromone concentrations ($|\beta_1 - \beta_2|$) at the end of experiments. **(a, b, c)** Initially R1 is selected. After 600 seconds, the path to R1 is blocked. The robots adapt their choice towards the remaining resource R2. **(d, e, f)** Initially R2 is selected as it is the only resource available. After 300 seconds, R1 becomes available. The robots detect the new and better resource R1 and select it. **(g, h, i)** Initially R1 is selected as it is the only resource available. After 300 seconds, R2 becomes available. The robots detect the new resource R2 but maintain their initial choice in favor of the better resource R1.

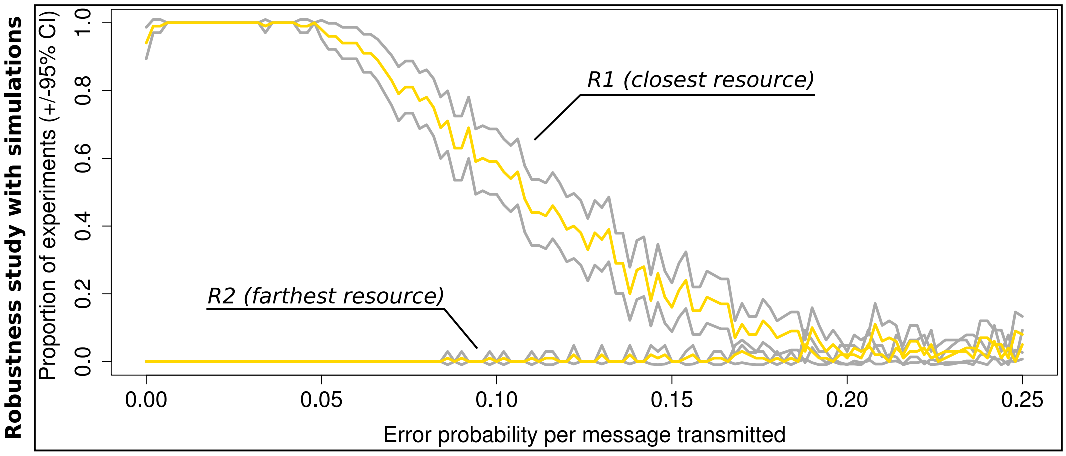


Figure 6.10: Robustness of the selection process in simulation experiments. We report the proportion of trials ($\pm 95\%$ confidence intervals) that end up with the selection of R1 or R2 as a function of the probability error per message transmitted. The performance of the robots degrades gracefully until the probability is greater than 0.188, in which case no path is selected.

In the first set of experiments, the selected path to the closest resource is suddenly blocked (see Figure 6.9a, b, and c): after 600 seconds, we introduce an obstacle along the selected path to the closest resource R1. After the obstacle is introduced the swarm reorganizes itself and selects the only path left. This is the outcome of two factors: first, the pheromone concentration in the shortest path drops quickly as virtual ants are blocked by the obstacle. Second, the longest path has maintained a minimum concentration of pheromone ϵ that allows it to have a low activity. As soon as the swarm has forgotten about the unavailable resource R1, it strongly amplifies its activity in the second path.

In the second set of experiments, we provide only one resource (R2) and later we introduce a new resource closer to the central place (R1) (see Figure 6.9d, e, and f). The resource R1 becomes available in the environment 300 seconds after the start of the experiment. The robot that detects it immediately starts to emit virtual ants towards the central place. During 60 seconds on average (until $t = 360$ s), a competition between the two paths takes place. The new resource is closer and the detecting robot starts with a higher concentration of pheromone (we have set it to 1) but on the rest of the path, the concentration of pheromone is still minimal. Hence, many virtual ants are emitted at R1 to the central place, but none are sent back to R1. It takes several virtual ants emitted at R1 to level up the pheromone concentration along the whole new path so that virtual ants start to travel from the central place to R1. At the same time, the path initially selected experiences a decrease in the pheromone concentration due to collisions of virtual ants emitted on the short path. As expected, the short path receives virtual ants faster than the long path, and consequently reacts faster. The difference in reaction time of the short path is amplified and the robots select the short path in $95\,\%$ of the trials.

The previous experiments are not sufficient to claim the plasticity of the system. Indeed, the robots could systematically select the path to a new resource appearing in the environment, even if the new resource is less profitable than the one currently selected. To validate the plasticity of our system, we run a new experiment in which we provide one resource (R1) and introduce a new and more distant resource (R2) after 300 seconds (see Figure 6.9g, h, and i). Again, during a period of 60 seconds on average (until $t = 360$ s), the paths to the two resources compete. The selected short path has a small drop in the pheromone concentration incurred by the sudden activity on the long path. However, the pheromone concentration on the long path never grows because of the inherent disadvantage of profitability. The robots maintain their initial choice in 90 % of the cases.

In summary, plasticity allows robots to modify their collective choice in favor of a path to a newly detected resource if and only if that resource is more profitable than the one already selected.

6.2.6 Robustness to communication errors

We test the robustness of our system to communication errors by introducing a probability that a message sent by a robot is corrupted. When corrupted, messages can be either lost or transformed into any other type of message with equal probability. In total, we use only three types of messages: virtual ants (VA_{CP} and VA_{R}), and wavefront signals used to build the topology of the network (SPT).

In this specific experiment, we rely exclusively on simulation and we use a setup with resources located at different distances. We also must adapt the parameters of the system to make pheromone deposition and evaporation slower. Indeed, corrupted messages reduce the number of virtual ants traveling along the paths. An immediate consequence is that pheromone concentration drops faster when there are communication errors. To cope with this effect, we set the amount of pheromone deposited per virtual ant to $l = 0.06$ and the rate of pheromone evaporation to $f = 0.01$. All other parameters are kept identical to the default configuration reported in Table 6.1.

The probability that a message gets corrupted is studied in the range $[0, 0.25]$. In Figure 6.10 we report for each error probability tested the proportion of trials that end up with the selection of the short path and of the long path. When there are no communication errors, the robots select the short path in 95 % of the cases. When the error probability is in the range $[0.002, 0.056]$, we observe an improvement in the capability of the robots to select the short path, in the best case always selecting the short path. This is because errors are more likely to happen on the long path. The performance of the system gracefully degrades as the probability of communication errors increases and the robots either select the short path or do not make any choice. In any case, the long path is never selected in more than $5\,\%$ of the trials. When the error probability is equal to 0.188, the robots are not able anymore to perform a collective choice in $92\,\%$ of the trials, and if they do a collective choice, it is indifferently in favor of the short or of the long path. Finally, when the communication error probability is greater than 0.188 the robots are not anymore able to select any path reliably.

6.3 Discussion and Conclusions

In this chapter, we described an approach to path selection inspired by the trail laying behavior of ants during a foraging task. We validated the mechanism with real world experiments and simulations involving a swarm of 20 robots. Furthermore, we studied properties of plasticity and robustness to communication errors.

Artificial pheromone allowed the robots to perform a collective choice between two identical resources available in the environment. Moreover, the robots' choice was modulated by the profitability of the resources. When offered two resources located at different distances, robots selected the path to the closest resource most of the times. If a sudden change occurred in the environment and a resource became unavailable, robots quickly reoriented their choice in favor of a path to another available resource. Finally, if a new resource appeared, robots changed their choice if and only if that resource was more profitable than the one already selected.

Interestingly, the collective decision of the swarm is the result of a self-organized process that relies on indirect information transfer (here we have virtual ants that communicate indirectly by depositing artificial pheromone). Although it is in general difficult to use stigmergy in robot swarms, we demonstrate a general solution which consists in simulating the process inside the robots. Using robots as containers of alterable information provides new perspectives to implement stigmergy in robot swarms.

In the following chapter, we investigate an experiment in which robots have no possibility to perceive the characteristics of the available options, and do not communicate with each other. Despite the ignorance of the members, we show that the group is able to discriminate resources and select one that fits the needs of the group.

Chapter 7

Resource discrimination

Robot swarms making collective decisions are often designed with the successful examples of ant colony optimization or particle swarm optimization in mind (Dorigo *et al.*, 1996; Dorigo and Stützle, 2004; Poli *et al.*, 2007). For instance, systems inspired by the pheromone laying behavior of ants are typically used to identify the shortest route in a network (Di Caro and Dorigo, 1998; Sugawara *et al.*, 2004; Garnier *et al.*, 2007). Bee inspired systems try to locate the best resource available in the environment (Pham *et al.*, 2006). When robots use this type of decision making mechanism, they are often bound to choose an option that minimizes or maximizes a certain characteristic. There are, however, situations that require robots to choose an option that meets adequate specifications.

As an example, if the robots are not alone in their environment, large and rich resources are more likely to attract competitors, adding an extra cost for the protection of the resource (Nagamitsu and Inoue, 1997; Holway and Case, 2001; Lichtenberg *et al.*, 2010). Moreover, robots may be forced to spread over a larger space to occupy the whole resource, hence impairing intra-group cooperation or reducing the benefits of working in group (Krause and Ruxton, 2002; Sumpter, 2010). In all these situations, it is more advantageous for groups to select resources that correspond closely to their needs, and to avoid oversized ones. But this task requires to evaluate the overall needs of the group in addition to the capacity of the available resources.

In this chapter, we propose and investigate a self-organized mechanism to discriminate which of several resources is the one that best fits a group's needs. In our experiment, robots can exploit resources directly on the spot. The robots can perceive whether they are exploiting a resource, but they are not endowed with sufficient cognitive abilities to evaluate the dimensions of the resources. Moreover, robots do not communicate, and they are only able to perceive the local presence of conspecifics. Strictly speaking, robots do not have sufficient knowledge to select the proper resource. Instead of gathering knowledge to make the decision, we rely on the interplay of two opposite feedbacks to drive the choice of the group towards the target resource.

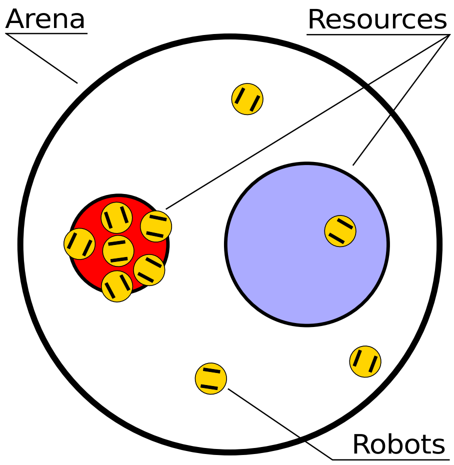


Figure 7.1: The experimental setup. In a circular arena of 1 m radius, the two resources are represented by cardstock discs (radius of 0.25 m and 0.45 m) fixed to the ground.

This chapter is organized as follows. In Section 7.1.1, we describe the task of the robots and the experimental setup we used. Section 7.1.2 details the control algorithm implemented on the robots, and how it takes inspiration from biological studies of cockroaches and ants. Next, we explain our methodology to parameterize our experiments in Section 7.1.3. Section 7.2 reports the results of our experiments with real robots, along with additional simulations results. Lastly, Section 7.3 concludes this chapter.

7.1 Methods

7.1.1 The task and the experimental setup

Our experiment is inspired from a biological study of the cockroaches' aggregation behavior (Amé *et al.*, 2004, 2006). We have adapted the experimental setup used with cockroaches to our needs. To this end, we notice that shelters can be seen as resources for cockroaches and that the shelters' surfaces correspond to the capacities of the resources. In addition, the total surface of the cockroaches' bodies corresponds to the group needs.

In our experiment, the task of the robots consists in selecting the resource that fits their needs. We place a group of 10 e-puck robots in a circular arena, searching for resources during one hour (see Figure 7.1). The target resource has the necessary and sufficient capacity to host all the robots involved in the task. The other resource is either larger or smaller. The robots must gather on the target resource.

In more details, the environment in which experiments take place is a circular arena of 1 m radius (see Figure 7.1). Since robots' perception relies on measures of infrared light, the arena is enclosed in a room without window to prevent natural light from entering the setup. Two compact 18 Watt fluorescent lamps placed 2 m above the arena shed light in the room. The role of resources is played by two dark cardstock discs fixed to the ground. One resource, called the target resource, has a carrying capacity that matches the number of robots (0.25 m radius). Its dimensions are obtained using simulations (see 7.1.3). The other resource is larger than the target by a factor of 1.8 (0.45 m radius). We recorded the experiments with a camera placed above the setup. Data was extracted from the videos using a tracking system designed at the IRIDIA laboratory that identified how many robots were at each resource. Additional results are obtained using simulations.

7.1.2 Robots' controller

Our starting point is the analytical model proposed by Amé *et al.* (2006) to explain the collective choice behavior of cockroaches when they select one shelter out of several identical ones. Amé *et al.*'s model is based on the assumption that the probability Q_i for a cockroach to leave a shelter i decreases with the density $D_i = X_i/S_i$ of individuals (X_i) in the shelter (of capacity S_i):

$$Q_i = \frac{\theta}{1 + \rho D_i^2},$$

where parameters θ and ρ determine the minimum and maximum probabilities of leaving the shelter depending on D_i. The model predicts that, when each shelter is sufficiently large to house all the cockroaches, the group will aggregate in only one of them. If shelters are too small the model predicts that the group will use two or more shelters equally.

As in the model, robots decide to stay at or leave a resource as a function of the density of robots already present. However, computing the density of a region requires knowledge of both the surface of the region and the number of individuals present in the region. This is a non-trivial task for a single individual, especially if the number of individuals and/or the surface to cover are important. Moreover, the robots have very limited capabilities. They can only perceive whether or not they are on a resource, and locally detect obstacles or other robots. They are not endowed with sufficient perceptual or cognitive abilities to measure the size of the resources nor to count the total number of robots.

We solve this problem by relying on the rate of encounters with other robots to evaluate the density of individuals at a resource: the more contacts they have with other robots, the greater their estimated value of the density. Interestingly, a recent study of emigrating ants *Temnothorax albipennis* (Pratt, 2005) reported that these insects

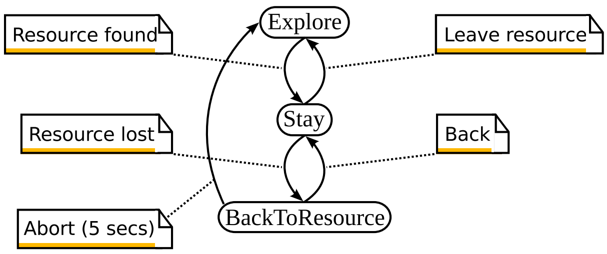


Figure 7.2: The robots' controller. Three different states are present in the FSM. Arcs denote transitions between the states.

also use the rate of encounters to evaluate their density in a cavity. Using this simple mechanism, it is thus possible to implement the collective selection process described previously.

The behavior of the robots is therefore a combination of the cockroaches' and of the ants' behavior (Pratt, 2005; Amé *et al.*, 2006). The robots' controllers are implemented as probabilistic finite state machines (Rabin, 1963). When a robot is not at a resource, it performs a random walk (Reif, 1965; Codling *et al.*, 2008) with obstacle avoidance till a resource is found again. When at a resource, the robot also performs a random walk, trying to remain there by turning around upon encountering borders. Every 30 s, the robot can decide with probability Q to leave the resource. The density D is approximated by the number of collisions with others robots measured during this time interval. The parameters θ and ρ are obtained using a genetic algorithm designed to favor a fast and stable collective choice of the target resource (see 7.1.3).

Robots are controlled by a finite state machine. The different states of the probabilistic finite state machine are represented in Figure 7.2, with arcs denoting the transitions between them.

In the following we summarize the possible behavioral states. The controller is initialized in the *Explore* state.

- **Explore.** The robot performs a random walk in the environment. An obstacle avoidance subroutine is triggered when needed. The robot switches to the *Stay* state when it encounters a resource.

- **Stay.** The robot performs a random walk inside the resource. Every 30 seconds, the robot decides with probability Q to leave the resource and enter the *Explore* state. If the robot finds itself outside the resource it switches to the *BackToResource* state.

- **BackToResource.** The robot has reached the border of the resource where it stays. It performs a U-turn then keeps turning on the spot. If the robot finds itself in the resource it switches to the *Stay* state. If 5 seconds have elapsed and the robot still does not perceive the resource, it switches to the *Explore* state.

7.1.3 Parameters tuning

The parameters θ and ρ determine when robots make the decision to leave the resources. For a robot, the probability Q to leave a resource is expressed as $\theta/\left(1+\rho D^2\right)$, where D is an estimate of the robot density in the neighborhood. If the resource is crowded, $D \approx 1$, and $Q \approx \theta/(1+\rho)$. If the resource is empty, $D \approx 0$, and $Q \approx \theta$. Therefore the parameters θ and ρ determine the maximum and minimum probabilities to leave a resource.

To ensure an effective collective behavior, we tune these parameters with a simple generational genetic algorithm (Goldberg, 1989). We define a genotype as a vector of two real values to be assigned to θ and ρ. We run the genetic algorithm for 1000 iterations, during which we breed new generations of 64 genotypes. The genetic algorithm loop consists in the evaluation, the selection and the reproduction of the genotypes.

To evaluate the fitness of a given genotype, we parameterize the controller of 10 simulated robots with the genotype. We run 50 simulated experiments with a target resource (0.25 m radius) and a larger resource (0.3 m radius). We also run 50 experiments with a target resource and a smaller resource (0.2 m radius). The fitness of the evaluated genotype is computed as an indicator of the ability of the robots to make a choice that is fast, lasting, and in favor of the target resource: $fitness = n \cdot (1-s)/T \cdot d/T \cdot c$, where n is the proportion of experiments in which a collective choice of the robots occured, s is the average starting time of the choices, d is the average duration of the choices, T is the total duration of an experiment and c is the proportion of choices made in favor of the target resource.

After evaluation, we rank the genotypes according to their fitness and create a new generation. The best 5% genotypes are cloned. Then, genotypes are picked randomly from the best 70% and mutated with a probability of 0.2 or reinitialized randomly with a probability of 0.07. A mutation consists of adding to the genotype random values drawn from Gaussian distributions. For θ, we use a Gaussian with $\mu = 0$ and $\sigma = 0.1$. For ρ, we use a Gaussian with $\mu = 0$ and $\sigma = 1.1$. During evolution, all vector component values are constrained to remain within the ranges $[0, 1]$ for θ, and $[0, 10^9]$ for ρ.

The analysis of the results reveals that the collective behavior of a group of 10 robots in our experiments is the most effective when $\theta = 1$ and $\rho = 600$.

7.1.4 Scaling of the experimental setup

Simulated experiments involve up to 100 robots. Because larger groups of robots span over a larger surface and need more resources, we have to scale the size of the arena and the size of the resources to match the group size considered. Also, since the robots

Group size	Target resource radius (m)	Arena radius (m)	Experiment duration (s)
10	0.250	1.000	3600.0
20	0.334	1.335	4804.6
30	0.385	1.541	5546.3
40	0.440	1.761	6340.0
50	0.476	1.905	6858.6
60	0.502	2.010	7235.3
70	0.519	2.078	7479.3
80	0.544	2.177	7837.6
90	0.570	2.279	8204.5
100	0.578	2.312	8323.1

Table 7.1: Parameters' values. Summary of the main parameters' values used in our experiments with respect to the group size considered.

do not move faster while the arena is larger, we have to scale the duration of the experiments. In order to make meaningful comparisons of the results across different group sizes, we ensure that for any size of the target resource a robot alone in the setup spends the same proportion of the experiment duration looking for the resources. To this end, we keep a constant ratio between the size of the target resource and the size of the arena (Blanco and Fournier, 2003), and that same ratio is also used to scale the duration of the experiments. Therefore, the whole scaling procedure depends solely on the size of the target resource. The size of the target resource is identified using simulations in which resources of various sizes are presented to the robots. The target resource is the one constantly preferred by the group. For each group size considered, Table 7.1 summarizes the target resource size, the arena size and the experiments duration that we used to parameterize our experiments.

7.2 Results

7.2.1 Collective discrimination

In a first set of experiments, we assess the robots' capability to discriminate between two different resources. The robots are offered a target resource that provides enough space for the group, while the area of the other one is 1.8 times larger. At the end of an experiment, we consider as chosen the resource occupied by the largest number of individuals. To find out if robots can distinguish between two resources, we record the resources chosen in a serie of 35 repeated experiments. We then use a binomial test to

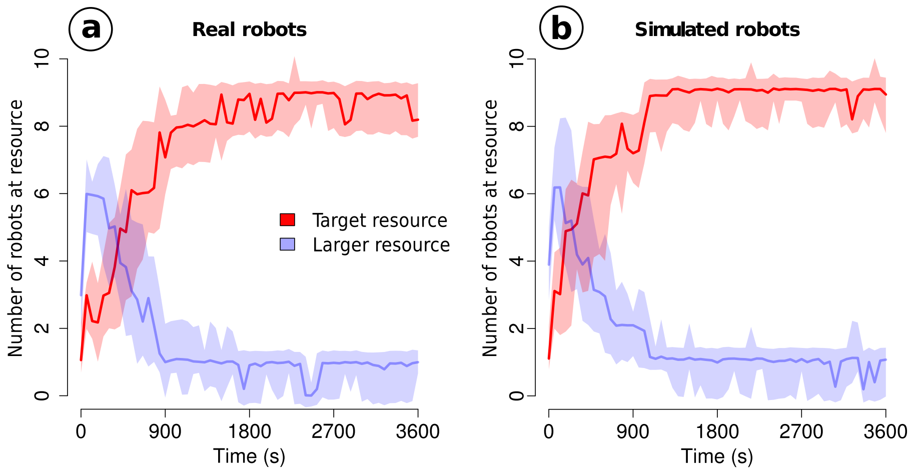


Figure 7.3: Collective discrimination between two different resources. We use a target resource of ideal dimensions and a larger resource. The figures show the number of robots (median $\pm$ 95% CI) at each resource as a function of time in reality and in simulations respectively (35 trials). Initially robots find the larger resource more easily, and then their collective choice changes quickly in favor of the target resource.

assess if the collective choice is significantly different from random, that is, if robots are more likely to choose one of the two resources.

The average number of robots found at each resource is reported in Figure 7.3. At the beginning of the experiments, robots are mostly located at the large resource, but they quickly leave it to gather at the target resource. After one hour, robots have collectively selected the target resource in all the trials. Robots exploring the environment were more likely to first encounter the large resource by chance, due to its size. However, the low density of robots at this resource prevented them from remaining in this location. On the contrary, the target resource, once discovered by the robots, provided them higher densities and therefore longer staying times. Finally, robots were able to discriminate between resources of different sizes, choosing the one that best fits the group size.

7.2.2 Accuracy and scalability

The second set of experiments sheds light on the discriminatory power of the group. In order to allow a large number of replications, we rely here on simulations that were validated against the first set of experiments (see Figure 7.3b). We first introduce a target resource in the environment. In successive tests, we add resources of growing size and observe which one is chosen by the simulated robots. The size of the presented resources varies from 0.2 to 2 times the size of the target resource. With 10 robots, we observe that the group successfully recognizes the target resource when the other

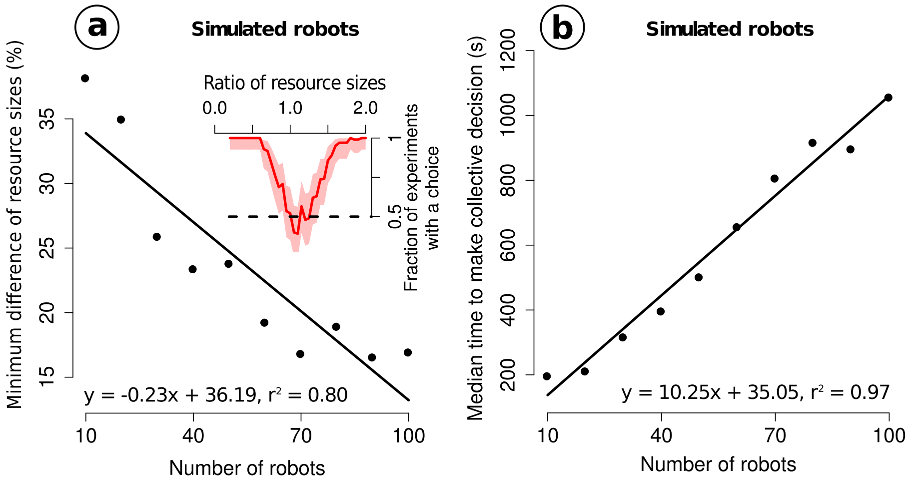


Figure 7.4: Scalability study of the collective discrimination behavior. **(a)** Median time to make the collective decision in function of the group size. We report the first time when the target resource contains most robots. The time grows linearly with the number of robots involved in the task ($r^2 = 0.97$, t-test $p < 0.001$). **(b)** Discriminatory power of the collective behavior as a function of the group size. The inset shows the choice of the target resource by a group of 10 simulated robots when the other resource has different sizes. When the two resources have similar dimensions, the robots are not able to discriminate them properly, and we observe a random choice of the resource. The main plot shows the minimum resource size difference necessary to observe discrimination. This minimum difference decreases with the group size (100 trials). A linear regression performed on the data indicates a significant improvement of the discriminatory power ($r^2 = 0.80$, t-test $p < 0.001$).

resource is smaller or larger by a factor of 0.91 or 1.38 (see inset of Figure 7.4a). When the resources do not differ enough, the robots are not able to discriminate them anymore. They are instead making random choices (binomial test on the number of choices in favor of the target resource with respect to the total number of trials). We measure the difference of resource sizes needed to observe selection of the target resource for a growing number of robots. As the number of robots grows, we see a rapid increase in the discriminatory power (see Figure 7.4a). With 10 robots, a minimum difference of 38% is necessary to observe selection of the target resource, while 100 robots only require a difference of 17%: larger groups of robots discriminate between resources more accurately.

We also report the median time needed by robots to make their collective decision in Figure 7.4b. To calculate when the group has chosen the target resource, we rely on the average dynamic of the collective choice, that is, the average number of robots at each resource during the experiments. For all group sizes and on average, we observed that robots gather at the large resource and then switch to the target resource. We define the decision time as the moment when there are more robots (on average) at the target resource than at the large resource. The decision time grows linearly with the number of robots involved in the task ($r^2 = 0.97$).

7.2.3 Adaptivity

The third set of experiments shows the adaptivity of the robots' collective choice when a better opportunity appears in the environment. We first perform experiments with 10 robots and then explore the impact of increasing group size with simulations. Experiments start with a single resource in the environment, which is 1.8 times bigger than the target resource. As this is the only option available, robots aggregate at this resource (see Figure 7.5a-b). After five minutes, we add a target resource inside the arena. With 10 robots, the group adapts its choice to the new settings and selects the target resource. It takes on average 720 s to observe this adaptation, which we continue to observe in simulation with larger groups of robots. This is shown in Figure 7.6, where we report the median time of adaptation with respect to the number of simulated robots. With 100 robots, adaptation occures after 3.13×10^6 s. Adaptation time grows exponentially with the number of robots involved ($r^2 = 0.96$).

7.3 Discussion and Conclusions

Our results illustrate how simple interactions can lead a group to collectively discriminate between resources one that closely matches its needs. The collective discrimination arises from the interplay of several factors. On the one hand, individuals prefer to stay

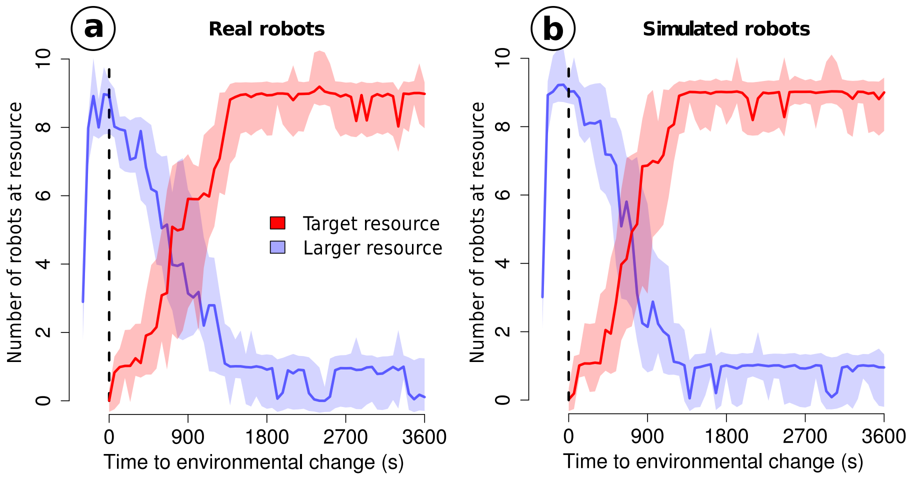


Figure 7.5: Adaptivity of the collective choice. Ten robots are presented a single large resource and gather in it. After five minutes (dashed line) a target resource is introduced in the environment. Figures show the number of robots (median $\pm$ 95% CI) at each resource as a function of time in reality and in simulations respectively (35 trials). We observe a quick adaptation of the collective choice towards the target resource after its introduction.

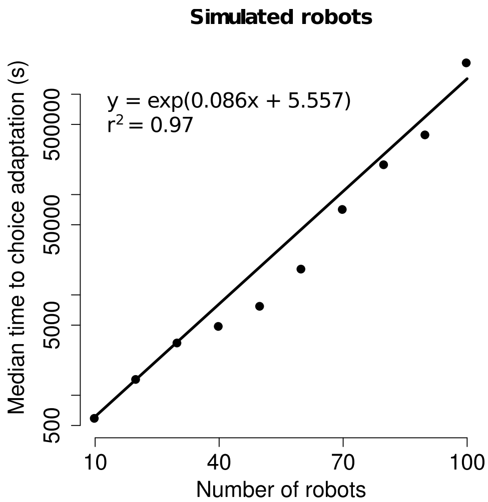


Figure 7.6: Adaptivity of the collective choice when group size increases. The median adaptation time (plotted in log scale) grows exponentially with the number of robots involved (100 trials). We considered that the choice was reverted as soon as there were more robots at the target resource than at the large resource.

at resources where their density is higher. This positive feedback strongly favors the selection of smaller resources where higher densities can be achieved. On the other hand, it is more difficult to join a resource where the density is high. This negative feedback favors the selection of larger resources that can host additional individuals. Moreover, smaller resources are less likely to be discovered and if one is selected the group may be forced to split (Amé *et al.*, 2006). Because of this, competition may happen with any other resource found by these excluded individuals. In our model, these factors balance each other out in one case: when the capacity of a resource matches the size of the group.

We found that the accuracy of the discrimination increases with the group size: larger groups are able to detect proportionally lower differences between two resources. We also found that the time required to make this collective decision grows linearly with the group size. In our experiments, we had to scale the size of the setup while increasing the number of robots, which implies that robots must travel longer distances to move from one resource to the other. Had these experiments been carried out in a fixed size environment, it is possible that the decision time might have decreased with group size. In addition, we observed that the collective choice is flexible. If the group has selected a resource, it is able to switch for a better one introduced afterwards.

Collective discrimination is achieved by agents with very limited perceptual and cognitive abilities, as demonstrated by our robotics implementation. In our experiments, robots can only detect when they are at a resource site, and they are neither able to measure the capacities of the different resources nor to evaluate the number of robots using them. Moreover, the robots do not communicate any information explicitly, and solely rely on the detection of nearby robots to make decisions. As a consequence, the collective discrimination process is fully distributed.

However, the collective behavior presented in this chapter has one noticeable limitation: it is not possible to have the robots select another resource than the one matching their needs. The following chapter addresses this problem and shows how to specify to the robots the size of the resource they should be looking for.

Chapter 8

Controlling resource discrimination

In the previous chapter, we provided a first example of collective discrimination obtained by combining specific behaviors of ants and cockroaches. Collective discrimination was demonstrated using a group of robots to which resources of different sizes were offered. Single robots could detect resources but they could not perceive their characteristics, and in particular they were not able to evaluate their size. Moreover, robots did not explicitly communicate; they only detected the presence of nearby robots. The group of robots could overcome the limited capabilities of the individuals and was able to select the resource corresponding to the needs of the group, that is, the resource just large enough to receive the whole group. However, robots could only select a resource of a specific carrying capacity, matching the size of the group. A further step, which would bring collective discrimination closer to possible real world applications, would consist in the modification of the mechanism so that the collective choice of the robots can be tuned.

In this chapter, we extend the collective discrimination behavior to allow a group of robots to discriminate a variety of resources based on their size by controlling the behavioral parameters of the robots. The main modification brought to our mechanism lies in the tuning of the obstacle avoidance distance, which defines the minimum distance at which a nearby robot can be detected. By controlling this parameter, we define the surface spanned by the swarm. We show how the collective decision of the group can be modulated so as to choose a specific option in a continuum of possibilities. By parameterizing the robots appropriately, it is possible to have the robots search and locate a specific resource in the environment. This is achieved with robots that do not explicitly communicate with each other and cannot estimate directly the characteristics of the resources available.

In the following, we detail our experimental setup (Section 8.1.1) and the adaptation we brought to the collective discrimination behavior (Section 8.1.2). We also explain in Section 8.1.3 how the behavioral parameters of the robots were chosen, depending on the situation tested. Results are presented in Section 8.2. We describe the ability of the robots to focus on any particular resource, provided that control parameters are

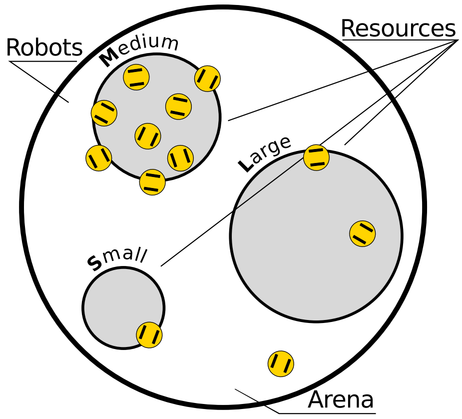


Figure 8.1: Experimental setup. Three different resources are used for the experiments: small, medium and large resource.

properly adjusted. In addition, we show how the collective choice of the robots can be controlled online; in this last experiment, the swarm must search and locate resources that are requested in turn. Finally, we provide a short discussion and concluding remarks in Section 8.3.

8.1 Methods

8.1.1 The task and the experimental setup

We place a group of robots inside an environment in which three resources of different sizes are available. The experiments aim at showing that robots can choose a resource of specified size as a function of their behavioral parameters, and that the collective decision of the robots (*i.e.*, at which resource they gather) can be controlled by issuing a single command that contains the appropriate behavioral parameters.

The experiments are run only in simulation and take place in a circular arena of $R_A = 2.5\,\text{m}$ radius (see Figure 8.1). The role of resources is played by three black discs on the ground; they are named the small, large, and medium resources. We denote the radiuses of this resources R_S, R_M, and R_L respectively. The exact size of the resources is not critical in these experiments because the robots are specifically parameterized to find them. However, we make sure the resources are sufficiently different in size in order to facilitate their discrimination. We run experiments with 10 simulated robots. All the experiments are repeated 100 times.

The ground sensors of the robots are simulated without noise to avoid that a robot leaves a resource by mistake in the case it can not perceive it reliably. Indeed, in earlier tests, we found that the probability to spontaneously leave a resource is a behavioral

Table 8.1: Summary of variables and parameters of the experiments, along with their default values.

Parameters	Description	Default Value
N	number of robots	10
T	duration of an experiment	5 hours
R_S	radius of the disc corresponding to the small resource	0.25 m
R_M	radius of the disc corresponding to the medium resource	0.50 m
R_L	radius of the disc corresponding to the large resource	0.75 m
R_A	radius of the arena	2.50 m
θ	parameterizes the probability to leave Q	0.76
ρ	parameterizes the probability to leave Q	120.00
τ	decision delay	30.00 s
δ	detection distance	0.16 m
ϵ	probability to leave sponteanously	3.29×10^{-3}
$\overline{\omega}$	random walk mean free path at resource	0.51 m
ω	random walk mean free path in the environment	0.72 m

parameter useful to achieve better control over the collective decision. Therefore, we prefer to define the probability of spontaneous exit as a tunable parameter rather than a static property of the robot.

8.1.2 Robots' controller

The behavior of a single robot mainly consists in trying to find a resource and staying there. However, the robot may leave the resource with some probability if it does not perceive enough other robots in the vicinity. When the robot leaves a resource, it simply explores randomly its environment until it finds a resource again, which also means it can return by chance to the same resource it just left.

This simple behavior is based on the aggregation behavior of cockroaches (Amé *et al.*, 2004, 2006) and the nesting behavior of ants (Pratt, 2005). More precisely, the aforementioned behavior is produced on the robots with a software controller implemented as a probabilistic finite state machine (Rabin, 1963). This means that robots move from one state to another as a function of their environment's perception and the decision to switch state is not necessarily deterministic but can be made with some probability. In particular, robots are constantly using isotropic random walks to move around. In this kind of random walk, the motion of robots is a sequence of straight runs in random directions. Before choosing a new random direction of movement, robots cover on average a distance ω which is called the mean free path (for more details, see Codling *et al.*, 2008).

The probabilistic finite state machine described here follows a logic very similar to the state machine presented in the previous chapter (see Figure 7.2). In the following, we summarize the three states implemented:

- **Explore**: The robot performs a random walk in the environment with mean free path ω. An obstacle avoidance subroutine is triggered when the arena border or another robot is detected at less than δ m. The robot switches to the *Stay* state when it encounters a resource.

- **Stay**: The robot performs a random walk in the disc that symbolizes the resource with mean free path ϖ. Every τ seconds, the robot decides with probability Q to leave the resource and switch to the *Explore* state. If the robot finds itself outside the disc that symbolizes the resource, it decides with probability ϵ to leave the resource and switch to the *Explore* state. Otherwise, the robot tries to stay at the resource and switches to the *BackToResource* state.

- **BackToResource**: The robot performs a 180° turn, then it keeps turning on the spot. If the robot finds itself at the resource it switches to the *Stay* state. If after five seconds the robot still does not perceive the resource, it switches to the *Explore* state.

As in the previous chapter, the probability Q for a robot to leave a resource depends on the density D of robots at the resource approximated by the collision rate:

$$Q = \frac{\theta}{1 + \rho D^2}. \tag{8.1}$$

Table 8.1 summarizes all the parameters used in our experiments, including the default behavioral parameters of the robots.

8.1.3 Parameters tuning and setup scaling

As in the previous chapter, we rely on a genetic algorithm to tune the main behavioral parameters of the robots. Because the experiment covered in this chapter is very similar to the one of the previous chapter, the whole methodology of parameter tuning remains the same. Thus, we do not repeat the full description of the algorithm, and we only mention the changes. We define a genotype as a vector of five real numbers to be assigned to θ, ρ, ϵ, ω, and ϖ. The ranges of possible values for each parameter are: $\theta \in [0, 1]$, $\rho \in [0, 2000]$, $\epsilon \in [0, 0.01]$, $\omega \in [0, 64.40]$, and $\varpi \in [0, 64.40]$. We run the genetic algorithm for a total of 300 iterations.

To evaluate the fitness of a given genotype, we parameterize the controller of 10 simulated robots with the values of θ, ρ, ϵ, ω, and ϖ encoded in the genotype. We run $R = 15$ simulated experiments with a small, a medium and a large resource. Experiments last $T = 10800$ seconds. In Equation 8.2, the fitness F of the evaluated genotype g is computed as an indicator of the robots ability to make a choice that is fast, lasting,

Table 8.2: Behavioral parameter values obtained by the genetic algorithm so as to select the medium resource when the whole setup is scaled by a factor k (corresponding results are in Fig. 8.2).

k	R_S	R_M	R_L	R_A	θ	ρ	τ	δ	ϵ	ϖ	ω
1	0.25	0.5	0.75	2.5	0.76	120	30	0.158	0.0033	0.52	0.77
2	0.50	1.0	1.50	5.0	0.79	211	60	0.316	0.0031	4.77	1.67
3	0.75	1.5	2.25	7.5	0.81	303	90	0.474	0.0029	9.02	2.71
4	1.00	2.0	3.00	10.0	0.84	394	120	0.632	0.0026	13.27	3.61
5	1.25	2.5	3.75	12.5	0.86	486	150	0.790	0.0024	17.39	4.64
6	1.50	3.0	4.50	15.0	0.89	578	180	0.948	0.0022	21.64	5.67
7	1.75	3.5	5.25	17.5	0.91	669	210	1.106	0.0020	25.89	6.57
8	2.00	4.0	6.00	20.0	0.94	761	240	1.264	0.0018	30.14	7.60
9	2.25	4.5	6.75	22.5	0.96	853	270	1.422	0.0015	34.39	8.50
10	2.50	5.0	7.50	25.0	0.99	944	300	1.580	0.0013	38.64	9.53

and in favor of the resource A and not of the other resources (B and C). Depending on which resource the robots had to select, A was set to be the small, medium or large resource. We define the fitness F as follows:

$$F = \frac{1}{NR} \sum_{i=0}^{R} \Bigg(\max\left(A_i - B_i, 0\right) + \max\left(A_i - C_i, 0\right) \Bigg), \tag{8.2}$$

where A_i, B_i, and C_i are the number of robots at each resources at the end of the replication i. In a nutshell, the fitness function is designed to maximize the difference of robots present at resources A and B, A and C at the end of the experiments.

All the parameters values of θ, ρ, ϵ, ω, and ϖ reported in the experiments are obtained with this genetic algorithm. Moreover, the avoidance distance δ is calculated so that the group of N robots is able to "fill" the resource to be selected. Denoting R the radius of the disc corresponding to this resource, the size of the resource is $2\pi R^2$. We set δ so that each robot uses the same proportion of the resource size $2\pi R^2/N$. Thus, we have $\delta = \sqrt{\frac{1}{2\pi} 2\pi R^2/N} = R/\sqrt{N}$. The decision delay τ is set to 30 s for an arena of 2.5 m of radius. The decision delay $\tau = 30R_A/2.5$ is scaled proportionally to the radius of the arena so as to allow robots operating in larger arenas to wait longer for other robots at the resources.

8.2 Results

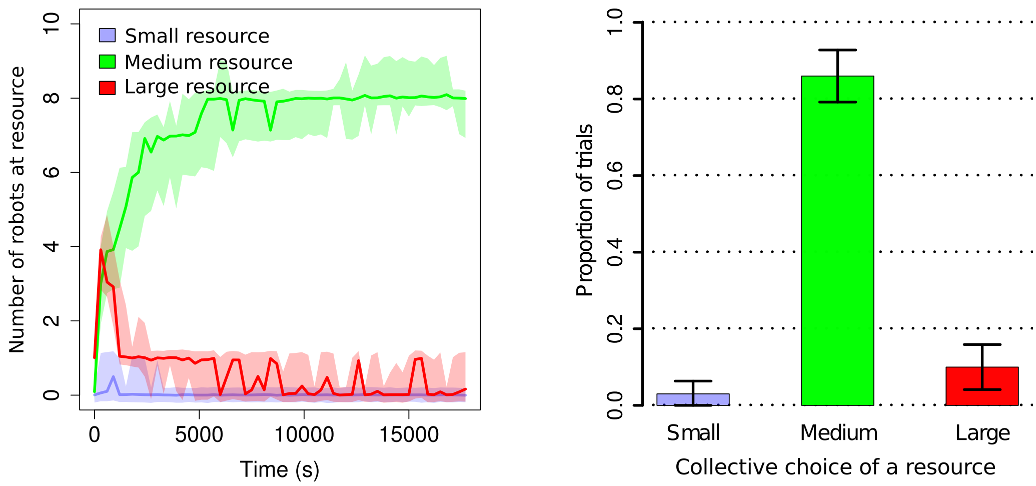
8.2.1 Discrimination of a resource of fixed size

In this experiment, we assess the ability of the robots to select a medium size resource when offered three different resources: small (S), medium (M), and large (L) (see Figure 8.1). The main difference of this experiment with respect to the previous chapter lies in the size of the resource that robots select. Thanks to the modification brought in the behavior of the robots, the swarm is not anymore bound to select a resource that has the necessary and sufficient capacity to cover the needs of all the robots (in this work it is the small resource). Instead the robots are able to select the medium resource, symbolized by a disc which has a radius twice larger than the one of the small resource.

Figure 8.2a shows the median number of robots at each resource during an experiment that lasts 5 hours. We used the default values of the behavioral parameters reported in Table 8.1. At the beginning of the experiment, the large resource receives more robots as they are more likely to find it by chance, when performing their exploratory random walk. However, after 1500 seconds, robots leave the large resource and start to occupy the medium resource. From this moment the number of robots in the medium resource grows until it stabilizes to 8 robots. At the end of the experiments, not all of the robots are at the medium resource because they always have a small probability to leave resources and explore the environment. These exploring robots do not stay at the non selected resources but rather come and go after brief periods of time. Figure 8.2b shows that the robots select the medium resource in most of the trials ($86\,\%$), while the small and large resources are selected only in $3\,\%$ and $10\,\%$ of the trials, respectively.

8.2.2 Discrimination of resources characterized by various sizes

In the previous section we have shown that a swarm can select a medium resource of $0.5\,\mathrm{m}$ of radius. We perform additional experiments in which we test if selection is still effective with a large range of radiuses for the medium resource (from 0.5 to $5\,\mathrm{m}$ in steps of $0.5\,\mathrm{m}$). Let k be the ratio by which the medium resource is enlarged. We also scale the large and small resources by the factor k to maintain surface ratios of 0.5 and 1.5 among the different resources. The size of the arena is also scaled by k, to have all the resources fit inside the environment. As a consequence, robots need more time to explore the arena. We have therefore also multiplied the duration of the experiments by the factor k. This scaling leads us to study simulations that have a rather limited interest for experiments with real robots such as e-pucks, because the radius of the arena can be 25 m long, and e-pucks need more than 6 minutes to go accross such a large arena. The study is nevertheless useful to check if the collective decision in favour



(a) (b)

Figure 8.2: Selection of the medium resource, typically much larger than required to host the whole swarm. **(a)** Number of robots at each resource during an experiment of 5 hours (median ±95 % confidence interval, 100 trials). At the beginning of the experiment, robots find resources by chance and the large resource is therefore the one with most robots. After 1500 s, robots leave the large resource and focus on the medium resource. At the end of the experiments, the number of robots at the medium resource is stabilized to 8 robots. **(b)** Distribution of the collective decisions of the robots at the end of the experiments. We consider that the chosen resource is the one containing the majority of robots. The medium resource is selected in 86 % of the trials, which is significantly more than the other resources (multinomial test, $p < 0.001$).

Table 8.3: Behavioral parameter values obtained by the genetic algorithm to select the one of the three resource available in the environment (corresponding results are in Fig. 8.4 and 8.5).

$Resource$	R_S	R_M	R_L	R_A	θ	ρ	τ	δ	ϵ	$\overline{\omega}$	ω
S	0.25	0.5	0.75	2.5	0.76	120	30	0.079	0.0008	1	5.58
M	0.25	0.5	0.75	2.5	0.76	120	30	0.158	0.0033	4	5.58
L	0.25	0.5	0.75	2.5	0.76	120	30	0.237	0.0074	50	5.58

of the medium resource holds for various setup sizes, while a physical implementation with these dimensions would probably require to use another type of robot.

The behavior of the robots needs similar adjustments to perform well. For instance, the probability to leave a resource has to be lowered in larger environments, so that robots wait longer for their partners which have to travel longer distances. As mentioned in Section 8.1.3, we rely on the genetic algorithm to determine the appropriate parameters that allow robots to select a resource of a given size in a given arena. The values of these parameters are reported in Table 8.2.

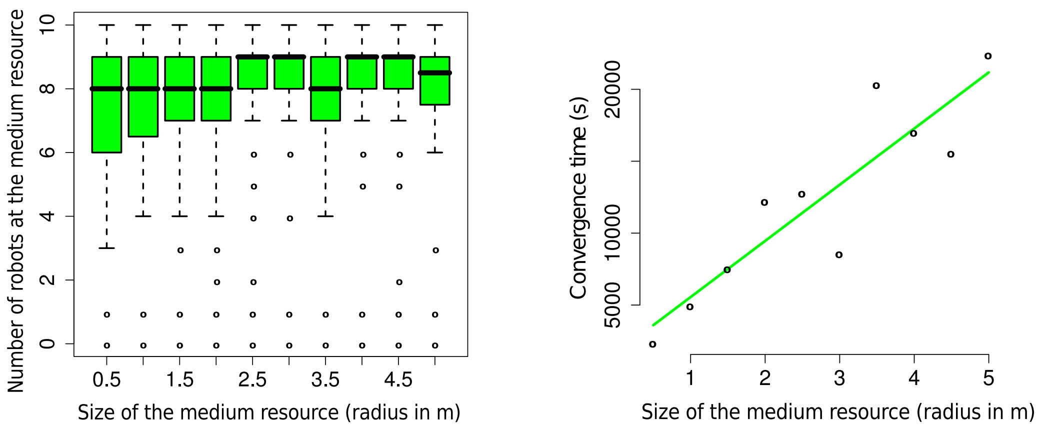
When properly parameterized, the swarm of robots selects the medium resource with high accuracy, whatever the size tested. In Figure 8.3a, we use boxplots to report the number of robots present at the medium resource at the end of the experiments. For diameters ranging from 0.5 m to 5 m, there are on average 8 robots. For any resource size tested, the medium resource is selected in at least 83 % of the trials. The collective decision of the robots is always significantly in favor of the medium resource (multinomial test, $p < 0.001$). Figure 8.3b shows the average time necessary to gather more than 75 % of the robots at the medium resource. With a bigger arena, robots need proportionally more time to reach the same level of selection ($r^2 = 0.78$, t-test $p < 0.001$).

8.2.3 Controlling the swarm collective decision

The last experiment illustrates the control of the collective decision of the robots, that is, how to tell the robots to look for a specific resource in the environment by sending a single command.

We put 10 robots in the default environment with the 3 resources. Every 5 hours, we send them a new command that indicates which resource should be selected. The robots are preprogrammed with 3 different parameter sets, each corresponding to the selection of one resource. These parameters are identified beforehand using the genetic algorithm (see Table 8.3). When a robot receives a command (S, M, or L), it loads the corresponding set of parameters and its behavior is consequently modified.

To observe the capability of the robots to execute the command and discriminate a desired resource at any time, we have devised a specific sequence of commands. Robots should be able to discriminate and select a resource even if their current collective de-



(a) (b)

Figure 8.3: Selection of the medium resource in function of its size (all the setup is proportionally scaled: radius of the arena, the small, medium and large resources). **(a)** Boxplots of the number of robots found at the medium resource at the end of the experiments (100 trials) in function of the size of the resource. Each box comprises observations ranging from the first to the third quartile. The median is indicated by a horizontal bar, dividing the box into the upper and lower part. The whiskers extend to the farthest data points that are within 1.5 times the interquartile range. Outliers are shown as circles. We observe that the robots gather at the medium resource in at least $83\,\%$ of the trials for any resource size tested. **(b)** Average time necessary to gather at least $75\,\%$ of the swarm at the medium resource. The measured time grows linearly with the size of the resource ($r^2 = 0.78$) which also means that swarms need the same proportion of experimental time to perform their collective selection, given that the duration is also scaled linearly with the size of the resources.

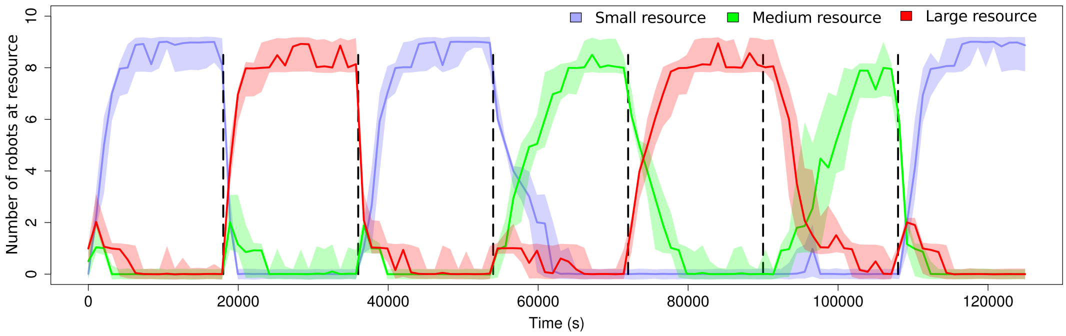


Figure 8.4: Selection sequence of the different resources in the arena. The sequence ordered is S-L-S-M-L-M-S where all the six possible transitions are represented. After a command is issued, the robots need a maximum of 8000 s to select the requested resource.

cision bears already on a different resource. The sequence of commands ensures that robots will go through all the six possible transitions between the three resources: S to M, S to L, M to S, M to L, L to S, and L to M. The chosen sequence is S-L-S-M-L-M-S. This implies that the whole experiment lasts 7 times 5 hours, that is 35 hours.

In Figure 8.4, we report the median number of robots at each resource during the experiment. Every 5 hours a new command is issued and we observe that the robots are modifying their collective decision accordingly. Robots are capable of moving from any resource to any other one upon request, which clearly shows there is no memory effect preventing the adaptation of the collective decision. In all transitions, after maximum 8000 s the robots reach the steady-state at the commanded resource. In Figure 8.5 we show separately the six possible resource transitions. The separate studies are in agreement with the longer experiment.

8.3 Discussion and Conclusions

In this chapter, we achieved the control of the decision made by a robot swarm performing collective discrimination. We did so by carefully tuning the behavioral parameters of the robots, in particular the obstacle avoidance distance. As in the previous chapter, robots were not endowed with complex communication or perception means. They could only perceive whether they were at a resource or not and they could detect the presence of nearby robots.

We demonstrated that the robots are able to discriminate among the different resources available in the environment and to specifically select one of them. At any time, provided that robots could stabilize their collective decision, they could successfully comply with a given command and choose a newly designated resource. We observed

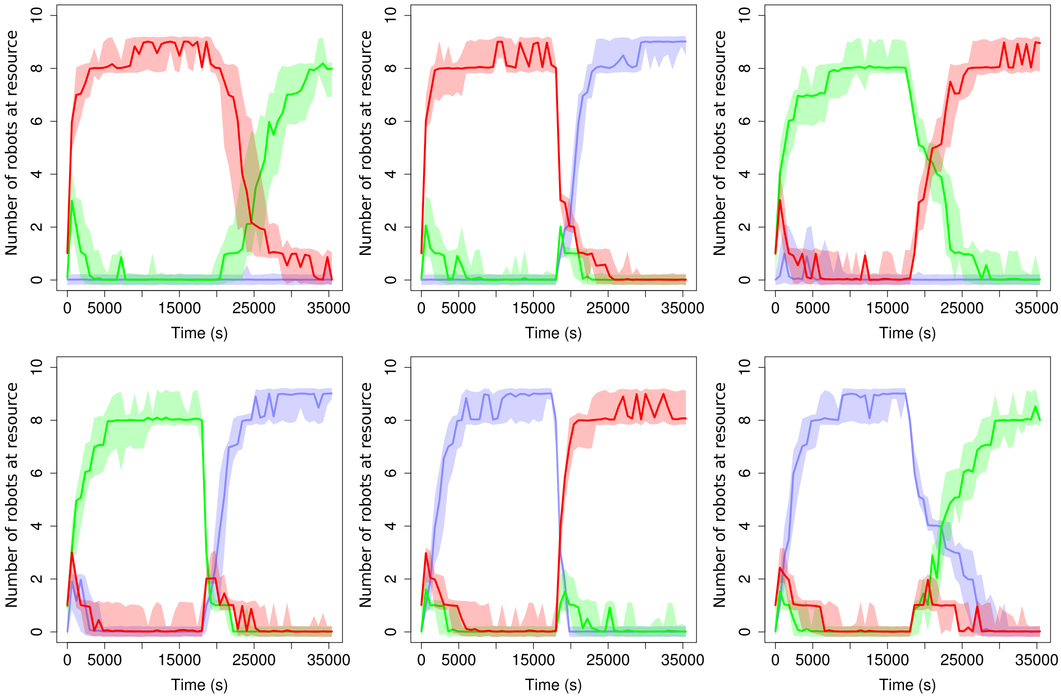


Figure 8.5: Selection of the different transitions in separate experiments in order to study the memory effect. We observe results are in agreement with the continuous experiment.

that this adaptation occurred effectively for any transition requested, and was always achieved after the same amount of time.

Chapter 9

Conclusions

Creating robot swarms that make effective collective decisions requires a careful design of the way robots behave and interact. In this thesis, we relied on imitation and amplification mechanisms to trigger self-organized decisions in the groups of robots. We investigated swarm decision making in a series of four different foraging problems. Through these four studies, we devised different implementations to achieve decision making in groups of robots. Moreover, these studies also constitued a contribution to the state of the art in robot foraging. In particular, we demonstrated the use of direct and indirect information transfers to implement interactions between robots.

9.1 Summary of contributions

Our first contribution was to detail the background and context to the research presented in this thesis. We introduced the concept of self-organization and summarized interesting properties that can be displayed by self-organized systems. We traced the historical landmarks that led to the birth of swarm robotics, and presented the design problem, which is fundamental to this field. We underlined that the design problem translates to understanding what interactions and behaviors bring the robots to make self-organized decisions. We described what kind of interactions can take place in a swarm and we described the mechanisms of imitation and amplification at the core of collective decision making processes. Finally, we presented the foraging task which we used to study various decision making problems. In addition, we reviewed the literature specific to the four different tasks investigated.

Our second contribution consisted of the presentation of the tools that we used to perform our experiments. We provided a description of the two robotic platforms employed, the s-bot and the e-puck. We also introduced two original communication systems that we designed, namely the visual range and bearing and the infrared range and bearing communication systems. We also presented our simulation platform.

Our third contribution was the study of a task of cooperative transport in which robots had statistically identical knowledge about the direction in which they should pull a heavy object. We proposed a negotiation mechanism that allows robots to exchange their opinions and agree about the goal direction. We demonstrated that this mechanism leads robots to coordinate their efforts and successfully transport the heavy object. Furthermore, we have shown that individual knowledge about the goal direction improves when using negotiation, by taking advantage of redundant information available in the group of robots.

The fourth contribution concerned the study of a task of collective localization. In a central place foraging experiment, robots had to keep track of the locations of the resources and of the central place by means of odometry. Because of odometry errors, robots could easily get lost in the environment. We proposed a knowledge sharing mechanism called social odometry which allows robots to improve their estimates of the foraging locations. In this experiment, the robots that had traveled less had better information about the foraging locations. We showed that our mechanism is capable of exploiting this heterogeneity to propagate better information in the group using an indicator of confidence for each robot. Finally, we have shown that social odometry induces a collective decision in favor of the closest resource because robots exploiting that resource travel less, have a higher confidence level in their personal information, and are therefore more likely to be imitated.

The fifth contribution was a new method to use indirect information transfers in robot swarms. Because of the difficulty for robots to modify their environment, we used the robots themselves as interactive nodes that can hold stigmergic information. We demonstrated this idea in a resource selection experiment. Robots established chains marking out paths from the central place to the resources. We simulated virtual ants traveling along the robot chains and depositing artificial pheromone inside the robots. While robots did not have explicit knowledge about the profitability of the resources, they could select the most profitable resource. Moreover, we showed the adaptivity of the system when a resource suddenly becomes unavailable, and its robustness to communication errors.

Finally, our last contribution consisted of a new mechanism for collective discrimination inspired by the aggregation behavior of cockroaches and the nesting behavior of ants. In our experiment, robots were offered several resources and had to discriminate and select a resource matching their needs, *i.e.* a resource neither too small nor too large. Robots did not have any information about the characteristics of the resources. Robots could merely detect if they were at a resource, and they could detect the presence of nearby robots. We demonstrated that our mechanism produces resource discrimination and we showed properties of scalability and adaptivity. Additionally, we showed how to control the collective behavior of the robots so that they could be requested to search for a particular resource in their environment.

9.2 Imitation as a source of collective decision

We showed that collective decisions in robot swarms can be obtained using a simple imitation mechanism. We discuss here the different implementations of imitation as a function of the information available to the robots. Moreover, we examine below the three main aspects at the origin of the different flavors of the imitation mechanism we studied. We also try to provide some insights on the key properties of this mechanism, including scalability, robustness and adaptivity, speed and accuracy.

9.2.1 Available information guides the design of imitation

The robots make their collective decision based on an input of information about the alternatives between which they must choose. We distinguish two ways this information can be transfered: directly or indirectly.

In the case of direct information available, robots can measure one or more characteristics of the available alternatives. When **all robots can perceive** accurately the alternatives, making a collective decision is straightforward and does not require a particular coordination mechanism provided that all robots behave the same and independently make the same choice. An example of such consensus is given by robots that must collectively transport an object towards a well advertised goal (Groß *et al.*, 2006b).

When **only some robots perceive** the necessary information to make the collective decision, a simple mechanism of information propagation is sufficient to synchronize the whole swarm. For instance, in experiments of collective hill crossing, a single robot detects the hill and informs the rest of the group about it (O'Grady *et al.*, 2005).

When **all robots partially perceive** information, for instance when perception is noisy, a distributed mechanism of averaging allows the robots to pool information and obtain more accurate information. We showed in Chapters 4 and 5 that local exchanges of information and averaging allow the robots to coordinate and to make an effective collective decision. In addition, a structure can appear inside the population of robots if some individuals hold better information than others or if they have better means to acquire information. To take advantage of this structure, we suggested in Chapter 5 the use of a measure of confidence to allow faster propagation of better measures in the swarm.

In the case of indirect information available, **no single robot perceives** the characteristics of the alternatives. We proposed two different solutions to produce a collective decision in these conditions. With stigmergy, we showed in Chapter 6 that robots can aggregate information and use it to trigger a collective decision. We also showed in Chapter 7 how the interplay of two opposite feedbacks can lead to a collective decision.

9.2.2 When, who, and how to imitate ?

We suggest here that the different flavors of imitation are produced by the variation of three main factors.

Robots can decide **when** to imitate or not a conspecific. They can for instance use a constant probability to imitate a neighbor. But they may also modulate their imitation probability as a function of the number other neighbors that have a given behavior. This type of imitation is frequently observed in animals, and at very different scales (Meunier *et al.*, 2006; Amé *et al.*, 2006; De Monte *et al.*, 2007; Pillot *et al.*, 2011). The probability to imitate can also be modulated by environmental factors, for instance when cockroaches mostly imitate stopped conspecifics when under shelters (Amé *et al.*, 2004).

Then, once the decision to imitate is made, robots must decide **who** to imitate, that is, which information to use (when using stigmergy, information is not necessarily linked to individuals anymore but the same problem of selection arises). In the simplest case, interactions only happen between pairs of robots and robots have no choice but to use information provided by their peer. When more robots are involved, each individual can imitate one or more neighbor. Obviously, paired interactions are simpler to model but they do not necessarily lead to the same collective behavior as interactions that simultaneously involve many individuals. The question of who imitates who requires attention: it is a difficult but critical design choice in artificial systems, and a debated question in biology (Ballerini *et al.*, 2008).

In addition, when studying animal behavior, biologists distinguish private and public information, that is, information which is available to conspecifics by simple observation and information that requires intentional communication to be transferred (Danchin *et al.*, 2004; Dall *et al.*, 2005; Dall, 2005). In the same way, robots can also select to whom they transmit information. The problem of information selection is therefore symmetric depending on whether it is seen from the point of view of the emitter or of the receiver.

Lastly, after all the necessary information to perform imitation is selected, robots have to process it and change their behavior accordingly. We have used only two sort of information processing to define **how** imitation is implemented in this thesis. Either robots copied the behavior of conspecifics, or they averaged information. Other types of information processing might not be considered as imitation and could lead to very different types of collective behaviors. Whether copying and averaging are the only ingredients of imitation or not is a question that requires more exploration.

9.2.3 Properties of collective decisions based on imitation

Drawing definite general conclusions on the properties of the different collective decision mechanisms exposed in this thesis is not possible because the specific details of our experiments may play an important role and impact these properties. However, there are a number of relationship for which we have an intuition, and that have also been reported in other areas of the literature.

First, when only a few individuals propagate information in a group we can expect robustness to decrease because any failing individual may propagate wrong information. Systems that average information are more likely to resist to bad information. At the same time, these systems are slower to produce a collective decision because information is repeatedly exchanged until convergence. Conversely, systems that only propagate information can be very fast. For instance the propagation of an alert signal in fish schools allow them to promptly react to the attack of a predator (Couzin, 2009).

Second, when the probability to imitate is modulated by the number of neighbors, a tradeoff between the speed and the accuracy of the collective choice can be observed. If imitation is strong with only few neighbors, individuals have less time to explore their environment before a collective decision is triggered. As a consequence, the probability to choose a less good alternative increases with the speed of the collective decision. This tradeoff has been reported and investigated in numerous biological studies and is rather well known (Franks *et al.*, 2003; Marshall *et al.*, 2006; Pratt and Sumpter, 2006; Chittka *et al.*, 2009; Sumpter and Pratt, 2009; Latty and Beekman, 2011)

Third, when the size of the group increases we may expect, in some conditions, to see the accuracy of the collective decision increase as shown in Chapter 7. In this work we also showed that the decision speed decreased linearly with the group size. However, this last result is very specific to the experimental setup we used. It is possible that in different conditions the speed of the collective decision increases with the group size, as well as the accuracy of the discrimination.

9.3 Perspectives

In the following, we present several open issues and perspectives for future research in collective decision making, swarm robotics, and more generally swarm intelligence.

9.3.1 How to know when a decision is made ?

Mechanisms of collective decision should be integrated with other collective behaviors and they should allow a group of robots to structure its activity, switching to one behavior or another as a function of the decision made. In this way, swarms could choose among different tasks and tackle them in a particular order. Consider for instance robots that have chosen to exploit a specific resource as in the experiment described in Chapter 7. Once their collective decision is made, they may start transporting objects from that particular resource to a central place.

The main obstacle to integrate decision making with other behaviors lies in the fact that robots often can not monitor the unfolding of the collective decision process. If robots do not know when they have reached a collective decision, they are unlikely to be able to switch to another task. To the best of our knowledge, there is no general solution to this problem. In Chapter 6, the decision mechanism based on artificial pheromone allows robots to know if they are not part of a selected path by monitoring the local level of pheromone, which in turn allows them to change their behavior as a function of the collective decision. However, in our other works, robots have not such information. The general problem of finding out when a decision is made remains an open issue and needs further investigation.

9.3.2 What if options are similar and the decision is uncertain ?

An interesting feature of human cognition is the ability to monitor ongoing decision processes and find out when a choice is too hard to make. This happens for instance when two alternatives are so similar that discrimination becomes random. In that case, being openly uncertain can be more effective than making a mistake because of a random decision. Evidence of this metacognitive process has been found in humans and in animals such as dolphins and monkeys (Smith *et al.*, 2003; Smith and Washburn, 2005).

Implementing a similar feature in robot swarms would lead to increased efficiency in discrimination tasks. Metacognition may be implemented in two different ways. Robots could discriminate between three alternatives, and the uncertainty of the swarm would then be expressed when the intermediate alternative is chosen. Robots could rely on a discrimination process that takes more time when offered similar alternatives. A simple timeout would allow them to avoid making a difficult decision.

9.3.3 Making decisions in discrete and continous domains

In our research, robot swarms had to reach a consensus so as to pull in the same direction or go to the same locations, or they had to choose between two or three resources. More generally, these collective decisions can be categorized as either discrete or continuous: the possible options available belonged either to a continuous domain (cooperative transport, collective localization) or to a discrete domain (resource selection, resource discrimination).

We posit that there is not a fundamental difference between these two types of processes: they both rely on imitation. A solution to unify them might be to see the continuous domain as the limit case of a discrete domain with a large number of options.

To imitate each other, robots would not select exactly the same options, but they would have higher probabilities to choose similar options. For this purpose, the notion of similarity should be defined with an appropriate metric in the space of the possible choices.

In this context, it is interesting to notice that imitation is often implemented differently depending on the nature of the domain concerned. When dealing with a discrete domain, imitation is typically implemented as a simple copy of an opinion. When continous domains are the subject of interest, an averaging function is used. The role of this function is to make robots' opinions gradually converge until they reach a consensus. However, using this function makes sense only if the opinions of the robots belong to a Gaussian (or similar) probability density function. Consider a group of flocking robots that encounter an obstacle: the swarm has to choose between left and right, or it may also temporarily split. Robots at the extremities of the flock try to avoid the obstacle, but the robots in the middle receive conflicting information. If they average the opinions of their neighbors, they go straight into the obstacle. In that case, a better solution would be to imitate only one neighbor.

9.3.4 The future of swarm robotics

Swarm intelligence is not the only approach to control groups of robots. In fact, as noticed already by Matarić (1992b), there is a whole spectrum of control strategies, from centralized systems to fully distributed systems. Swarm intelligence is at the extremity of the spectrum, and it is by no means the approach to use by default. For instance, the coordination of five robots for the inspection of a small, static area can be achieved very effectively with a central manager exploiting, for example, a Wi-Fi network. More generally, real world applications are likely to require pragmatism and the mix of several different approaches.

Nevertheless, we believe it is important to explore all the possibilities offered by swarm intelligence in collective robotics. Sometimes, this implies using swarm intelligence to solve robotics problems that could be better or more easily solved by traditional strategies. Hence, the benefits of these studies may appear limited, especially in the short term. However, the knowledge accumulated about robots swarms may become of paramount importance with the advances of technology and the appearance of new types of robots. In particular, a change of scale could open large perspectives for swarm robotics while other approaches may have difficulties to adapt.

At the macroscopic scale, we can easily imagine the benefits of having large groups of autonomous robots deployed in the oceans, in space or underground. These robots could explore their surroundings, exploit inaccessible resources and gather critical information about our environment. There are also ongoing experiments to produce fully autonomous cars (for instance the Google cars) which could cooperate in real time with each other and with devices such as satellites to improve safety and reduce traffic congestion.

At the micro or nano scale, robot swarms may be used for medical applications. This would imply a totally different conception of robots because gravity is much less important than other forces such as van der Waals attraction between molecules for instance. A promising direction lies in the reuse of existent biological material from simple and well understood bacterias. Ultimately, this could lead to the possibility of employing nano-machines to monitor our body, and maybe fix problems locally.

9.3.5 Designing complex swarm systems

Researchers in swarm intelligence are currently investigating specific behaviors such as decision making, collective motion, division of labor. In the future, if robots swarms are taken from laboratories to real life situations, they will have to handle numerous tasks. Robots may have to divide labor and let some members specialize in particular subtasks; they may also need to make several collective decisions about different subjects, sometimes simultaneously. As a consequence, a crucial problem in swarm robotics, and more generally in swarm intelligence, will be how to engineer complex collective behaviors for swarms.

Instead of redesigning from scratch all the interactions and behaviors of the robots for every new task, new collective behaviors may be designed by plugging together known ones. The challenge is to understand how these behaviors can be plugged into each other, and how they may interact or interfer. This also prompts us to find out what modifications should be brought to the behavior of the individuals and to the nature of information transfer inside the swarm in order to facilitate or minimize the aforementioned interactions and interferences.

We envision modularity will play an important role to achieve this complexity (Simon, 1962). Another promising source of inspiration comes from the very active field of evolutionary developmental biology (informally called *evo-devo*). This field is concerned with the study of developmental processes, their evolution, their plasticity, and how they lead to the production of complex systems (Bonner, 1988).

Bibliography

A

E. S. Adams. (1990). Boundary disputes in the territorial ant *Azteca trigona*: effects of asymmetries in colony size. *Animal Behaviour*, 39(2):321–328. p. 37

J.-M. Amé, C. Rivault, and J.-L. Deneubourg. (2004). Cockroach aggregation based on strain odour recognition. *Animal Behaviour*, 68:793–801. p. 25, 106, 119, 132

J.-M. Amé, J. Halloy, C. Rivault, C. Detrain, and J.-L. Deneubourg. (2006). Collegial decision making based on social amplification leads to optimal group formation. *Proceedings of the National Academy of Sciences of the United States of America*, 103(15):5835–5840. p. 27, 37, 106, 107, 108, 115, 119, 132

I. Aoki. (1982). A simulation study on the schooling mechanism in fish. *Bulletin of the Japanese Society of Scientific Fisheries*, 48(8):1081–1088. p. 32

R. C. Arkin. (1998). *Behavior-Based Robotics*. MIT Press, Cambridge, MA. p. 17

W. Ashby. (1947). Principles of the Self-Organizing Dynamic System. *The Journal of General Psychology*, 37(2):125–128. p. 13

B

T. Balch. (1999). The impact of diversity on performance in multi-robot foraging. In *Proceedings of the Third Annual Conference on Autonomous Agents*, pages 92–99. ACM Press, New York, NY. p. 30

M. Ballerini, N. Cabibbo, R. Candelier, A. Cavagna, E. Cisbani, I. Giardina, V. Lecomte, A. Orlandi, G. Parisi, A. Procaccini, M. Viale, and V. Zdravkovic. (2008). Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. *Proceedings of the National Academy of Sciences*, 105(4):1232–1237. p. 132

M. Batalin, G. Sukhatme, and M. Hattig. (2004). Mobile robot navigation using a sensor network. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 1, pages 636–641. IEEE Press, Piscataway, NJ. p. 23

R. Beckers, J. Deneubourg, S. Goss, and J. Pasteels. (1990). Collective decision making through food recruitment. *Insectes Sociaux*, 37:258–267. p. 11

R. Beckers, J.-L. Deneubourg, and S. Goss. (1992). Trails and U-turns in the selection of a path by the ant *Lasius niger*. *Journal of Theoretical Biology*, 159(4):397–415. p. 11

G. Bekey. (2005). *Autonomous Robots: From Biological Inspiration to Implementation and Control*. MIT Press, Cambridge, MA. p. 15

R. Bellman. (1958). On a routing problem. *Quarterly of Applied Mathematics*, 16(1):87–90. p. 36

G. Beni and J. Wang. (1989). Swarm intelligence in cellular robotic systems. In *Proceedings of NATO Advanced Workshop on Robots and Biological Systems*. Tuscany, Italy. p. 13, 14

G. Beni. (2005). From swarm intelligence to swarm robotics. In *Swarm Robotics*, volume 3342 of *Lecture Notes in Computer Science*, pages 1–9. Springer-Verlag, Berlin/Heidelberg, Germany. p. 13, 14

C. Bernard. (1864). Physiologie générale. Cours du Collège de France. In *Revue des cours scientifiques*, volume 2. Germer Baillière, Paris, France. p. 12

C. Bernard. (1871). La chaleur animale. Cours du Collège de France. In *Revue des cours scientifiques*, volume 1. Germer Baillière, Paris, France. p. 12

A. Birk and S. Carpin. (2006). Rescue robotics - A crucial milestone on the road to autonomous systems. *Advanced Robotics Journal*, 20(5):595–605. p. 30

S. Blanco and R. Fournier. (2003). An invariance property of diffusive random walks. *Europhysics Letters*, 61(2):168–173. p. 110

E. Bonabeau, M. Dorigo, and G. Theraulaz. (1999). *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, New York, NY. p. 21, 22, 27

J. Bonner. (1988). *The Evolution of Complexity by Means of Natural Selection*. Princeton University Press, Princeton, NJ. p. 136

J. Borenstein and L. Feng. (1996). Measurement and correction of systematic odometry errors in mobile robots. *IEEE Transactions on Robotics and Automation*, 12(6):869–880. p. 31, 33

V. Braitenberg. (1986). *Vehicles: Experiments in Synthetic Psychology*. MIT Press, Cambridge, MA. p. 16

R. Brooks. (1990). Elephants don't play chess. *Robotics and autonomous systems*, 6(1-2):3–15. p. 17

R. A. Brooks. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23. p. 17

R. A. Brooks. (1991). Intelligence without representation. *Artificial Intelligence*, 47(1-3):139–159. p. 17

J. Buck. (1935). Synchronous flashing of fireflies experimentally induced. *Science*, 81(2101):339–340. p. 25

J. Buck. (1988). Synchronous rhythmic flashing of fireflies. II. *Quarterly Review of Biology*, 63(3):265–289. p. 25

W. Burgard, D. Fox, D. Hennig, and T. Schmidt. (1996). Estimating the absolute position of a mobile robot using position probability grids. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 896–901. AAAI Press/MIT Press, Cambridge, MA. p. 33

W. Burgard, A. Derr, D. Fox, and A. B. Cremers. (1998). Integrating global position estimation and position tracking for mobile robots: The dynamic Markov localization approach. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 1–6. IEEE Press, Piscataway, NJ. p. 33

L. Bury. (2007). Conception et implémentation en c++ d'un simulateur pour les robots e-puck et réalisation de tests de validation pour la cinématique de base. Master's thesis, IRIDIA, Université libre de Bruxelles, Brussels, Belgium. p. 53

C

S. Camazine, J.-L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. (2001). *Self-Organization in Biological Systems*. Princeton Studies in Complexity. Princeton University Press, Princeton, NJ, USA. p. 21, 27, 36

A. Campo and M. Dorigo. (2007). Efficient multi-foraging in swarm robotics. In *Proceedings of the* 9^{th} *European Conference on Advances in Artificial Life*, volume 4648 of *Lecture Notes in Computer Science*, pages 696–705. Springer-Verlag, Berlin/Heidelberg, Germany. p. 7, 30

A. Campo, S. Nouyan, M. Birattari, R. Groß, and M. Dorigo. (2006a). Negotiation of goal direction for cooperative transport. In *Proceedings of the* 5^{th} *International Workshop on Ant Colony Optimization and Swarm Intelligence*, volume 4150 of *Lecture Notes in Computer Science*, pages 191–202. Springer-Verlag, Berlin/Heidelberg, Germany. p. 5

A. Campo, S. Nouyan, M. Birattari, R. Groß, and M. Dorigo. (2006b). Enhancing cooperative transport with negotiation of goal direction in swarm robotics. In *Proceedings of the* 18^{th} *Belgium-Netherlands Conference on Artificial Intelligence*, pages 365–366. University of Namur, Namur, Belgium. p. 5

A. Campo, S. Garnier, O. Dedriche, M. Zekkri, and M. Dorigo. (2010a). Self-organized collective discrimination: Finding resources of specific availability with minimal cognitive requirements. Technical Report 2010-021, IRIDIA, Université Libre de Bruxelles, Brussels, Belgium. p. 6

A. Campo, A. Gutiérrez, S. Garnier, M. Moussaïd, and M. Dorigo. (under review). Controlling the choice of robot swarms using collective discrimination. *IEEE Transactions on Systems, Man, and Cybernetics–Part B*. p. 6

A. Campo, A. Gutiérrez, S. Nouyan, C. Pinciroli, V. Longchamp, S. Garnier, and M. Dorigo. (2010b). Artificial pheromone for path selection by a foraging swarm of robots. *Biological Cybernetics*, pages 1–14. p. 6

W. Cannon. (1929). Organization for physiological homeostasis. *Physiological Reviews*, 9(3):399. p. 12

Y. U. Cao, A. S. Fukunaga, and B. Kahng, Andrew. (1997). Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, 4(1):7–27. p. 17, 18, 28

G. Caprari, T. Estier, and R. Siegwart. (2002). Fascination of down scaling - Alice the sugar cube robot. *Journal of Micro-Mechatronics*, 1(3):177–189. p. 36

A. R. Cassandra, L. P. Kaelbling, and J. A. Kurien. (1996). Acting under uncertainty: Discrete Bayesian models for mobile-robot navigation. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 963–972. IEEE Press, Piscataway, NJ. p. 33

J. M. Chambers, W. S. Cleveland, B. Kleiner, and P. A. Tukey. (1983). *Graphical Methods for Data Analysis*. The Wadsworth statistics / probability series. Wadsworth and Brooks/Cole, Pacific Grove, CA. p. 61, 62

L. Chittka, P. Skorupski, and N. Raine. (2009). Speed-accuracy tradeoffs in animal decision making. *Trends in Ecology & Evolution*, 24(7):400–407. p. 133

K. Chong and L. Kleeman. (1997). Accurate odometry and error modelling for a mobile robot. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2783–2788. IEEE Press, Piscataway, NJ. p. 33

A. L. Christensen. (2005). Efficient neuro-evolution of hole-avoidance and phototaxis for a swarm-bot. Technical Report 2005-014, IRIDIA, Université libre de Bruxelles, Brussels, Belgium. p. 53

A. Christensen and M. Dorigo. (2006). Incremental evolution of robot controllers for a highly integrated task. In *Proceedings of The* 9^{th} *International Conference on the Simulation of Adaptive Behavior*, Lecture Notes in Computer Science, pages 473–484. Springer-Verlag, Berlin/Heidelberg, Germany. p. 20

A. Christensen, R. O'Grady, and M. Dorigo. (2009). From fireflies to fault-tolerant swarms of robots. *IEEE Transactions on Evolutionary Computation*, 13(4):754–766. p. 25

W. J. Clancey. (1997). *Situated Cognition: On Human Knowledge and Computer Representations*. Cambridge University Press, Cambridge, UK. p. 24

C. Clark and M. Mangel. (1986). The evolutionary advantages of group foraging. *Theoretical Population Biology*, 30(1):45–75. p. 14

A. Cobham. (1965). The intrinsic computational difficulty of functions. In *Proceedings of the International Congress for Logic, Methodology and Philosophy of Science*, pages 24–30. North Holland Publishing Co., Amsterdam. p. 16

E. A. Codling, M. J. Plank, and S. Benhamou. (2008). Random walk models in biology. *Journal of the Royal Society Interface*, 5(25):813–834. p. 108, 119

L. Conradt and C. List. (2009). Group decisions in humans and animals: a survey. *Philosophical Transactions of the Royal Society of London. Series B, Biological sciences*, 364(1518):719–742. p. 21

L. Conradt and T. J. Roper. (2005). Consensus decision making in animals. *Trends in Ecology & Evolution*, 20(8):449–456. p. 21

S. Cook. (1971). The complexity of theorem-proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pages 151–158. ACM Press, New York, NY. p. 16

S. J. Cooper. (2008). From Claude Bernard to Walter Cannon. Emergence of the concept of homeostasis. *Appetite*, 51(3):419–427. p. 12

N. Correll and A. Martinoli. (2006). Collective inspection of regular structures using a swarm of miniature robots. *Experimental Robotics IX*, pages 375–386. p. 30

I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks. (2002). Collective memory and spatial sorting in animal groups. *Journal of Theoretical Biology*, 218(1):1–11. p. 32, 58

I. D. Couzin. (2009). Collective cognition in animal groups. *Trends in cognitive sciences*, 13(1):36–43. p. 133

D

S. R. Dall, L.-A. Giraldeau, O. Olsson, J. M. McNamara, and D. W. Stephens. (2005). Information and its use by animals in evolutionary ecology. *Trends in Ecology & Evolution*, 20(4):187 – 193. p. 132

S. Dall. (2005). Defining the concept of public information. *Science*, 308(5720):353. p. 132

É. Danchin, L. Giraldeau, T. Valone, and R. Wagner. (2004). Public information: from nosy neighbors to cultural evolution. *Science*, 305(5683):487. p. 132

S. De Monte, F. d'Ovidio, S. Danø, and P. G. Sørensen. (2007). Dynamical quorum sensing: Population density encoded in cellular dynamics. *Proceedings of the National Academy of Sciences of the United States of America*, 104(47):18377–18381. p. 132

J.-L. Deneubourg and S. Goss. (1989). Collective patterns and decision making. *Ethology Ecology and Evolution*, 1(4):295–311. p. 23, 87, 90

J.-L. Deneubourg, S. Aron, S. Goss, and J.-M. Pasteels. (1990). The self-organizing exploratory pattern of the argentine ant. *Journal of Insect Behavior*, 3(1):159–168. p. 90

G. Di Caro and M. Dorigo. (1998). AntNet: Distributed stigmergetic control for communications networks. *Journal of Artificial Intelligence Research*, 9:317–365. p. 36, 105

M. Dorigo and E. Şahin. (2004). Guest Editorial. Special Issue: Swarm Robotics. *Autonomous Robots*, 17(2–3):111–113. p. 36, 40

M. Dorigo and T. Stützle. (2004). *Ant Colony Optimization*. MIT Press, Cambridge, MA. p. 105

M. Dorigo, V. Trianni, E. Şahin, R. Groß, T. H. Labella, G. Baldassarre, S. Nolfi, J.-L. Deneubourg, F. Mondada, D. Floreano, and L. M. Gambardella. (2004). Evolving self-organizing behaviors for a swarm-bot. *Autonomous Robots*, 17(2–3):223–245. p. 40

M. Dorigo, V. Maniezzo, and A. Colorni. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics–Part B*, 26(1):29–41. p. 36, 105

G. Dudek and P. Mackenzie. (1993). Model-based map construction for robot localization. In *Proceedings of Vision Interface*, pages 97–102. CIPPR Society Press, Toronto, ON. p. 33

G. Dudek, M. Jenkin, E. Milios, and D. Wilkes. (1996). A taxonomy for multi-agent robotics. *Autonomous Robots*, 3(4):375–397. p. 17

E

H. Endres, W. Feiten, and G. Lawitzky. (2002). Field test of a navigation system: Autonomous cleaning in supermarkets. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 2, pages 1779–1781. IEEE Press, Piscataway, NJ. p. 28

F

L. Feng, J. Borenstein, and H. Everett. (1994). *Where am I? Sensors and Methods for Autonomous Mobile Robot Positioning*. University of Michigan Press, Ann Arbor, MI. p. 71

K. Finkenzeller. (2010). *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication*. John Wiley & Sons, Ltd., Chichester, UK. p. 87

J. Foley, A. Van Dam, S. Feiner, J. Hughes, and R. Phillips. (1994). *Introduction to Computer Graphics*, volume 1990. Addison-Wesley, Reading, MA. p. 44

D. Fox, W. Gurgard, H. Kruppa, and S. Thrun. (2000). Collaborative multi-robot exploration. *Autonomous Robots*, 8(3):325–344. p. 31, 33

N. R. Franks, A. Dornhaus, J. P. Fitzsimmons, and M. Stevens. (2003). Speed versus accuracy in collective decision making. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 270 (1532):2457–2463. p. 133

G. Frege. (1879). *Begriffsschrift: Eine der Arithmetischen Nachgebildete Formelsprache des Reinen Denkens*. Louis Nebert, Halle, Germany. p. 16

M. Frison, N.-L. Tran, N. Baiboun, A. Brutschy, G. Pini, A. Roli, M. Dorigo, and M. Birattari. (2010). Self-organized task partitioning in a swarm of robots. In *Swarm Intelligence*, volume 6234 of *Lecture Notes in Computer Science*, pages 287–298. Springer-Verlag, Berlin/Heidelberg, Germany. p. 31

G

M. R. Garey and D. S. Johnson. (1979). *Computers and Intractability: A Guide to the Theory of $\mathcal{NP}$-Completeness*. Freeman, San Francisco, CA. p. 16

S. Garnier, C. Jost, R. Jeanson, J. Gautrais, M. Asadpour, G. Caprari, and G. Theraulaz. (2005). Aggregation behaviour as a source of collective decision in a group of cockroach-like-robots. *Advances in Artificial Life*, pages 169–178. p. 27

S. Garnier, J. Gautrais, and G. Theraulaz. (2007). The biological principles of swarm intelligence. *Swarm Intelligence*, 1(1):3–31. p. 21, 31, 36, 105

S. Garnier, C. Jost, J. Gautrais, M. Asadpour, G. Caprari, R. Jeanson, A. Grimal, and G. Theraulaz. (2008). The embodiment of cockroach aggregation behavior in a group of micro-robots. *Artificial Life*, 14(4):387–408. p. 25

J. Gautrais, C. Jost, M. Soria, A. Campo, S. Motsch, R. Fournier, S. Blanco, and G. Theraulaz. (2009). Analyzing fish movement as a persistent turning walker. *Journal of mathematical biology*, 58(3):429–445. p. 7

J. Giles. (2005). Internet encyclopaedias go head to head. *Nature*, 438(7070):900–901. p. 11

D. E. Goldberg. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA. p. 109

D. Goldberg and M. J. Matarić. (1997). Interference as a tool for designing and evaluating multi-robot controllers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 637–642. AAAI Press, Cambridge, MA. p. 15, 31, 37

S. Goss, S. Aron, J.-L. Deneubourg, and J.-M. Pasteels. (1989). Self-organized shortcuts in the Argentine ant. *Naturwissenschaften*, 76(12):579–581. p. 92

R. Grabowski, L. E. Navarro-Serment, C. J. J. Paredis, and P. K. Khosla. (2000). Heterogeneous teams of modular robots for mapping and exploration. *Autonomous Robots*, 8(2):293–308. p. 33

P.-P. Grassé. (1959). La reconstruction du nid et les coordinations interindividuelles chez *Bellicositermes natalensis* et *Cubitermes sp*. La théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. *Insectes sociaux*, 6(1):41–80. p. 22

P.-P. Grassé. (1984). *Termitologia, Tome II. Fondation des sociétés. Construction*. Masson, Paris. p. 22

W. Grey Walter. (1950). An imitation of life. *Scientific American*, 182(5):42–45. p. 15

W. Grey Walter. (1951). A machine that learns. *Scientific American*, 185(2):60–63. p. 15

W. Grey Walter. (1954). The Living Brain. *The Journal of Nervous and Mental Disease*, 120(1-2):144. p. 15

R. Groß and M. Dorigo. (2009). Towards group transport by swarms of robots. *International Journal of Bio-Inspired Computation*, 1(1–2):1–13. p. 34

R. Groß, F. Mondada, and M. Dorigo. (2006a). Transport of an object by six pre-attached robots interacting via physical links. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1317–1323. IEEE Press, Los Alamitos, CA. p. 18

R. Groß, E. Tuci, M. Dorigo, M. Bonani, and F. Mondada. (2006b). Object transport by modular robots that self-assemble. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2558–2564. IEEE Press, Los Alamitos, CA. p. 18, 31, 59, 131

R. Groß, S. Nouyan, M. Bonani, F. Mondada, and M. Dorigo. (2008). Division of labour in self-organised groups. In *Proceedings of the* 10^{th} *International Conference on Simulation of Adaptive Behavior*, volume 5040 of *Lecture Notes in Artificial Intelligence*, pages 426–436. Springer-Verlag, Berlin, Germany. p. 30

A. Gutiérrez, A. Campo, M. Dorigo, J. Donate, F. Monasterio-Huelin, and L. Magdalena. (2009a). Open e-puck range and bearing miniaturized board for local communication in swarm robotics. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3111–3116. IEEE Press, Piscataway, NJ. p. 5, 48, 89

A. Gutiérrez, A. Campo, M. Dorigo, D. Amor, L. Magdalena, and F. Monasterio-Huelin. (2008). An open localization and local communication embodied sensor. *Sensors*, 8(11):7545–7563. p. 5, 48, 89

A. Gutiérrez, A. Campo, F. C. Santos, F. Monasterio-Huelin, and M. Dorigo. (2009b). Social odometry: Imitation based odometry in collective robotics. *International Journal of Advanced Robotic Systems*, 6 (2):1–8. p. 6

A. Gutiérrez, E. Tuci, and A. Campo. (2009c). Evolution of neuro-controllers for robots' alignment using local communication. *International Journal of Advanced Robotic Systems*, 6(1):1–10. p. 7

A. Gutiérrez, A. Campo, F. Monasterio-Huelin, L. Magdalena, and M. Dorigo. (2010). Collective decision making based on social odometry. *Neural Computing & Applications*. p. 5

J.-S. Gutmann, T. Weigel, and B. Nebel. (1999). Fast, accurate and robust self-localization in polygonal environments. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 1412–1419. IEEE Press, Piscataway, NJ. p. 33

J.-S. Gutmann, T. Weigel, and B. Nebel. (2001). A fast, accurate, and robust method for self-localization in polygonal environments using laser-rangefinders. *Advanced Robotics Journal*, 14(1):1–17. p. 33

H

D. Hahnel, W. Burgard, D. Fox, K. Fishkin, and M. Philipose. (2004). Mapping and localization with RFID technology. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1015–1020. IEEE Press, Piscataway, NJ. p. 23

P. Hammerstein. (2003). *Genetic and Cultural Evolution of Cooperation*. MIT Press, Cambridge, MA. p. 73

A. Hayes and P. Dorminiani-Tabatabaei. (2002). Self-organized flocking with agent failure: Off-line optimization and demostrantion with real robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3900–3905. IEEE Press, Piscataway, NJ. p. 32

B. Hayes-Roth. (1985). A blackboard architecture for control. *Artificial intelligence*, 26(3):251–321. p. 18, 23

B. Hölldobler and E. Wilson. (1990). *The Ants*. Belknap Press, Harvard University Press, Cambridge, MA. p. 14, 23

D. A. Holway and T. J. Case. (2001). Effects of colony-level variation on competitive ability in the invasive Argentine ant. *Animal Behaviour*, 61(6):1181–1192. p. 37, 105

A. Huth and C. Wissel. (1992). The simulation of the movement of fish schools. *Journal of Theoretical Biology*, 156:365–385. p. 32

J

S. R. Jammalamadaka and A. SenGupta. (2001). *Topics in Circular Statistics*. World Scientific Press, Singapore. p. 56, 58

R. Jeanson, J.-L. Deneubourg, A. Grimal, and G. Theraulaz. (2004). Modulation of individual behavior and collective decision-making during aggregation site selection by the ant *Messor* barbarus. *Behavioral Ecology and Sociobiology*, 55(4):388–394. p. 25

K

R. E. Kalman. (1960). A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering, Series D*, 82(1):35–45. p. 33

I. Kelly and A. Martinoli. (2004). A scalable, on-board localisation and communication system for indoor multi-robot experiments. *Sensor Review*, 24(2):167–180. p. 32

I. Kelly and D. Keating. (1996). Flocking by the fusion of sonar and active infrared sensors on physical autonomous mobile robots. In *Third International Conference on Mechatronics and Machine Vision in Practice*, pages 1–4. Pergamon Press, Oxford, NY. p. 32

J. Kennedy, R. C. Eberhart, and Y. Shi. (2001). *Swarm Intelligence*. Morgan Kaufmann, San Francisco, CA. p. 21

E. Klavins. (2002). Communication complexity of multi-robot systems. In *Algorithmic Foundations of Robotics V*, Springer Tracts in Advanced Robotics 7, pages 275–292. Springer-Verlag, Berlin, Germany. p. 14, 18

T. Kohonen. (1982). Self-organized formation of topologically correct feature maps. *Biological cybernetics*, 43(1):59–69. p. 46

T. Kohonen. (2002). The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480. p. 46

J. Krause and G. D. Ruxton. (2002). *Living in Groups*. Oxford University Press, Oxford, UK. p. 37, 105

R. Kurazume, S. Hirose, S. Nagata, and N. Sashida. (1996). Study on cooperative positioning system (basic principle and measurement experiment). In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 1421–1426. IEEE Press, Piscataway, NJ. p. 33

L

T. H. Labella, M. Dorigo, and J.-L. Deneubourg. (2004). Efficiency and task allocation in prey retrieval. *Biologically Inspired Approaches to Advanced Information Technology*, pages 274–289. p. 31

T. Larsen, M. Bak, N. Andersen, and O. Ravn. (1998). Location estimation for autonomously guided vehicle using an augmented Kalman filter to autocalibrate the odometry. In *Proceedings of the FUSION 98 Spie Conference*, pages 33–39. CSREA Press, Las Vegas, NV. p. 33

T. Latty and M. Beekman. (2011). Speed–accuracy trade-offs during foraging decisions in the acellular slime mould physarum polycephalum. *Proceedings of the Royal Society B: Biological Sciences*, 278 (1705):539–545. p. 133

K. Lerman, A. Galstyan, A. Martinoli, and A. Ijspeert. (2001). A macroscopic analytical model of collaboration in distributed robotic systems. *Artificial Life*, 7(4):375–393. p. 19

K. Lerman, A. Martinoli, and A. Galstyan. (2005). A review of probabilistic macroscopic models for swarm robotic systems. In *Swarm Robotics*, volume 3342 of *Lecture Notes in Computer Science*, pages 143–152. p. 19

E. M. Lichtenberg, V. L. Imperatriz-Fonseca, and J. C. Nieh. (2010). Behavioral suites mediate group-level foraging dynamics in communities of tropical stingless bees. *Insectes Sociaux*, 57(1):105–113. p. 37, 105

M

J. A. R. Marshall, A. Dornhaus, N. R. Franks, and T. Kovacs. (2006). Noise, cost and speed-accuracy trade-offs: decision-making in a decentralized system. *Journal of the Royal Society, Interface*, 3(7): 243–254. p. 133

A. Martinelli and R. Siegwart. (2003). Estimating the odometry error of a mobile robot during navigation. In *Proceedings of the 1^{st} European Conference on Mobile Robots*, pages 218–223. Zturek Research Scientific Institute Press, Warsaw, Poland. p. 33

A. Martinoli, K. Easton, and W. Agassounon. (2004). Modeling swarm robotic systems: A case study in collaborative distributed manipulation. *The International Journal of Robotics Research*, 23(4-5):415–436. p. 19

A. Martinoli and K. Easton. (2003). Modeling swarm robotic systems. *Springer Tracts in Advanced Robotics*, 5:297–306. p. 19

M. J. Matarić. (1997). Learning social behavior. *Robotics and Autonomous Systems*, 20(2):191–204. p. 37

M. Matarić. (1992a). Behavior-based control: Main properties and implications. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 46–54. IEEE Press, Piscataway, NJ. p. 17

M. Matarić. (1992b). Minimizing complexity in controlling a mobile robot population. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 830–835. IEEE Press, Piscataway, NJ. p. 17, 135

J. Maxwell. (1868). On governors. *Proceedings of the Royal Society of London*, 16:270–283. p. 13

P. J. McCarthy and J. Hayes. (1969). Some philosophical problems from the standpoint of artificial intelligence. In *Machine Intelligence*, pages 463–502. Edinburgh University Press, Edinburgh, UK. p. 16

L. Merino, F. Caballero, J. Martínez-de Dios, J. Ferruz, and A. Ollero. (2006). A cooperative perception system for multiple UAVs: Application to automatic detection of forest fires. *Journal of Field Robotics*, 23(3-4):165–184. p. 23

N. Meteopolis and S. Ulam. (1949). The Monte Carlo method. *Journal of the American Statistical Association*, 44(247):335–341. p. 19

H. Meunier, J. Leca, J. Deneubourg, and O. Petit. (2006). Group movement decisions in capuchin monkeys: the utility of an experimental study and a mathematical model to explore the relationship between individual and collective behaviours. *Behaviour*, 143(17):1511–1527. p. 132

R. E. Mirollo and S. H. Strogatz. (1990). Synchronization of pulse-coupled biological oscillators. *SIAM Journal of Applied Mathematics*, 50:1645–1662. p. 25

F. Mondada, M. Bonani, A. Guignard, S. Magnenat, C. Studer, and D. Floreano. (2005). Superlinear physical performances in a SWARM-BOT. *Advances in Artificial Life*, pages 282–291. p. 14

F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J.-L. Deneubourg, S. Nolfi, L. M. Gambardella, and M. Dorigo. (2004). Swarm-Bot: A new distributed robotic concept. *Autonomous Robots*, 17(2):193–221. p. 40, 41

F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. M. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli. (2009). The e-puck, a robot designed for education in engineering. In *Proceedings of the* 9^{th} *Conference on Autonomous Robot Systems and Competitions*, pages 59–65. IPCB-Instituto Politécnico de Castelo Branco, Castelo Branco, Portugal. p. 41

M. Monta, N. Kondo, and Y. Shibano. (2002). Agricultural robot in grape production system. In *Proceedings of the IEEE International Confereence on Robotics and Automation*, volume 3, pages 2504–2509. IEEE Press, Piscataway, NJ. p. 28

N

T. Nagamitsu and T. Inoue. (1997). Aggressive foraging of social bees as a mechanism of floral resource partitioning in an Asian tropical rainforest. *Oecologia*, 110(3):432–439. p. 37, 105

K. Nealson, T. Platt, and J. Hastings. (1970). Cellular control of the synthesis and activity of the bacterial luminescent system. *Journal of bacteriology*, 104(1):313. p. 27

A. Newell and H. A. Simon. (1956). The logic theory machine: A complex information processing system. *IRE Transactions of information theory*, 2-3:61–79. p. 16

A. Newell and H. A. Simon. (1976). Computer science as empirical enquiry: Symbols and search. *Communications of the ACM*, 19(3):113–126. p. 16

G. Nicolis, I. Prigogine, and G. Nicolis. (1977). *Self-Organization in Nonequilibrium Systems: From Dissipative Structures to Order through Fluctuations*. John Wiley & Sons, New York, NY. p. 13

S. Nolfi and D. Floreano. (2000). *Evolutionary robotics: The Biology, intelligence, and Technology of Self-Organizing Machines*. MIT Press, Cambridge,MA. p. 20

S. Nouyan, A. Campo, and M. Dorigo. (2008). Path formation in a robot swarm: Self-organized strategies to find your way home. *Swarm Intelligence*, 2(1):1–23. p. 6, 31, 34, 88

S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. (2009a). Teamwork in self-organized robot colonies. *IEEE Transactions on Evolutionary Computation*, 13(4):695–711. p. 18

S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. (2006). Group transport along a robot chain in a self-organised robot colony. In *Proceedings of the 9^{th} International Conference on Intelligent Autonomous Systems*, pages 433–442. IOS Press, Amsterdam, The Netherlands. p. 31, 34, 88

S. Nouyan, R. Groß, M. Bonani, F. Mondada, and M. Dorigo. (2009b). Teamwork in self-organized robot colonies. *IEEE Transactions on Evolutionary Computation*, 13(4):695–711. p. 34

O

R. O'Grady, R. Groß, F. Mondada, M. Bonani, and M. Dorigo. (2005). Self-assembly on demand in a group of physical autonomous mobile robots navigating rough terrain. In *Advances in Artificial Life*, volume 3630 of *Lecture Notes in Computer Science*, pages 272–281. Springer Berlin / Heidelberg. p. 131

P

D. Paley, N. Leonard, R. Sepulchre, D. Grünbaum, and J. Parrish. (2007). Oscillator models and collective motion. *IEEE Control Systems Magazine*, 27(4):89–105. p. 25

L. E. Parker. (1998). ALLIANCE: an architecture for fault tolerant multirobot cooperation. *IEEE Transaction on Robotics and Automation*, 14(2):220–240. p. 30

J. M. Pasteels, J.-L. Deneubourg, and S. Goss. (1987). Self-organization mechanisms in ant societies (I) : trail recruitment to newly discovered food sources. In *From Individual to Collective Behavior in Social Insects*, volume 54, pages 155–176. Birkhäuser Verlag, Basel, Switzerland. p. 90

D. Payton, M. Daily, R. Estowski, M. Howard, and C. Lee. (2001). Pheromone robotics. *Autonomous Robots*, 11(3):319–324. p. 35

S. Pedersen, S. Fountas, H. Have, and B. Blackmore. (2006). Agricultural robots—system analysis and economic feasibility. *Precision Agriculture*, 7(4):295–308. p. 28

D. T. Pham, A. Ghanbarzadeh, E. Koç, S. Otri, S. Rahim, and M. Zaidi. (2006). The bees algorithm, a novel tool for complex optimisation problems. In *Proceedings of the 2nd Virtual International Conference on Intelligent Production Machines and Systems*, pages 454–459. Science Publishers Ltd., Oxford, UK. p. 36, 105

A. Pickering. (2010). *The Cybernetic Brain: Sketches of Another Future*. University Of Chicago Press, Chicago, IL. p. 15

M.-H. Pillot, J. Gautrais, P. Arrufat, I. D. Couzin, R. Bon, and J.-L. Deneubourg. (2011). Scalable rules for coherent group motion in a gregarious vertebrate. *PLoS ONE*, 6(1):e14487. p. 132

R. Poli, J. Kennedy, and T. Blackwell. (2007). Particle swarm optimization. *Swarm Intelligence*, 1:33–57. p. 36, 105

E. Prassler, A. Ritter, C. Schaeffer, and P. Fiorini. (2000). A short history of cleaning robots. *Autonomous Robots*, 9(3):211–226. p. 28

S. C. Pratt. (2005). Quorum sensing by encounter rates in the ant *Temnothorax albipennis*. *Behavioral Ecology*, 16(2):488–496. p. 27, 38, 107, 108, 119

S. C. Pratt and D. J. T. Sumpter. (2006). A tunable algorithm for collective decision-making. *Proceedings of the National Academy of Sciences of the United States of America*, 103(43):15906–15910. p. 133

J. Pugh and A. Martinoli. (2006). Relative localization and communication module for small-scale multi-robot systems. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 188–193. IEEE Press, Piscataway, NJ. p. 50

A. H. Purnamadjaja and R. A. Russell. (2004). Pheromone communicate: Implementation of necrophoric bee behaviour in a robot swarm. In *Proceedings of the IEEE Conference on Robotics, Automation and Mechatronics*, pages 638–643. IEEE Press, Piscataway, NJ. p. 21, 36

R

M. O. Rabin. (1963). Probabilistic automata. *Information and Control*, 6(3):230–245. p. 108, 119

M. Rachkov, L. Marques, and A. de Almeida. (2005). Multisensor demining robot. *Autonomous Robots*, 18(3):275–291. p. 28

A. Regmi, R. Sandoval, R. Byrne, H. Tanner, and C. Abdallah. (2005). Experimental implementation of flocking algorithms in wheeled mobile robots. In *Proceedings of the American Control Conference*, pages 4917–4922. IEEE Press, Piscataway, NJ. p. 32

F. Reif. (1965). *Fundamentals of statistical and thermal physics*. McGraw-Hill, New York, NY. p. 67, 73, 108

I. M. Rekleitis, G. Dudek, and E. Milios. (2001). Multi-robot collaboration for robust exploration. *Annals of Mathematics and Artificial Intelligence*, 31(1):7–40. p. 33

I. M. Rekleitis, G. Dudek, and E. Milios. (2003). Probabilistic cooperative localization and mapping in practice. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1907–1912. IEEE Press, Piscataway, NJ. p. 33

C. W. Reynolds. (1987). Flocks, herds, and schools: a distributed behavioral model. *Computer Graphics*, 21(4):25–34. p. 22, 32

J. Roberts, E. Duff, P. Corke, P. Sikka, G. Winstanley, and J. Cunningham. (2002). Autonomous control of underground mining vehicles using reactive navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 4, pages 3790–3795. IEEE Press, Piscataway, NJ. p. 28

R. A. Russell. (1999). Ant trails - An example for robots to follow? In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2698–2703. IEEE Press, Piscataway, NJ. p. 36

S. Russell and P. Norvig. (1995). *Artificial Intelligence: A Modern Approach*. Prentice hall, Upper Saddle River, NJ. p. 16, 36

S

F. C. Santos, J. M. Pacheco, and T. Lenaerts. (2006). Cooperation prevails when individuals adjust their social ties. *PLoS Computational Biology*, 2(10):e140. p. 73

Y. Sarig. (1993). Robotics of fruit harvesting: A state-of-the-art review. *Journal of Agricultural Engineering Research*, 54(4):265–280. p. 28

T. Schmickl and K. Crailsheim. (2008). Throphallaxis within a robotic swarm: Bio-inspired communication among robots in a swarm. *Autonomous Robots*, 25(1):171–188. p. 35

T. Seeley, P. Visscher, and K. Passino. (2006). Group decision making in honey bee swarms. *American Scientist*, 94(3):220. p. 27

T. D. Seeley. (2002). When is self-organization used in biological systems? *The Biological Bulletin*, 202 (3):314–318. p. 14, 37

R. Sepulchre, D. Paley, and N. Leonard. (2004). Collective motion and oscillator synchronization. *Cooperative control*, pages 466–469. p. 25

C. E. Shannon. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27: 379–423. p. 21

M. C. Silverman, D. Nies, B. Jung, and G. S. Sukhatme. (2002). Staying alive: A docking station for autonomous robot recharging. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1050–1055. IEEE Press, Piscataway, NJ. p. 30

R. Simmons and S. Koenig. (1995). Probabilistic robot navigation in partially observable environments. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1080 – 1087. Morgan Kaufmann, San Mateo, CA. p. 33

H. A. Simon. (1962). The architecture of complexity. *Proceedings of the American Philosophical Society*, 106(6):467–482. p. 14, 136

A. R. Smith. (1978). Color gamut transform pairs. *Computer Graphics*, 12(3):12–19. p. 44

J. Smith and D. Washburn. (2005). Uncertainty monitoring and metacognition by animals. *Current Directions in Psychological Science*, 14(1):19. p. 134

J. Smith, W. Shields, and D. Washburn. (2003). The comparative psychology of uncertainty monitoring and metacognition. *Behavioral and Brain Sciences*, 26(03):317–339. p. 134

J. E. Smith, J. M. Kolowski, K. E. Graham, S. E. Dawes, and K. E. Holekamp. (2008). Social and ecological determinants of fission-fusion dynamics in the spotted hyaena. *Animal Behaviour*, 76: 619–636. p. 37

R. C. Smith and P. Cheeseman. (1987). On the representation and estimation of spatial uncertainly. *International Journal of Robotics Research*, 5(4):56–68. p. 71

E. Stella, F. Musio, L. Vasanelli, and A. Distante. (1995). Goal-oriented mobile robot navigation using an odour sensor. In *Proceedings of the Intelligent Vehicles Symposium*, pages 147–151. IEEE Press, Piscataway, NJ. p. 21, 36

K. Støy. (2001). Using situated communication in distributed autonomous mobile robots. In *Proceedings of the 7^{th} Scandinavian Conference on Artificial Intelligence*, pages 44–52. IOS Press, Amsterdam, The Netherlands. p. 24

S. Strogatz. (2000). *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Westview Press, Cambridge, MA. p. 27

K. Sugawara, T. Kazama, and T. Watanabe. (2004). Foraging behavior of interacting robots with virtual pheromone. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3074–3079. IEEE Press, Piscataway, NJ. p. 36, 105

D. J. T. Sumpter. (2010). *Collective Animal Behavior*. Princeton University Press, Princeton, NJ. p. 37, 105

D. J. T. Sumpter and S. C. Pratt. (2009). Quorum responses and consensus decision making. *Philosophical Transactions of the Royal Society of London. Series B, Biological sciences*, 364(1518):743–753. p. 133

J. Svennebring and S. Koenig. (2003). Trail-laying robots for robust terrain coverage. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 75–82. IEEE Press, Piscataway, NJ. p. 21, 36

M. Szymanski, T. Breitling, J. Seyfried, and H. Wörn. (2006). Distributed shortest-path finding by a micro-robot swarm. In *Proceedings of the 5^{th} International Workshop on Ant Colony Optimization and Swarm Intelligence*, volume 4150 of *Lecture Notes in Computer Science*, pages 404–411. Springer-Verlag, Berlin, Germany. p. 33, 35

T

G. Theraulaz and E. Bonabeau. (1995). Coordination in distributed building. *Science*, 269(5224): 686–686. p. 23, 87

S. Thrun, D. Hahnel, D. Ferguson, M. Montemerlo, R. Triebel, W. Burgard, C. Baker, Z. Omohundro, S. Thayer, and W. Whittaker. (2003). A system for volumetric robotic mapping of abandoned mines. In *Proceedings of the IEEE International Conference on Robotics and Automation*, volume 3, pages 4270–4275. IEEE Press, Piscataway, NJ. p. 30

E. Toffin, D. Di Paolo, A. Campo, C. Detrain, and J.-L. Deneubourg. (2009). Shape transition during nest digging in ants. *Proceedings of the National Academy of Sciences*, 106(44):18616–18620. p. 7

A. Traulsen, M. Nowak, and J. Pacheco. (2006). Stochastic dynamics of invasion and fixation. *Physical Review - Serie E*, 74(1):011909.1–011909.5. p. 73

A. Traulsen, J. Pacheco, and M. Nowak. (2007). Pairwise comparison and selection temperature in evolutionary game dynamics. *Journal of Theoretical Biology*, 246(3):522–529. p. 73

V. Trianni. (2008). *Evolutionary Swarm Robotics. Evolving Self-Organising Behaviours in Groups of Autonomous Robots*, volume 108 of *Studies in Computational Intelligence*. Springer-Verlag, Berlin, Germany. p. 15, 20

V. Trianni and M. Dorigo. (2005). Emergent collective decisions in a swarm of robots. In *Proceedings of the IEEE International Symposium on Swarm Intelligence*, pages 241–248. IEEE Press, Piscataway, NJ. p. 18

A. E. Turgut, H. Çelikkanat, F. Gökçe, and E. Şahin. (2008). Self-organized flocking in mobile robot swarms. *Swarm Intelligence*, 2(2-4):97–120. p. 22, 32

V

R. T. Vaughan, K. Støy, G. S. Sukhatme, and M. J. Matarić. (2002). LOST: Localization-space trails for robot. *IEEE Transactions on Robotics and Automation*, 18(5):796–812. p. 33

T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet. (1995). Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75(6):1226–1229. p. 32

L. Von Bertalanffy. (1950a). The theory of open systems in physics and biology. *Science*, 111(2872): 23–29. p. 13, 19

L. Von Bertalanffy. (1950b). An outline of general system theory. *British Journal for the Philosophy of Science*, 1(2):134–165. p. 13

L. Von Bertalanffy. (1968). *General System theory: Foundations, Development, Applications*. George Braziller, New York, NY. p. 13

W

C. M. Wang. (1988). Location estimation and uncertainty analysis for mobile robots. *Autonomous Robot Vehicles*, 1(1):1230–1235. p. 33

J. Wawerla and R. T. Vaughan. (2008). Optimal robot recharging strategies for time discounted labour. In *Proceedings of the 11^{th} International Conference on Artificial Life*, pages 670–677. MIT Press, Cambridge, MA. p. 30

B. Werger and M. J. Matarić. (1996). Robotic food chains: Externalization of state and program for minimal-agent foraging. In *Proceedings of the 4^{th} International Conference on Simulation of Adaptive Behavior*, pages 625–634. MIT Press/Bradford Books, Cambridge, MA. p. 31, 34

A. N. Whitehead and B. Russell. (1910). *Principia Mathematica*. Cambridge University Press, Cambridge, UK. p. 16

N. Wiener. (1948). *Cybernetics: Or Control and Communication in the Animal and the Machine*. MIT Press, Cambridge, MA. p. 13, 21

A. Winfield. (2009). Towards an engineering science of robot foraging. In *Distributed Autonomous Robotic Systems 8*, pages 185–192. Springer-Verlag, Berlin/Heidelberg, Germany. p. 28, 30

A. Winfield, W. Liu, J. Nembrini, and A. Martinoli. (2008). Modelling a wireless connected swarm of mobile robots. *Swarm Intelligence*, 2:241–266. p. 19

A. T. Winfree. (1967). Biological rhythms and the behavior of populations of coupled oscillators. *Journal of Theoretical Biology*, 16(1):15 – 42. p. 25

Z

P. Zebrowski and R. T. Vaughan. (2005). Recharging robot teams: A tanker approach. In *Proceedings of the 12^{th} International Conference on Advanced Robotics*, pages 803–810. IEEE Press, Piscataway, NJ. p. 30

Artwork credits

The following images were reproduced in this thesis with the kind permission of their authors. The CC logos indicate that artwork is licensed under Creative Commons (see http://creativecommons.org/). The GNU logo indicates that artwork is licensed under GNU Free Documentation License (see http://www.gnu.org/licenses/old-licenses/fdl-1.2.html)

- **Figure 2.1 (a)**: Christoffer A. Rasmussen.
  Available at http://en.wikipedia.org/wiki/File:Fugle,_%C3%B8rns%C3%B8_073.jpg
- **Figure 2.1 (b)**: © Paul Martin.
  Available at http://www.flickr.com/photos/mahlathini23/2678450634/
- **Figure 2.1 (d)**: Barrett Lyon and Alex Adai.
  Available at http://www.opte.org/maps/
- **Figure 2.1 (e)**: Jim and Becca Wicks.
  Available at http://www.flickr.com/photos/68567841@N00/168870580
- **Figure 2.1 (f)**: Fir0002/Flagstaffotos.
  Available at http://en.wikipedia.org/wiki/File:Bee_swarm_feb08.jpg
- **Figure 2.2 (a)**: Mirko Junge.
  Available at http://de.wikipedia.org/wiki/Datei:Fliehkraftregler.jpg
- **Figure 2.3**: © Owen Holland, with kind permission of Alan Winfield.
  Available at http://www.ias.uwe.ac.uk/Robots/gwonline/gwonline.html
- **Figure 2.8 (a)**: © Robotic Harvesting, LLC.
  Available at http://www.roboticharvesting.com/index.html
- **Figure 2.8 (b)**: Joshua Marshall.
  Available at http://www.atlascopco.com/achieve/themes/Innovation/futuristicmining.asp
- **Figure 2.8 (c)**: Guzugi (English Wikipedia username).
  Available at http://en.wikipedia.org/wiki/File:Roomba3g.jpg
- **Figure 2.8 (d)**: Lance Cpl. Bobby J. Segovia.
  Available at http://en.wikipedia.org/wiki/File:IED_detonator.jpg
- **Figure 2.8 (e)**: David Axe.
  Available at http://www.flickr.com/photos/david_axe/372413687/
- **Figure 2.8 (f)**: Maas Digital LLC for Cornell University and NASA/JPL.
  Available at http://en.wikipedia.org/wiki/File:NASA_Mars_Rover.jpg
- **Figure 3.7 (a)**: Horst Frank, Michael Horvath.
  Available at http://en.wikipedia.org/wiki/File:HSV_color_solid_cylinder_alpha_lowgamma.png
- **Figure 3.7 (b)**: Horst Frank, Michael Horvath.
  Available at http://en.wikipedia.org/wiki/File:RGB_Cube_Show_lowgamma_cutout_b.png