Minimalist Coherent Swarming of Wireless Networked Autonomous Mobile Robots

Julien Nembrini

A thesis submitted in partial fulfilment of the
requirements of the University of the West of England, Bristol
for the degree of Doctor of Philosophy

Faculty of Computer, Engineering and Mathematical Sciences,
University of the West of England, Bristol

January 2005

Abstract

This thesis presents investigations on the possibilities of range-limited radio communication to self-organise a swarm of radio-connected robots. Firstly a decentralised algorithm relying only on local exchange of information to maintain the coherence of the communication network is presented. Then the potential of this algorithm to display complex behaviours is thoroughly studied, leading to the development of several variants displaying interesting global behaviours such as area coverage and connectivity control, taxis behaviour, group selection and segregation, and phase transition in axis formation. The proposed algorithms are tested in simulation and partially confirmed with real robot experiments. All global behaviours are emergent as the study constrains itself to using only a restricted range omni directional radio, a beacon sensor and obstacles avoiding sensors. Moreover, the fact that the algorithm and its variants rely only on local information makes the behaviours highly scalable with an increase in the number of robots. The author believes that the proposed approach has great potential for further developments.

Acknowledgments

This research has been possible firstly and above all with the help of Alan F.T. Winfield whose openness, kindness and faith in my work have created an understanding between us that do not suffer any noise but the inevitable continental-induced humorous slight “decalage”. May the distorted grammar of this sentence make the reader realise the extent of Alan’s contribution to this thesis.

I also thank Chris Melhuish for his insightful comments on the final draft. Thanks to all the critical lads in the lab. I am also highly indebted to all the Bristol friends that offered their hospitality. Cheers to Amelia, Vanessa, Madeleine, Jesse and Simon.

Par ailleurs, cette thèse n’aurait jamais commencé sans une suggestion de J.-B. Billeter et n’a aussi été possible que grâce à l’atelier et à la présence sympathique de Jean-Michel.

Je remercie Christelle pour ses visites inopinées qui m’ont rendu plus doux le sprint final.

Et tous les autres. juL.

Contents

Chapter 1

Introduction

"Wir sind die Robotern"

Kraftwerk.

The word *robot* comes from the Czech word for the verb "to work", *robota*. Indeed the motivation for the invention of robots has always been to construct machines in order to free mankind from tedious work or to carry out tasks in environments forbidden to man; hence the production line robotic arm or the deep sea diving robots. But it has been long realised that to carry out some of those tasks, the robot should have some degree of autonomy, if it was ever to be able to function in a normal dynamic environment. Indeed outside of its precise place on the production line a painting robotic arm loses its purpose completely. Analogously a deep sea diving robot is rather lost if the communication steering it happens to fail.

For these reasons, interest in the robotics community shifted from precise control algorithms and geometry of possible moves to the transformation of perception into meaningful action. Classical Artificial Intelligence tried to resolve this problem in constructing world models with their intrinsic logical laws. In this framework, perception was firstly carefully analysed according to the world model to deduce an action to execute. In a now famous experiment with the robot "shakey", this architecture failed completely to demonstrate an ability to cope with the dynamics of its own moves [Dreyfus, 1992]. A response to such failure came in the seminal work of Brooks [Brooks, 1986], introducing the concept of "behaviour-based robotics" that took inspiration from reflex behaviours, in which the purpose is to directly translate a perception into an action. He showed that in combining and coordinating several such simple reflexes (which he calls "behaviours"), an autonomous robot was able to display complex behaviour and could cope with the dynamics of the real world. The resulting behaviour *emerges* from the interaction of the reflexes. This had resonances beyond the field of robotics and launched a whole new area of research that is referred to as *new AI*.

Analogous to the search for autonomy in a single robot is the study of autonomous coordination in a group of robots. The same sort of problems arise in trying to make a multi-robot system perform a useful action. If the framework chosen is of a centralised coordination, the whole system is very dependent on the capacities of the coordinating unit, either in a time manner because of transmission delays or due to CPU congestion because of the exponentially increasing amount of data to process. The answer to such problems, in an analogy to the behaviour-based architecture, is to carefully design the autonomous behaviour of each robot using resources locally available such that the desired global behaviour emerges from the interaction of the behaviours of all robots. The robots *self-organise* without the help of an external coordinator. In doing so we step into the field of *collective* or *swarm robotics*. As the name itself suggests, a great deal of inspiration for this field comes from biological examples such as ant or termite colonies. These are animals with limited capacities that are collectively able to achieve amazing tasks.

Because of the current trend in miniaturisation, there has been in this field several studies on the minimal set of sensing abilities and rules needed to collectively achieve a given task [Melhuish et al., 1999a]. This is known as *minimalist robotics*. The research presented here fits into this framework as an investigation of the potential of crudely communicating robots to display self-organizing behaviour.

The beauty of an exchange of information lies in the fact that the informing party only lose the energy needed to make the other aware of its knowledge. Indeed, informing another does not imply one has to forget one's own knowledge. One may have to give away the potential position of power that the exclusivity of this knowledge could have represented, and this relates to the angst of the master being overthrown by its disciple, but the exchange of knowledge can also allow for the production of new knowledge that may benefit the informing party - a classical dilemma. Ideally, scientific progress relies on such open availability of new results within the community.

This thesis will investigate the potential of locally communicating robots to collectively achieve tasks beyond their intrinsic individual limitations. By the very act of sharing only local connectivity information, robots will *self-organise*, that is spontaneously coordinate themselves and reach a global equilibrium. Specialist robots sharing their knowledge with others are not considered, as in the master-disciple example, but instead homogeneous robots. Thus this work relates to the category of decentralised phenomena.

1.0.1 decentralised phenomena

When confronted with certain phenomena the first explanation one can usually think of involves a single cause; one rarely takes into account all the interactions at play. Although many things have been considered in this way, as centuries of efficient science can testify, the focus of recent decades of work on self-organising phenomena such as chemical reactions or stock market fluctuations has revealed the inadequacy of this paradigm to understand large parts of reality.

Indeed such phenomena resist the cause and effect framework. In such examples a change of state is not the result of one or several causes but instead the result of all interactions between elements of the system. No interaction is more significant; each has an impact on the future state of the system. This has opened an area of research on *decentralised* phenomena, also described as *distributed* in the literature. Its aim is to understand the dynamics at work in such systems.

1.0.2 constructionism

Yet the centralised way of thinking is still deeply ingrained. So much so that one would rather look for a cause than try to understand phenomena in a decentralised way. This is what Mitchell Resnick calls the *centralised mindset* [Resnick, 1999]. To overcome this mindset Resnick proposes *constructionism* as an alternative to the classical analytical methodology.

As an alternative to classical analysis through observation and prediction, constructionism proposes to understand and analyse phenomena by recreating - reconstructing - them. The aim is to synthesise [Braitenberg, 1984] the phenomenological process to capture its internal dynamic. Using constructionism helps to circumvent the centralised mindset as a centralised solution to a distributed problem is unlikely to reveal the same properties.

1.0.3 biological examples

Life presents beautiful examples of decentralised systems. The nervous system of animals with its impressive efficiency seems to belong to this category. Another is the embryonic development of multicellular individuals [Turing, 1952].

But the most impressive example, perhaps because it is possible to observe it directly, comes from the insect world. Social insect species behave in a highly developed collaborative way. For example bees and their direction-indicative waggle dance, ants with their foraging trails and the air-conditioning structure of termite nests show task achieving abilities that seem completely impossible considering only a single individual. Although the presence of a queen in the colony might at first glance make one think of a centralised system, synthesising approaches have demonstrated that local interactions among individuals can generate the complexity of behavior observed. A good introduction to this area of research can be found in [Bonabeau et al., 1999].

Even simpler forms of life such as bacteria seem to present decentralised characteristics; although their simplicity deceives us into thinking that they lack the ability to form a group. But biological evidence shows that bacteria are able to differentiate and share different tasks within a colony [Shapiro, 1988].

Technological advances in computational power have provided the constructionist approach with its first success in explaining biological processes and has given rise to a whole new area of research that is known as *Artificial Life* (AL). Its focus is to study "Life as it is" in providing a framework to test concepts of theoretical biology, as well as to investigate "Life as it could be" in transposing those concepts into different environments or ecosystems. An example is the tierra project from Ray [Ray, 1992] that studies the evolution of short programs in the computer memory environment competing for CPU processing time.

This interest has led to an attempt to characterise decentralised systems more formally.

1.0.4 principles and properties

Decentralised systems make wide use of the phenomenon of *emergence*. While its definition is still a matter of contention,it is best illustrated by the following sentence:

> the Whole is greater than the sum of its parts.

In other words the group considered as an entity is more effective than the contribution of its components in a specified task.

In order to achieve this phenomenon the components must interact with each other and it is the product of this interaction that allows for the shift in performance. In decentralised systems these interactions are supposed to be local: components only interact with nearby neighbours or with the local environment; the consequence being that any global behaviour has to emerge from this *localisation.*

As a result such systems present several properties, one of which is the *robustness* of the process. Indeed a decentralised process is not subject to complete breakdown if a component fails as decentralisation implies that it does not rely only on this special component to perform adequately.

Another property is *adaptability.* Often centralised systems are designed to be efficient in very precise environments. A slight change of the interactions can lead to the failure of the process. Decentralisation will present more of a graceful degradation in efficiency as small changes in the environment might affect the interactions without altering them completely.

Finally the localisation of interactions gives the opportunity for the process to be *scalable*; that is to be open to an increase in the number of components without loss of efficiency. A centralised system would instead reach a maximum when the central unit is not able to take more components into consideration either due to finite processing or communication resources.

1.0.5 collective robotics

The properties and potential of decentralised systems have raised considerable interest in the robotic community. As a result a whole new area of research has flourished in the last decade taking its roots in the new paradigm of behaviour based robotics.

The approach introduced by Brooks [Brooks, 1991] challenges the classical paradigm in artificial intelligence that tried to symbolically represent knowledge inside the agent's memory, leading to the intractability of the real world. In contrast it proposes a reactive approach arguing that the quest for artificial intelligence should be considering sensorimotor problems before higher-level intelligence. For a description of the problems and issues related see [Pfeifer and Scheier, 1999].

In their comprehensive review of the field of collective robotics, Cao et al. [Cao et al., 1995] note the motivations to study groups of robots:

- Tasks might be impossible for a single robot to accomplish, given that a robot, however complicated, will suffer from its limited range of action;
- Building and using simple robots might be more flexible and more fault-tolerant;
- The constructive and synthetic approach can possibly yield insights into fundamental problems in social and life sciences.

As a result roboticists have sought inspiration in biological metaphors to address their problems while biologists - mainly ethologists - have seen the opportunity to put their theories under experimental pressure.

In a more recent review of the field, Parker [Parker, 1996] identifies eight primary research topics: biological inspiration, communication architectures, localisation/mapping/exploration, cooperative object transport or manipulation, motion coordination, reconfigurable robots and learning.

Advances in miniaturisation of actuators, processors and sensors have raised minimalism as another interesting challenge in collective robotics. It aims to study the collective ability of large groups of robots, which are as simple as possible, to achieve non-trivial tasks [Holland and Melhuish, 1996].

1.1 The Research

This research extends the work of Winfield [Winfield, 2000] related to sensor networks, as well as the work of Melhuish [Melhuish et al., 1999b] that focused on secondary swarming. This study looks at the employment of mobile robots to combine sensing, locomotion and morphological adaptivity. Following this preliminary direction of research, it is proposed here to study a swarm of autonomous mobile robots communicating with limited-range radios. Instead of a physically connected system, this approach investigates virtually (wireless) connected robots. It is believed that this approach brings greater versatility in morphology and robustness to failures and noise. Also the hardware involved is readily available, allowing us to concentrate on the development of the algorithms. Of course the drawback lies in the fact that much effort has to be made to keep the robots together, *i.e.* to maintain the *coherence* of the swarm.

The work presented here takes inspiration from biological distributed systems such as insects or bacterial colonies to construct a robot collective that will show the desirable properties outlined below.

1.1.1 general aims

This research is primarily concerned with developing algorithms that are:

scalable: An increase in the number of agents should not be critical to the efficiency of the algorithm. Indeed the efficiency should instead show graceful (or no) degradation to such an increase. The study therefore relies on distributed solutions that make use of strictly local information. The robots are all identical and exchange information only about their neighbourhoods.

robust: The algorithm should provide the swarm with the ability to cope with the unreliability of the real world and show graceful degradation when confronted with noise and component failures. The last requirement is again a reason to seek distributed algorithms controlling homogeneous robots.

1.1.2 minimalism

The aim is to investigate the adaptability of a swarm of robots, in its shape and function, to changes in the environment. Following the framework of minimalist design from Melhuish [Melhuish et al., 1998], the focus will be on very limited robots able to communicate locally but lacking global knowledge of the environment. The only sensor information available, beside the basic obstacle avoiding infra-red sensors, are the beacon sensor and the radio communication. It is assumed that the communication hardware has a limited range, is omni-directional and the quality of the transmission is not optimal.

The aim is to keep the robots as simple as possible. It is believed that stability is reachable only with a limited range radio device and proximity sensors for avoidance. The key idea is that the limited range gives enough information on relative position. Such severely constrained conditions oblige us to tie together the act of communication with other behaviours of the robot. It is referred in [Støy, 2001a] as *situated communication.*

1.1.3 coherence

The study starts from the preliminary work of Winfield [Winfield, 2000] on gathering sensory data across an ad-hoc wireless connected network of mobile robots with range-limited capabilities. It was focused on randomly moving robots in a bounded space. The aim here is to extend to an unbounded space controlling the behaviour of the robots in order to form a dynamically connected stable swarm, *i.e.* a *coherent* swarm. It is believed that the robots' limited skills are sufficient to firstly let them maintain the network of communication in a single connected component and secondly to allow adaptive changes in morphology of the whole swarm as well as specialisation of single robots into different tasks.

The minimalist assumptions on hardware make the requirement on stability even more critical because of the fact that a disconnected robot will not be able to return to the swarm as there is no information available about relative or absolute position. But as the availability of such accurate information is still an area of research in robotics, it is worthwhile developing algorithms that circumvent the need for it.

1.1.4 networks

Due to the underlying motivation on sensory network applications, it is also sought to minimise the communication overhead that is necessary to achieve stability, thus keeping maximum bandwidth available for data gathering. Such a consideration has had a major impact on the methodology followed starting from the "minimal degree reactive variant" (see α-algorithm section 4.2.2) and then adding "state exchange" to reach a sufficiently stable solution (see β-algorithm section 4.12).

Since the aim is to build a network, much inspiration comes from simple graph theory concepts such as connectivity degree, cluster coefficient, etc [Bollobás, 1998, Watts, 1999].

1.2 Hypotheses

The hypotheses of this study are as follows:

major hypothesis

That it is possible, given the constraint of limited-range wireless communication, to achieve a stable connected swarm in an unbounded environment. We shall call this property *coherence*. (Chapter 4)

minor hypotheses

That it is possible for a coherent swarm to display taxis - collective motion towards a beacon - without loss of coherence. (Chapter 5)

That it is possible, in a coherent swarm, to achieve decentralised control of global morphology, for both homogeneous and heterogeneous swarms. (Chapter 4 and Chapter 6 respectively)

1.3 Structure of the Thesis

This thesis is organised as seven chapters, outlined in the following paragraphs:

chapter 2 Research Review

This chapter presents the key concepts and issues involved in this research as well as a survey of the related research to date.

chapter 3 Methodology

Chapter 3 describes the methodology used throughout the thesis to support the investigation of the hypotheses. More specifically it describes the simulation details, the real robot experimental details and the measures defined to assess the performance of the algorithms developed.

chapter 4 Coherence

This chapter presents the different algorithms that were developed to achieve coherence and an extensive inquiry into the behaviour of the most successful one: the β-algorithm. Investigated are the influences of swarm size, β parameter, and noise on the global behaviour. Experiments are conducted in simulation and with real robots. The potential of strictly homogeneous swarms is investigated and the possibility to tune the spread and generate position estimates is presented.

chapter 5 Taxis

Chapter 5 develops and investigates, in simulation, a truly emergent taxis behaviour relying on the dynamic differentiation between robots within the swarm.

chapter 6 Shape

This chapter investigates, in simulation, the potential of static and dynamic differentiation to actually control the global shape of the swarm.

chapter 7 Conclusion

The final chapter summarises the work accomplished and draws insights for new areas of research.

1.4 Contributions

This thesis presents the results of several experiments involving a wireless connected swarm. Working initially in simulation and partially confirming the β-algorithm with real robot experiments, it is claimed that the following advances have been made in the field of collective robotics:

.

coherence

1. It is shown that an algorithm relying only on first order information (number of neighbours) is not able to guarantee coherence with the sole help of communication (see α-algorithm section 4.2.2).

2. An algorithm (the β-algorithm) is developed that is able using second order information (neighbours' list of neighbours) to guarantee the coherence of the swarm (see β-algorithm section 4.12). This algorithm has the capacity to tune through a single parameter the connectivity of the swarm (see section 4.3). This connectivity measure represents the resilience of the dynamical network to loss of connections and failure of the nodes. This means that the algorithm is able to tune the robustness of the connected network (see section 3.5.1). Furthermore the same parameter has the power to control the area covered by the swarm (see section 4.4). It is also shown that the behaviours presented show good scalability with increasing swarm size and good resilience to increasing noise (see section 4.3.2).

3. An analytical upper (and lower bound) of the area covered is computed which makes use of locally available information to derive an interval within which the actual area stands (see

sections 3.5.2 and 8.1). This represents initial work towards the analytical modeling of the properties of the β-algorithm.

4. The β-algorithm is implemented on real robots and the capacity of the algorithm to achieve coherence is confirmed, despite strong compromises on the requirements assumed by the simulation experiments. Also a thorough investigation of the possible reasons for differences noted in robot performance is conducted (see section 4.3.5).

5. It is shown by the *localization*-β-algorithm that the exchange of information required by the β-algorithm is sufficient for a robot to approximate its global relative position (see section 4.5).

6. The β-algorithm is extended to the *taxis*-β-algorithm, to present a truly emergent coherent swarming taxis-behaviour relying on the differentiation between robots. In this algorithm, a single robot is not able to reach the goal, but instead the swarm movement results from the interaction of many robots' behaviour (see sections 5.2 and 5.4). The behaviour presented also shows good degradation to noise and swarm size increase (5.4.2).

7. The taxis behaviour developed demonstrates emergent swarm obstacle avoidance (section 5.4).

8. The ability of heterogeneities in the β threshold within the swarm running the β-algorithm (*concentric*-β-algorithm) and the ability of the *radial*-β-algorithm variant to generate spatial segregation of several groups of robots is demonstrated in section 6.1.

9. Based on the taxis algorithm, it is shown with the help of the *shape*-β-algorithm, that the swarm can react to an external cue and adapt its shape in relation to that cue (see section 6.2).

10. A framework to undergo artificial evolution to evolve robot controllers able to display global behaviours is defined in section 6.3. It contributes to developing work towards a methodology for the evolution of emergence in the context of multi-robot systems.

A diagram summarising the structure of the contribution made by this thesis can be found in figure 1.2. It shows the development path of the different algorithms, starting from the upper left corner, following the arrows. Figure 1.1 presents, starting from the basic algorithm, the family tree of the different algorithms developed.

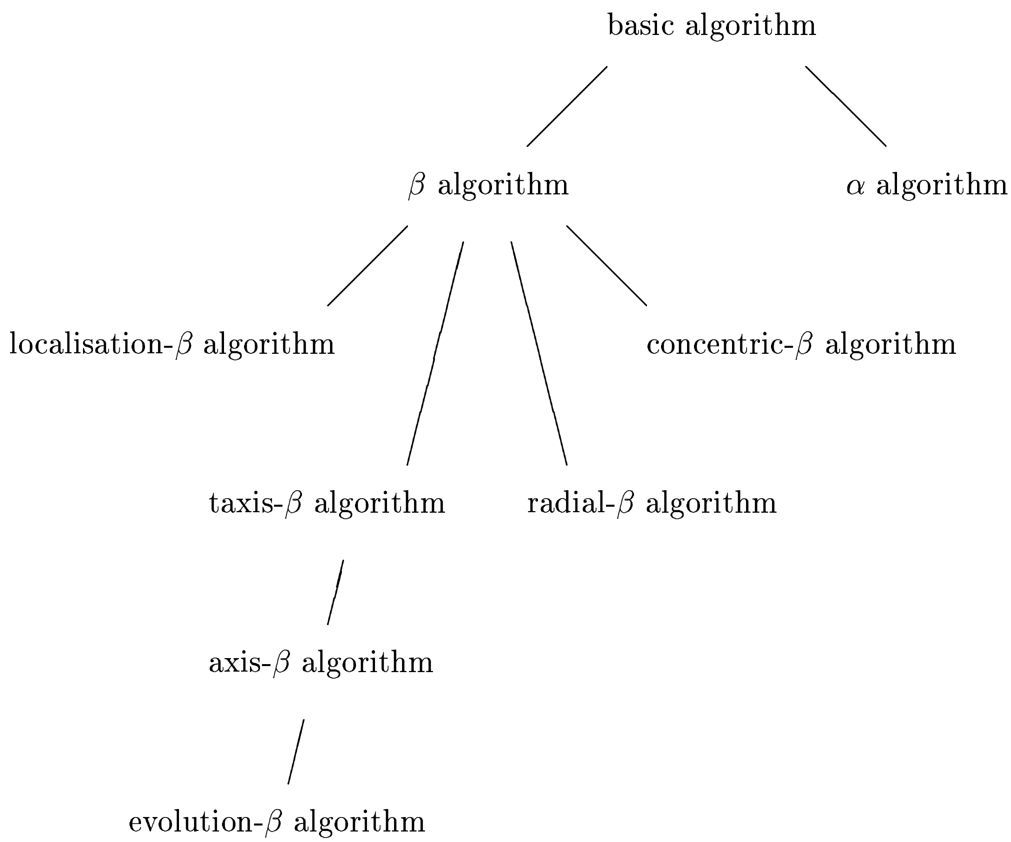


Figure 1.1: family tree of the different algorithms

Some elements of the results presented in this thesis have already been published in the Proceedings of the Edinburgh International Conference on the Simulation of Adaptive Behaviour in august 2002 [Nembrini et al., 2002]. Now that the claims to contributions have been made, let us begin with the core of this dissertation.

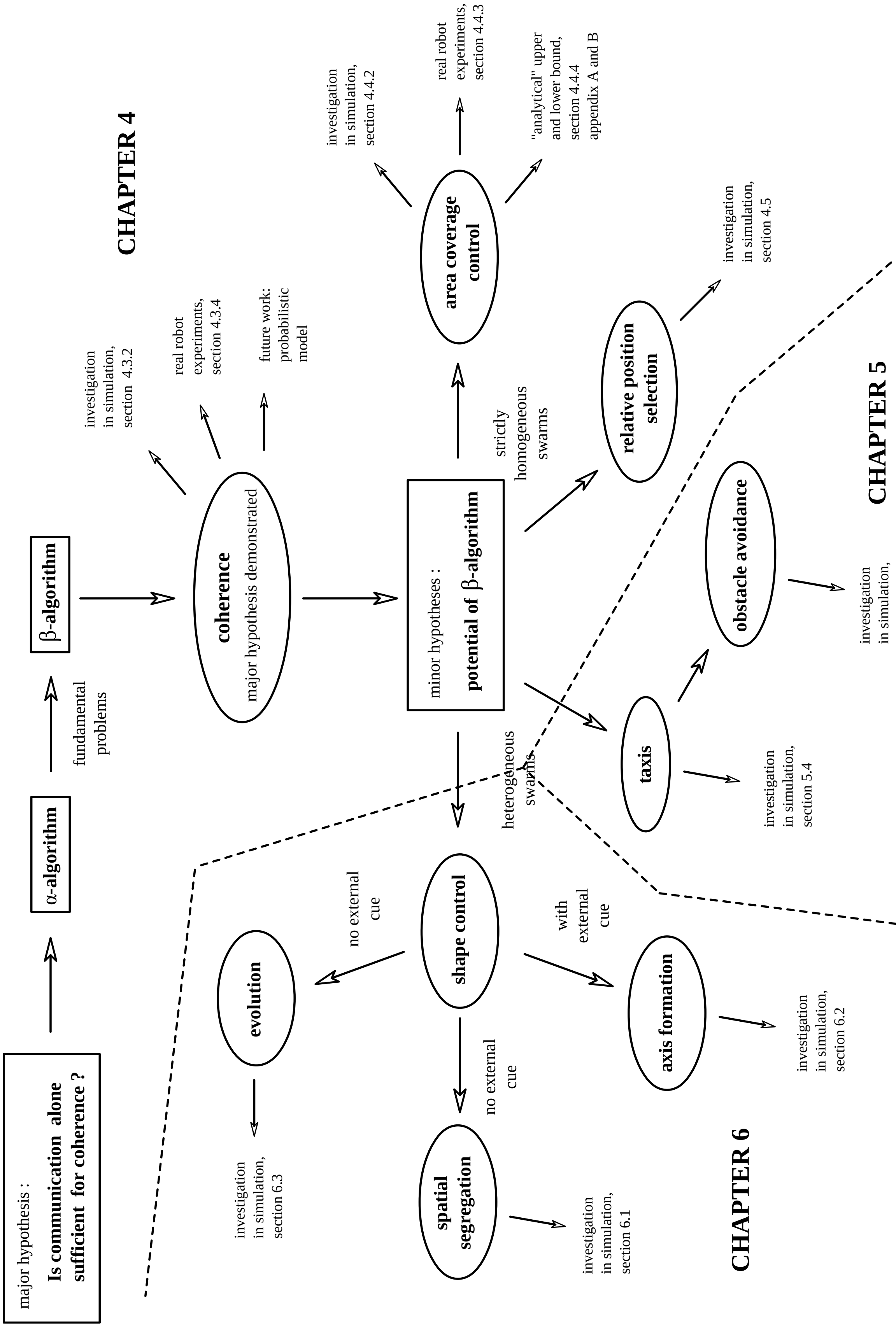


Figure 1.2: roadmap for the thesis

Chapter 2

Background and Literature

"It was the secrets of heaven and earth that I desired."

"Unhappy man! Do you share my madness? Have you drunk also of the intoxicating draught? hear me - let me reveal my tale and you will dash the cup from your lips!"

Mary Shelley's Frankenstein

Modern day robotics is in essence an interdisciplinary field. Investigations encompass questions from what is consciousness to the choice of motor control function or from the quest for true individual autonomy to the technical and algorithmic challenges of collective reconfiguration. Hence the involvement of many different disciplines such as psychology, computer science, neurobiology, ethology and mechanical engineering.

As a result this study owes its existence and grounds itself in results from a wide range of previous work and, conversely, its findings could be relevant to a wide range of research. In the following pages these results are presented in different sections although it should be understood that their inherent interdisciplinarity makes the boundaries of the sections somewhat permeable. Bearing this in mind, this chapter tries to take the reader from questions concerning the individual to those involving interactions between agents.

2.1 Intelligence

The definition of intelligence has always been controversial. Commonsense notions involve the concepts of problem solving, learning, memory, language, intuition, emotions or consciousness. But also the ability to survive in a complex world and perceptual or motor abilities are notions that are certainly relevant to a definition of intelligence. Because of the controversy, the term *cognition* will be used instead of intelligence, and we speak of an *intelligent* behaviour. Following Pfeiffer and Scheier, it is stated that an intelligent behaviour implies the ability of *"complying with existing rules and generating new behaviour"* [Pfeifer and Scheier, 1999].

The study of intelligent behaviour can be distinguished between the analytic and the synthetic approaches. The analytic approach involves the observation of an intelligent being in some carefully designed experiment and the results are then analysed to yield the confirmation or infirmation of the starting hypothesis. In the synthetic approach, the hypotheses stand as a design method that will be confirmed by the generation of the desired behaviour. A now classical example is the quizzical enquiry in synthetic psychology provided by Braitenberg in [Braitenberg, 1984]. In this approach understanding comes *by building*. In this respect it is related to the notion of *constructionism* discussed in chapter 3.

The synthetic approach is central to the field of *embodied cognitive science*. It aims to study cognition within a body rather than the potential of a disembodied mind, as was previously attempted in the field of Artificial Intelligence.

For many years the focus has been on trying to confer to a computer the same formal logic and abstract representations that we experience as conscious humans. In a well known experiment with the robot "shakey", such an attempt showed limitations in controlling a robot [Dreyfus, 1992]. Because of the complexity of the real world the representation of all relevant features made the robot engaged in a simple navigation task far too slow to choose which action to perform, at least with the computing power available at the time. This failure raised the question of whether the use of abstract representation and formal logic was needed for a task such as obstacle avoidance. Examples from nature reveal that most animals are very successful in doing such tasks, although they are not yet believed to master formal logic.

Lessons from nature suggests that the ability to cope with a highly dynamical environment lies in the coupling between cognitive and motor skills. The purpose of embodied cognitive science is to build artifacts to investigate this coupling and its relationship to the environment. Hence the importance of the key concepts of *autonomy, embodiment and situatedness* that are treated in the next section.

2.2 Autonomy, Embodiment and Situatedness

2.2.1 autonomy

The definition of intelligent behaviour as, inter alia, the capacity for generating new behaviour implies *autonomy*, that is freedom from external control. As stressed in [Pfeifer and Scheier, 1999], an agent is normally not fully autonomous or not, but instead achieves a certain degree of autonomy. Moreover complete autonomy is impossible as an agent always depends on external factors that are beyond its control.

This dependency can be divided into dependencies on the environment, such as organisms in need of food, and dependencies on other agents, such as robots to the roboticist for instance. An autonomous agent is always dependent upon the environment either for its energy supply or if only for the cues the environment provides to allow for proper functioning. Work in behaviour-based robotics concentrates largely on the latter, trying to define ways of choosing actions that are relevant to the situation without the help of external control, either human-like or computer-like. The well-known example is the subsumption architecture that divides behavioural modules into layers of priority [Brooks, 1986]. In this study the robots are required to be autonomous in this sense. As soon as the robot is running it has to choose its own actions; these can indeed be altered by other robots, but only altered and not controlled.

An increase of autonomy can be achieved through making the agent capable of *learning*, that is able to modify its behaviour according to past events. This plasticity of behaviour has the potential to decrease the dependency of the agent on the environment making it able to cope with environmental dynamicity, as well as to decrease the dependency of the agent to others. Examples of learning in the multi-robot case can be found in [Tangamchit et al., 2000], [Parker and Touzet, 2000], [Iijima et al., 2000], [Mataric, 1994]. As the present study is interested in the potential of a tight set of constraints (*i.e.* minimal sensing), learning is indeed a possibility for widening these potentialities.

However, in order to narrow down the scope of the research to be able to bring interesting insights, the robots are here kept only reactive and are not allowed to learn. However we believe that the structures we present have great potential for further research, including learning.

Another way to decrease the control the roboticist has on the design and therefore increase the autonomy, is to allow the robots to evolve. In this case the designer defines one or several dimensions that the agent's behaviour can span, optimising by trial and error. As this possibility has been chosen to explore the potential defined by the set of constraints, a section of the present chapter is devoted to the discussion of *artificial evolution* (see section 2.6.1).

2.2.2 situatedness

As many other key notions in the field, the concept of *situatedness* lacks a common definition. Following [Pfeifer and Scheier, 1999], we say of an agent that it is situated *"if it can acquire information about the current situation through its sensors in interaction with the environment"*, or alternatively if it is *"always already in some situation rather than observing from outside"* [Harvey, 2000].

2.2.3 embodiment

Another key concept is *embodiment*. In [Harvey, 2000], an embodied agent is *"a perceiving body, rather than a disembodied intelligence that happens to have sensors"*. Pfeifer and Scheier refer to *"autonomous agents as real physical agents; in other words they are embodied"* [Pfeifer and Scheier, 1999]. The importance of this reality is emphasised in the example of passive dynamic walkers: walking robots that don't require any electronic control as their behaviour relies only on the laws of physics [McGeer, 1990]. In Brooks' approach, the idea is that intelligence can emerge from this embodiment [Brooks, 1986]. However it is interesting to note that the modern robots being used in the laboratories today (such as *khepera* or *linuxbot* in this case) have - from a hardware point of view - the same characteristics as their classical AI predecessors (central CPU, I/O, sensors, actuators). True embodiment might well stand beyond this limitation, in robots with distributed control where actuators have local hardware control that can be synchronised at a higher level.

Interestingly Quick blurs the difference between embodiment and situatedness by stating his definition as the possibility of interaction between the agent and its environment [Quick et al., 1999a].

More formally:

A system X is embodied in an environment E if perturbatory channels exist between the two. That is, X is embodied in E if for every time t at which both X and E exist, some subset of E's possible states have the capacity to perturb X's state, and some subset of X's possible states have the capacity to perturb E's state.

This definition emphasizes the fact that an agent cannot be considered without its environment. Indeed its existence results from it being embodied within that environment. The definition allows consideration of different degrees of embodiment through the observation of the "bandwidth" of the perturbatory channels. A preliminary attempt is presented in [Nehaniv et al., 2003].

It has to be noted that embodiment does not imply situatedness: as an example take a totally remotely controlled robot. It is embodied but has no means to alter its behaviour according to its environment and is therefore non-situated.

2.3 Synthesis and Complex Systems

In his well known book *Vehicles - Experiments in synthetic psychology*, Braitenberg explored the potential of very simple vehicles to exhibit "intelligent" behaviour [Braitenberg, 1984]. By his *"thought experiments"*, he showed the potential power of explanation of an approach of building models, either real or virtual, that are able to reproduce a certain behaviour; and it can sometimes lead to surprising simplicity.

As stated in [Bonabeau and Theraulaz, 1994], any science having to deal with complex systems will find the synthetic approach attractive. A *complex system*, in the most commonly shared definition, is *"a network of interacting objects, agents, elements or processes that exhibit a dynamic, aggregate behaviour"*. Moreover, a complex system has *"many degrees of freedom that strongly interact with each other"*. It shows *"organised complexity as opposed to organised simplicity and disorganised complexity"* [Bonabeau and Theraulaz, 1994]. Living systems undoubtedly fall into this definition.

As a result, complex systems are very difficult to deal with analytically. Therefore *synthesis*, a bottom-up approach based on building artifacts either real or simulated, is a way to bypass this intractability and explore the behavioural space of complex systems. Inherent to this methodology are the following questions: if we use synthesis to reproduce the behaviour of a complex system, what are the criteria of reproduction ? Also to what extent is a phenomenon explained by the description of a model reproducing its phenomenological characteristics ? As most of epistemology has until today focused on analysis rather than synthesis, these remain open questions.

A synthetic methodology seeks to *understand by building*. In this sense, it is strongly related to *constructionism* as will be developed in chapter 3. Constructionism is primarily concerned with learning; learning by building. As doing science is essentially about learning (although to a certain extent unsupervised), a synthetic methodology is a constructionist approach to doing science.

2.4 Emergence and Self-Organisation

2.4.1 emergence

Pervasive and central to the fields of embodied cognitive intelligence, artificial life and behaviour-based robotics is the idea of *emergence*. Again there is no commonly accepted definition and in [Pfeifer and Scheier, 1999] three are articulated:

- A surprising property of a system that is not fully understood [Ronald and Sipper, 2001].
- A property of a system not contained in any one of its parts. It requires many components whose behaviour is based on local rules.
- Behaviour that arises from the agent-environment interaction. Whenever several, independent processes interact to produce a particular behaviour and the environment plays a significant role.

It is also illustrated by the sentence *the whole is more than the sum of its parts*. The interactions in a system displaying emergent phenomena are *non-linear* which implies that the *"overall behaviour cannot be obtained by summing the behaviours of the isolated components. There are regularities in the system behaviour that are not revealed by direct inspection of the laws satisfied by the components"* [Holland, 1998]. It has to be noted that emergence as well as self-organisation can happen through interactions between agents or through interactions of components within an agent. In this sense the role of the observer is crucial.

In [Bonabeau and Theraulaz, 1994] another definition is attempted: *"a process through which entirely new behaviours appear, whose properties cannot be derived from a given model of how the system behaves, so that another model has to be built in order to deal with these new behaviours"*. Here the role of levels of observation is made clear, as will be further developed in the next section.

2.4.2 self-organisation

Self-organisation was originally defined in the context of physics and chemistry to describe the emergence of macroscopic patterns out of processes and interactions defined at the microscopic level. It is related to chaos theory and dissipative systems far from equilibrium [Nicolis and Prigogine, 1977, Prigogine, 1994]. In [Pfeifer and Scheier, 1999] it is referred as *"a process by which patterns are formed in systems containing a large number of elements"*.

In [Bonabeau et al., 1999] self-organisation is extended to describe the interactions in social insects and defined as follow: *"Self-organisation is a set of dynamical mechanisms whereby structures appear at the global level of a system from interactions among its lower level components. The rules specifying the interactions among the system's constituent units are executed on the basis of purely local information, without reference to the global pattern, which is an emergent property of the system rather than a property imposed upon the system by an external ordering influence."* In this definition the intertwining between the notions of emergence and self-organisation appears.

We will, in the remainder of this study, use the notion of emergence as defined in [Bonabeau and Theraulaz, 1994] (see citation above) and relate to emergence as *a property describing a phenomenon or a behaviour* and to self-organisation as *the process in which this phenomenon or behaviour is produced.*

Bonabeau et al. mention four basic ingredients that are found in self-organising systems:

- *Positive feedback* that consists of simple behavioural rules that promote the creation of patterns.
- *Negative feedback* that counterbalances positive feedback and helps to stabilise the collective pattern.
- *Amplification of fluctuations*, randomness and errors are crucial to enable the discovery of new solutions. Patterns are created by amplification of this randomness

- *Multiple interactions* interactions can increase the non-linearity of the system.

They also characterise the properties of a self-organised phenomenon [Bonabeau et al., 1999]:

- The creation of *spatio-temporal structures* in an initially homogeneous medium.
- The possible coexistence of several stable states (*multistability*). Because structure emerges by amplification of random deviations, any such deviations can be amplified, and the system converges to one among several possible.
- The existence of parametrically determined *bifurcations*.

An interesting insight into self-organisation can be found in [Gershenson and Heylighen, 2003]. Starting their study with the assumption that a way to characterise self-organisation can be by using entropies as a measure of the increase of order, they define counter-examples showing that systems can show *"different degrees of order or "entropies" for different models or levels of observation of the same entity"*. In other words the way we define the state-space is crucial for detecting increase or decrease of "order". Therefore the role of the observer is again crucial to determine how the system is to be observed. It is only relative to this choice that a system can be considered self-organising. In our study we want to investigate self-organisation of robots therefore the level we consider is the one we can act on, namely the control of the robots individually.

There have been several attempts to formalise emergence and self-organisation [Holland, 1998]. But these attempts, although they give insights into the essence of the problem, usually lack any predictive power which, in the end, is the whole purpose of formalism. The formalisms developed are more descriptive than result-oriented. An example of such a formalism is used to describe the evolutionary framework presented in section 6.3.1. It cannot be used to predict the results of the evolution as there are functions that cannot be expressed in a formalised way.

2.4.3 the centralised mindset

The notions of emergence and self-organisation present explanations for phenomena that are based on interactions of components. The fact that a system's emergent behaviour is not specified in any of its parts, but typically arises from those interactions, makes it difficult to explain these phenomena by well-defined cause and effect.

Nevertheless, in his book, Resnick notes that when confronted by a given phenomenon, people will tend to look for THE - if possible unique - cause for explaining it. In many cases, this way of explaining phenomena is absolutely valid and, maybe because of this validity, it seems that people tend to generalise and look for an external cause for every phenomenon considered [Resnick, 1999]. This is what Resnick calls the *centralised mindset.*

In his investigations at the boundary between education and science, Resnick stresses that:

"People seem to have a strong preference for centralisation in almost everything they think and do. People tend to look for THE *cause,* THE *reason,* THE *driving force,* THE *deciding factor. When people observe patterns and structures in the world (...), they often assume centralised causes where none exist"* (emphasis in original).

He explores several explanations for this observation, the first being that many phenomena are indeed organised by a central designer or cause. Secondly most people participate in social systems *"where power and authority are very centralised (...) [and] these hierarchical systems serve as strong models"*. Thirdly our conception of the self as *"one entity in charge"* makes it natural that one should expect most systems to involve a central actor.

To actually design systems displaying emergent behaviour is difficult. Firstly, because of the lack of a formal definition there are no rule-of-thumb design guidelines. Secondly because of the centralised mindset, thinking in a distributed way, as would be needed for emergent phenomena, is not a mature approach.

2.5 Swarm Intelligence

Swarm intelligence refers to the capability of unintelligent agents to collectively perform tasks that we perceive as requiring intelligence [Bonabeau et al., 1999]. Classical examples of swarm behaviour are ant foraging tasks or collective nest building. In these examples, it has been shown that solutions exist that do not require the use of direct communication but instead relying on stigmergy and self-organisation (see sections 2.10.1 and 2.4).

The fact that self-organisation may be the principle underlying the organisation of social insects has potentially far reaching implications for engineering and computer science. Indeed the problems solved by the insects' colonies relate to problems such as sorting or distributed coordination. The beauty of the solutions found by evolution of ants and termites is that they are very flexible and robust to allow for adaptation and survival. Characteristics that are highly sought in engineering solutions. Also the ants and termites have limited cognitive capabilities which allow for synthesis of artifacts able to mimic their behaviour. As stated in [Bonabeau and Theraulaz, 1994], *"central to the idea of swarm intelligence is the coexistence of individual simplicity and collective complexity"*.

Because of these characteristics and despite the maintenance difficulties that arise when dealing with large groups of robots, there is growing interest into *collective robotics* or *swarm-based robotics*. Although much of this work is only loosely related to social insects, some try to explicitly mimic the behaviour found in insect colonies. Real robot implementations include Beckers et al. [Beckers et al., 1994] who implemented distributed clustering of pucks and Melhuish et al. [Melhuish et al., 1999a] who developed rules for two-color sorting inside a cluster and recently three-color sorting by adding sensory skills [Melhuish, 1999b]. Melhuish also implemented an example of cooperative wall building [Melhuish et al., 1998]. Another example is [Krieger and Billeter, 2000] where individual activation thresholds are able to solve the task allocation problem in foraging robots. We dedicate a whole section of the present chapter to the more general field of collective robotics (see section 2.8)

This study, though its interest is in decentralised solutions based on local communication, is part of the framework of swarm-based robotics, but its aim is decidedly not to reproduce any behaviour observed in nature. Actually, as will be presented later, it is only subsequently that the behaviours generated in this work were associated with their counterparts in nature.

2.6 Artificial Life

The concepts of self-organisation, emergence and the synthetic methodology have already been mentioned. Now a field that strongly relies on such notions is the field of *artificial life* (AL). In the introduction of the proceedings of the first workshop on the topic, Chris Langton states his definition as:

"Artificial Life is the study of man-made systems that exhibit behaviours characteristic of natural systems. It complements the traditional biological sciences concerned with the analysis of living organisms by attempting to synthesise life-like behaviours within computer and other artificial media.

By extending this empirical foundation upon which biology is based beyond carbon-chain life that has evolved on Earth, Artificial Life can contribute to theoretical biology by locating life-as-we-know-it within the larger picture of life-as-it-could-be" [Langton, 1989].

In their critique of the field, Bonabeau and Theraulaz state another definition:

"[Artificial life is] a general method consisting in generating at a macroscopic level, from microscopic, generally simple, interacting components, behaviours that are interpretable as life-like" [Bonabeau and Theraulaz, 1994].

They remind us that most work in AL relies on the assumption that *"synthesis is the most appropriate approach to the study of complex systems in general and of living complex systems in particular"*. Their definition strongly emphasises the importance of ways of achieving emergence to generate complex behaviours. In their view, the AL methodology is a *"completely reductionist one, because it is aimed at explaining high-level behaviours from low-level causes"*. But they distinguish between ontological, methodological and epistemological reductionism, stating that AL is only methodologically reductionist.

Both definitions are rather vague in the fact that the interpretation of a behaviour as life-like or being characteristic of natural systems is strongly dependent on what criteria are used to test those characteristics.

Two key notions in Artificial Life are collective behaviour and artificial evolution. The first has been introduced in section 2.5 and will be further developed in section 2.8. The latter is the subject of the next section.

2.6.1 artificial evolution

Because of the intractability of real experiments in natural evolution, one of the key topics in AL is the study of *artificial evolution*. It was originally developed as an optimising tool aiming at searching for the optimal solution in a given parameter space [Bäck et al., 1991], and has been used as such to find, for example, particular rules for cellular automata [Sipper, 1997]. Taking advantage of recent developments in realistic simulation, artificial evolution has moved towards a synthetic study of the dynamics of evolution. Examples range from evolution of programs [Miller, 2003], formation of virtual cells [Madina et al., 2003], to evolution of whole ecosystems [Penn, 2003].

mechanism

Artificial evolution works as follow: the system is composed of an environment and a population of *genomes*. Each *generation* follows the three steps below (for a more thorough description, see [Pfeifer and Scheier, 1999], chapter 8):

- *Development*: From each genome is derived an individual (*phenotype*), either directly by some description function or through a development phase.
- *Selection*: Each individual is tested in the environment and according to this test is assigned a *fitness*.
- *Reproduction*: A new population is created from the previous one by mating two individuals (*cross-over*) and/or by random changes in the genome (*mutation*).

The result is an individual that optimises the fitness function, provided a "sufficient" number of generations and a system genotype-phenotype that has the potential to do so.

There are many variants depending on the coding chosen to describe an individual, to assign fitness or to choose the individuals allowed to mate. A critique of what can and cannot be achieved with a genetic algorithm, related to engineering, and comparisons with other architectures also implementing artificial evolution such as *Genetic Programming (GP)* or *Evolutionary Strategies (ES)*, can be found in [Toffoli, 2000].

embodied evolution

Because of the length of the evolutionary process, most of the work in evolutionary robotics has relied on simulation to find optimal solutions and then transfered these results to real robots [Baldassare et al., 2003]. Examples can be found in [Jakobi et al., 1995], [Quinn et al., 2002b], [Nolfi and Floreano, 2000]. Because of the discrepancies between the real world and the simulated model, this transfer can be difficult [Jakobi et al., 1995] (see section 2.9).

An interesting alternative to such methodology is the framework of *embodied evolution* presented by Watson et al. [Watson et al., 2002], [Watson et al., 1999]. The idea is to use many robots as the population of genome and let them interact and exchange genes in performing the required task. More successful robots have better chances to propagate their genes than less fit ones and the system eventually converges towards a solution. Other examples can be found in [Nehmzow, 2001, Kubik, 2003]

Obviously the application of such a framework strongly depends on the task being studied, but we believe that our multi-robot set-up could be a platform for such investigations.

morphology

Most of the work in evolutionary robotics has concentrated on the control algorithms of robots with fixed morphology. This of course is not the case for natural evolution and such a restriction may obviously prevent solutions that require a change in morphology [Pfeifer, 2000]. Examples can be found in [Floreano and Nolfi, 1998] or [Pollack et al., 2000]. But a striking example, built on Sims work [Sims, 1994], shows the result of the transfer from a previously simulated evolution of morphological structure for locomotion, into a real robot that shows qualitatively the same behaviour [Lipson and Pollack, 2000] and later [Hornby et al., 2001]. Further work on *virtual embodied evolution* can be found in the work of Bongard et al. [Bongard and Pfeifer, 2001], [Bongard, 2002], [Bongard and Paul, 2000], [Frutiger et al., 2002].

The experimental set-up for this study presents a swarm of real and simulated robots that behaves as a whole and chapter 6 investigates whether the distributed control of global shape through the help of communication alone can be evolved.

genetic regulatory networks

As a matter of fact most evolutionary algorithms lack a development phase translating the genotype into a phenotype. In such cases the complexity of the individual is completely contained in the genome.Indeed the function that maps the genome into a phenotype can at most be iterative, as in [Sims, 1994].

If you allow instead the individual to go through a growth phase controlled by the genome, before it is actually tested for fitness, then the complexity of the genome can be greatly reduced by making use of mechanisms of self-organisation during this growth process. Examples can be found in [Bongard and Pfeifer, 2001], [Bongard, 2002]. In these cases, the individual's genome is a model of a biological DNA string. Genes are located on this string and expressed according to the *operon model* of Jacob and Monod [Jacob and Monod, 1961]. It is a system in which genes and their products directly or indirectly regulate one another's activities. Such a genotype forms a *self-regulatory network* as described in [Kauffman, 1989].

A self-regulatory network is a directed boolean graph in which each node typically has K entrant edges and any number of exiting edges. Each edge has a boolean value that is updated by a boolean function of the values of the nodes at the origin of the K entrant edges. This is an idealisation of the operon model where edges are proteins regulating the expression of the genes represented by the nodes. Self-regulatory networks are also known as *random boolean networks* (RBN).

In his study of the behaviour of such networks, Kauffman found that complex behaviour arises when $K = 2$ with boolean *canalysing* functions. A boolean function is canalysing if one of the input values constrains the output completely. In the example of table 2.1, the value 0 for input 1 completely determines the output.

input 1	input 2	output
0	0	1
0	1	1
1	0	0
1	1	1

Figure 2.1: example of canalysing boolean function

In his studies, Kauffman concentrated on synchronous random boolean networks (RBN). The update of the nodes' states occurs simultaneously. In our case we are dealing with asynchronous update as robots typically desynchronise and the task of synchronising the robot would be counter to our aim of minimalism. In their study of asynchronous RBN, Mesot and Teuscher show that the behaviour of such systems is of a quite different nature [Mesot and Teuscher, 2003].

In order to cope with asynchronicity and achieve smoother behaviour, continuous values between 0 and 1 are considered in the "gene-protein" models called *artificial genetic regulatory networks.* A node in the network represents a *"gene"* that has an *operator site* and a *expression site.* The expression site specifies which *"protein"* is expressed and to what *concentration.* The operator site specifies which protein has an influence on the expression of the gene - inhibitory or enhancing - and in which concentration thresholds; indeed this is the self-regulation. In this model, instead of influencing directly the downstream nodes, the result of the boolean function changes the concentrations that will then act on the receptive downstream nodes.

By binding the value of the nodes to some phenotypic variable, such as for instance growth of a body segment or growth of neural control as in [Bongard, 2002], the complex dynamics of the genetic regulatory network can generate complex morphology and behaviour of an artifact.

artificial embryogeny

The development phase of the genome can be done through direct mapping or through a development phase. In the first case the complexity of the genotype grows linearly with the complexity of the resulting artificial organism. But the example of human DNA presents a proof of existence of an extremely complex organism (100 trillion neural connections in the brain) in which genotype complexity is several orders of magnitude lower (only about 30 thousand active genes).

Such a difference of magnitude is only possible through gene reuse. A methodological approach to allow for such reuse is to let the artificial organism go through a development phase. The individual starts from a small initial structure and its genotype acts more as local rules that guide growth, than an overall building plan.

Artificial embryogeny, as defined in [Stanley and Miikkulainen, 2003], encompass studies that investigate different approaches to implement this development phase. Stanley and Miikkulainen divide the field into *grammatical* and *cell chemistry* approaches, although they stress that this is only an apparent difference.

The first group involves the use of an algebraic description of the organism and the development phase consists of rewriting rules of the starting description. Examples consist mainly in variants of Lindenmayer's *L-systems* [Lindenmayer, 1968] and include [Sims, 1994, Lipson and Pollack, 2000].

The second group builds on the initial paper of Alan Turing [Turing, 1952] (see section 2.12.3) that describes his intuitions on the existence of diffusing *morphogens* and their potential to generate patterns. Studies from this group try to model the diffusion of such morphogens in a cartesian space and their action on the growth of the organism. Continuous genetic self-regulatory networks provide a model for the production of such morphogens (see chapter 4). The examples presented below belongs to this group.

In the work of Bongard [Bongard and Pfeifer, 2001], [Bongard, 2002], a virtual artifact is firstly grown with its morphology and neural control in the development phase, and then tested in the task of pushing a box in a physically realistic environment. Evolution eventually resulted in individuals that were able to start pushing the box and then that are more and more successful at it. In this case, the optimal genomes are less complex than the resulting artifact's morphology and behaviour. The drawback is in the fact that the framework chosen - body segments that grow - does not allow for a real robotic transfer.

In [Streichert et al., 2003], a model of cell growth according to a self-regulatory network is evolved and shows good ability to limit the growth and undergo self-repair but they also mention that the morphology of their artifacts depends highly on small brownian fluctuations. An application of genetic regulatory networks (see section 6.3) in robotics can be found in [Quick et al., 1999b], although the implementation suffers from high bias at the beginning.

In [Bentley, 2003], such a genetic network is used to control a robot. They were able to evolve, in simulation, a path through obstacles. The transfer performed well, not least because the robot chosen for the experiment used high-level instructions for movement.

A biologically more plausible model is introduced by Banzhaf [Banzhaf, 2003]. Instead of considering specific "protein" matching for regulation, he considers complementarity matching, using bit strings and the XOR operation. The result corresponds to continuity in the matching process. The behaviour of such a model shows small time fluctuations to bit changes in matching, making it of particular interest for artificial embryogeny.

The study presented in chapter 6 is an enquiry into the potential of such self-regulatory networks to cope with noise, dynamicity and randomness. The task is to break the symmetry to keep a linear shape in an otherwise amorphous swarm. Because of the dynamic nature of our experimental set-up, we actually concentrate in this case on the development phase. Although the robots we use will have to be improved to be able to implement the algorithm using self-regulatory genetic networks, we can realistically hope for a successful transfer of evolved characteristics into a real robot implementation, unlike in [Bongard, 2002].

2.7 Robotics

According to the dictionary, a *robot* (from the Czech *robota*, compulsory labor) is[1]:

1. a machine that looks like a human being and performs the various complex acts (as walking and talking) of a human being,

2. a mechanism guided by automatic control.

Taking the second meaning of the word, the field of robotics can be divided into two groups: robots that are constructed to perform a fixed task in a fixed environment and robots that are expected to cope flexibly with a changing one. Robots belonging to the first group are usually found in production lines and their design raises problems related to mathematics, engineering and control; the best known example being the robotic arm (multi-axis manipulator). In this case, the environment is carefully designed and constrained to enable the robot to carry on its task without being disturbed by unexpected events.

This possibility to constrain the environment makes the behaviour of the robot typically have, for every possible configuration of the environment, a sensible solution. The effort is then concentrated into recognising these configurations and matching the actual movement to the solution dictated by the controller. This is mainly done to reduce the possibility of errors, either at the sensory level or at the actuating level.

Because of their inherent purpose, the robots belonging to the second group add to the problems of sensible movement within a known environment, the problems of sensing the changes in this environment and acting "sensibly" according to these changes. Because of this difference, these robots cannot function within the same paradigm as the ones belonging to the first group. Indeed the ability to adapt to changing situations contradicts the possibility to map a sensible move to every possible configuration. Furthermore, the understandable interest in designing robot devices able to display autonomous mobile behaviours makes the interconnection of the problems of weight, energy and computational power of great relevance.

The first approach that was attempted relied on building an abstract world model and used logical inference to create new "knowledge" about the sensed world. However, this approach suffered from

[1] Merriam-Webster Collegiate Dictionary edition of 1999

the rapid intractability of the real world. The conjunction of the combinatorial explosion of logical rules with the difficulty of correspondence between the abstract model and the real world made the approach fail to convince that it could lead to an autonomous system. As an example, even in drastically simplified worlds, the dynamicity induced by the movement of the robot itself was already a change too difficult to cope with [Pfeifer and Scheier, 1999].

However, Nature provides us with countless examples that have been used lately as an inspiration to design successful devices, such as the approach of *behaviour-based robotics* (see the next section). Additionally the possibility of the robots to actually embody the fruit of theoretical investigations has widened the scope of interest and has involved researchers from psychology, neuroscience or biology, among others. It is now referred as *bio-inspired robotics* [Parker, 2000, Vaughan et al., 2000a].

2.7.1 behaviour-based robotics

The traditional approach to building robots relies on functional decomposition: sensing and proprioception[2] is performed first. The information coming from the environment is then processed and integrated in an abstract model, the *world model.* According to this model an action is planned that will eventually be transformed into an acting phase and the cycle loops. This cycle is referred as SMPA or sense-model-plan-action [Pfeifer and Scheier, 1999].

As mentioned earlier, this approach has demonstrated low performance, especially because of the evident simplifications that take place between the S and M phases that is later translated into potential problems between the P and A phase. Also the possible complexity of real situations can rapidly make the P phase computationally intractable.

The failure of real robot experiments following this framework, such as *"shakey"* and others [Dreyfus, 1992], suggested the inability of this approach to cope with the uncertainties of the real world (although the considerable progress achieved since in computing power calls for a new study of this hypothesis). To analyse a situation and then act on the basis of this analysis is actually computationally very expensive and rapidly becomes intractable if the environment is not highly constrained (such as in production lines). To tackle this restriction on autonomy, Brooks proposed to study robots that instead of analysing, simply *react* to changes in their sensorial environment, *i.e.* to achieve a direct coupling between sensors and actuators. The problem then becomes the

[2] the set of perceptions that the agent has of its own body

coordination of these reactions such that meaningful behaviour results [Brooks, 1986]. Note that although the robot might benefit from this behaviour, the meaning stands actually only in the observer's eye.

More precisely, the behaviour-based approach, with the well known subsumption architecture [Brooks, 1986], builds control architectures by incrementally adding *behaviours*, each of these running in parallel and being responsible for a restricted part of the overall behaviour of the robot. These behaviours are organised in successive *layers* functioning relatively independently and often representing the levels of priority for the safety of the robot. This layered approach provides a framework for the coordination of such parallel behaviours to display meaningful global behaviours.

This proposition has shown great potential and its early success [Brooks, 1986, Maes, 1991] has started a whole field of research on autonomous robots that is commonly called *behaviour-based robotics* [Arkin, 1998, Maes, 1993]. It involves the contemporaneous enquiry of software and hardware problems such as definitions of control architectures [Albiez et al., 2002], design and use of sensors [Lungarella et al., 2002] or actuators [Yoshida et al., 2000], control of several robots (section 2.8), etc. As such, behaviour-based robotics is a sub-field of the framework of embodied cognitive science presented in section 2.1. This new approach together with recent progress in hardware miniaturisation and cheap production has made the number of laboratories working on mobile autonomous robotics multiply in the past few years.

It is worth noting here that this approach does not state that the use of internal representation of the external world must be avoided. The point is firstly to show that much can be achieved using purely reactive robots; while in the case when there is necessity for internal state representing knowledge of the environment, it may well be sufficient, or even better, to distribute that knowledge among the behaviours in an implicit way, rather than hold it centrally in an explicit way.

This research studies the potential for behaviour-based robots to be able to coordinate themselves into potentially very large groups. It is related to the more particular field of *swarm* or *collective robotics.*

2.8 Collective Robotics

The study of collective robotics naturally extends research on single robots, but it is also a domain in its own right as multiple-robot systems can accomplish tasks that are beyond the scope of single robots. Distributed robotic systems are also different from other distributed systems implemented in software because of their real-world constraints.

In [Beni and Wang, 1991] Beni and Wang state the theoretical problems raised by collective robotics. Cao et al. present in [Cao et al., 1995] a deep survey of the field and more recently Parker [Parker, 2000] presents the state-of-the-art in the domain and identifies eight primary research topics:

- biological inspiration,
- communication architectures,
- localisation/mapping/exploration,
- cooperative object transport or manipulation,
- motion coordination,
- reconfigurable robots,
- learning.

The research of this thesis has resonances with most of these research topics (omitting only learning and cooperative object transport). It is related to biological examples (section 2.12); it is relevant to distributed exploration and localisation tasks (section 2.11.2) and to reconfigurable robots(chapter 6); it involves the motion coordination of a large number of robots (chapter 5) and crucially enquires into the potential for communication within a swarm of robots (chapters 4, 5 and 6).

The field of collective robotics is growing very rapidly and [Bonabeau et al., 1999] mentions several possible reasons for its current success:

- Collective robotics becomes an alternative to classical AI, relying on the non-classical idea that a group of robots might be more efficient than a single "intelligent" one.
- The progress of hardware, especially in miniaturisation, has made real multi-robot experiments possible.

- The field of Artificial Life (section 2.6) has propagated ideas about emergent behaviour that might have otherwise remained unknown to roboticists.

- By positive feedback. Work on the topic attracts more work, even if Bonabeau et al. note a decrease in the conceptual originality of recent publications

The field of collective robotics naturally divides itself into two different directions. One is primarily concerned with morphology and considers simple robotic modules physically attached together, which self-organise to generate global movement or to adapt to the environment. This is done either by controlling the joints [Lee and Sanderson, 2001, Takahashi et al., 2001] or by changing the configuration of the attachment of the robots, the latter being referred to as *self-reconfigurable robots* [Kotay and Rus, 1999], [Unsal et al., 2000], [Yoshida et al., 2000] (section 2.8.3).

The second direction studies the multiple interactions of autonomous robots cooperating in the execution of a task. Such work either relies on swarm intelligence or on more complex robots with higher skills. Foraging tasks, collective transportation [Kube. and Zhang, 1992], [Kawai and Hara, 1994] or distributed movement in formation are well-studied examples [Beckers et al., 1994], [Vaughan et al., 2000b] (section 2.8.2). Other work includes distributed learning [Mataric, 1994], cooperative map building [Dedeoglu and Sukhatme, 2000], etc.

The work presented here apparently belongs to the second direction. However as global connectivity of the robot network is an aim, the group formed can also be considered as a single entity and, therefore, also falls within the first direction.

2.8.1 minimalist robotics

Advances in materials science could lead to the construction of microscopically small robots. Nanoscale robots will have to *"operate in very large groups or swarms to affect the macroworld"* [Holland and Melhuish, 1996] while being very limited in computation, communication and sensing capabilities as well as largely unreliable. Within this framework, direct communication or explicit formation control are not likely to be realistic. The swarm intelligence paradigm described in section 2.5 provides an environment for solutions that might scale up to huge numbers of robots [Gage, 1993, Defago, 2001].

The work of Melhuish fits into this framework. It sought to minimise the computational and mechanical complexity of the robots to meet miniaturisation requirements. This work investigated minimal sorting, minimal clustering, minimal wall building and minimal swarming [Melhuish et al., 1999a], [Melhuish et al., 1998], [Melhuish, 1999b], [Melhuish et al., 1999b], [Melhuish, 1999a]. This work bases itself on biological examples. Other works include [Lewis and Bekey, 1992], [Genovese et al., 1992]. But in these examples, the research is primarily in simulation only and therefore lacks the real robot confirmation that would strengthen their argument.

However, the means to achieve the miniaturisation that is one driver of the minimalist approach is still an area of research. Thus, it is possible that the actual microrobots that will eventually be built may not be able to implement the algorithms developed. Indeed the paradigm of the Turing Machine that implements the minimalist rules might not be possible in such tiny machines. For example the work of Weiss and Knight, related to the embryonic field of *wet robotics*, tries to apply techniques from molecular biology in order to engineer automated responses in bacteria through genetic engineering [Weiss and Knight, 2000]. Because of the nature of genetic dynamics (see chapter 6 for introduction) the minimalist rules developed by Melhuish could turn out to be extremely difficult to implement in, for example, genetically engineered nanobots.

Because of the strong constraints chosen initially, the study presented here could be seen as part of the minimalist approach. Indeed what is here investigated is the potential of communication by itself to achieve useful behaviour. However, the hardware platform chosen to test the algorithms is certainly not minimal and uses high-level communication protocols and devices. Solutions could be found to bypass this problem; for instance using the same techniques as in [Weiss and Knight, 2000]. But then the same criticism could apply as to the work of Melhuish.

2.8.2 flocking and formation control

The task of flocking and formation keeping is one of the first investigated in the field of collective robotics. This early interest might be due to the work of [Reynolds, 1987] on his *"boids"*, mimicking the flock behaviour of simulated birds. Or might also have been motivated by military interests in formation keeping.

Flocking (also *swarming*) is the task of forming a group of robot when the actual shape of the group is not relevant. The group might be formed because of mutual attraction or because the group is attracted by the same attractor source (a *beacon*), referred to as *pseudo-swarming* in [Melhuish et al., 1999a].

The classical reference on flocking is the work of Reynolds. Primarily interested in designing a way to render visually the behaviour of a flock of birds for computer graphics purposes, he defined an agent-based model that showed the ability to avoid obstacles and reform the flock afterwards [Reynolds, 1987]. Although not aimed at being biologically plausible, this work has raised questions on how actual birds or fishes display such behaviours. Work on this topic in robotics can be found in [Lewis and Bekey, 1992], [Mataric, 1992], [Melhuish et al., 1999a], [Melhuish, 1999a], [Hayes et al., 2000]. This example introduces the idea of *secondary swarming* where individuals relay the signal of the beacon which makes the robots move as a group [Melhuish, 1999a].

Similarly, Weßnitzer and Melhuish present an algorithm for target hunting where the robots share their perceived progress towards the target within their neighbourhood [Weßnitzer and Melhuish, 2003]. Because of line-of-sight obstruction, some of the robots cannot see the prey. The robots that do see it then act similarly as a secondary beacon. It results in cooperative hunting of the prey, whose ability to escape is greatly reduced by the swarm acting like a catching net.

When the shape of the swarm is relevant to the task and needs to be maintained, we speak of *formation control*. In most cases, the task for these robots is to move in the same direction while trying to maintain relative distances the same. This typically involves a constrained number of robots.

In [Balch and Arkin, 1998] the authors state that research on formation control can be divided into unit-center-, leader- or neighbour-referenced control. Typically the first solution needs a global positioning ability in order to compute the position of the unit-center. This requirement on global position can be found in numerous examples [Defago and Konagaya, 2002, Lewis and Tan, 1997]. The leader solution is also highly dependent on the proper functioning of this leader. Naffin presents a way to get round this by the periodic local election of the leader [Naffin and Sukhatme, 2004].

The neighbour referenced solution is then a possibility for decentralised control with local sensing. Typically the formation that has to be reached is predefined in the controller [Sugihara and Suzuki, 1990], [Fredslund and Mataric, 2001], [Weßnitzer et al., 2001], with sometimes the investigation of the possibility to switch between two different formations [Weßnitzer et al., 2001], [Fredslund and Mataric, 2001]. The most interesting work on formation control for this research is the work presented in [Balch and Hybinette, 2000] where the use of a neighbour-referenced algorithm allows for scalability. In this case the global formation is not stated a-priori but instead emerges from the local relative positions the robots try to maintain. Nevertheless this example relies on the sensing of local positions of neighbours in order to compute a movement vector. This sensing is notoriously noisy and involves high level abilities for the robots.

Another area of work on formation is the investigation of the theoretical problems stated by formation control, for instance the possibility of a group of "robots" to actually reach the formation, agreement on coordinate system, etc. [Prencipe and Gervasi, 2002, Suzuki and Yamashita, 1994]. These examples make assumptions on the abilities of the robots that are often unrealistic, such as global position sensing, or sometimes state theoretically unstable extreme points in state space that do not happen in real systems because of noise. One of the most interesting examples belonging to that branch is the work on distributed formation control by Yamaguchi et al. who are able to formally prove the asymptotic stability of their approach [Yamaguchi et al., 2001].

A research in between flocking and formation is the work by Quinn et al. which evolves a formation behaviour on very simple robots by selecting the most successful controllers at moving the group [Quinn et al., 2002a, Quinn et al., 2002b]. In this case the successful controller makes the group adopt a formation, but this formation is only the result of the scarce sensory resource, and not a prerequisite of the research. The study presents work on simulation and successful implementation on real robots. In [Baldassare et al., 2003], formations are also evolved in simulation, but in this case the group behaviour is explicitly selected by the fitness function. Another interesting example of evolutionary computation involving a genetic regulatory network for the control of a robot swarm can be found in [Taylor, 2004].

The work presented in this thesis relates to flocking, swarming and formation keeping. Chapter 4 investigates flocking through the possibility for locally communicating robots to guarantee the global connectivity of the swarm; in chapter 5, swarming by allowing the flock to be attracted by a simulated light source. Finally in chapter 6, the swarm aims to maintain a global shape - although loosely - while moving.

2.8.3 reconfigurable robots

Awareness of situatedness has placed a focus on the morphology of the robots [Pfeifer, 2000], [Støy, 2001a] and opened large areas of interest such as reconfigurable robots [Kotay and Rus, 1999], [Unsal et al., 2000, Yoshida et al., 2000] consisting of elementary robots that reassemble themselves to form a bigger entity, modular robots whose morphology can be changed easily [Lee and Sanderson, 2001], [Takahashi et al., 2001], [Kawai and Hara, 1994] and evolution of morphology in simulation [Sims, 1994], [Bongard and Pfeifer, 2001], [Bongard, 2002] and with real world confirmation [Lipson and Pollack, 2000].

Reconfigurable robots are, in effect, robot collectives that behave as a whole. The motivation is to achieve function from shape, Allowing individual modules to connect and reconnect in various ways in order to achieve the required function. For instance, overall movement is performed by many single moves that when coordinated, achieve the task. This of course raises complex problems of distributed control and communication, but the main problem to be overcome appears to be the physical construction of those devices and much work is dedicated to this task. As a result simulation tools are widely used, raising problems discussed in section 2.9

For examples of distributed control, see the impressive work of Støy on arrangements of articulated modules that can be reconfigured on the fly [Støy et al., 2002, Støy, 2004]. Other examples can be found in [Christensen et al., 2004, Østergaard and Lund, 2004]. For technical descriptions of hardware implementation, see [Kurokawa et al., 2004, Murata et al., 2000, Kotay and Rus, 1999, Unsal et al., 2000, Yoshida et al., 2000].

This thesis' study is concerned with groups of robots that are more loosely connected. While in the field of reconfigurable robots, individuals are normally physically linked together, the approach here considers limited-range radio connections that build a dynamic wireless network. Connections are unreliable and often lost. An interesting analogous example of group pattern formation using wireless connections can be found in [Weßnitzer et al., 2001]. A study situated half-way between our work and reconfigurable robots is the SWARMBOT project that considers ant-like mobile robots able to physically attach to each other to behave as a single multi-robot system [Mondada et al., 2004, Trianni et al., 2004].

2.9 Real World Issues

Robotics is of course primarily concerned with building real artifacts that can operate in the real world. But the cost, both in time and money, of building and testing real robots has made simulation a necessary and unavoidable tool.

The work presented here follows the common approach of conducting early experiments in simulation and then transferring them to real robots. The advantage of such a methodology is the substantial gain in development time as hardware problems are considered after the algorithms have been developed. The major drawback underlined by Brooks [Brooks, 1991] comes from the fact that a simulation always makes assumptions in modeling the real world. This can lead to results that would be infeasible in reality, and can also lead the researcher in completely the wrong direction.

As implementing a realistic simulation can be a major undertaking in itself, a trade-off between realism and oversimplified simulation must be found. Work from [Martinoli and Mondada, 1998] and [Hayes et al., 2000] discusses the opportunity of high-level simulations, showing that very simple probabilistic models that consider the robots interacting among themselves, and with the environment as a sum of probabilistic events, are able to render the dynamics involved in a foraging or swarming task. More work on probabilistic modeling can be found in [Martinoli et al., 1999c, Martinoli et al., 1999b, Martinoli et al., 2003, Billard et al., 1999a, Trianni et al., 2002]. The potential for the use of these models for the purpose of analysis is discussed in section 2.13.

As a matter of fact this question relates to the concept of embodiment introduced earlier (see section 2.1). As an agent cannot be considered without its environment, research relying only on simulations, which by their essence are not able to render the complexity of real-world interactions, can lead to results that make false assumptions on the nature of the agent-environment interactions.

The problem of discrepancy between real robot experiments and simulation is particularly acute in research in robot evolution (see section 2.6.1). The use of simulation to model the evolutionary process can indeed bias the result [Jakobi et al., 1995]. On the other hand, real robot evolutionary experiments are the exception because of the time scale involved. To circumvent this Watson et al. [Watson et al., 2002, Watson et al., 1999] introduced the concept of *embodied evolution* where a group of real robots co-evolve in a shared environment, exchanging genes. This setup seems to be able to speed up the evolutionary process and converge to a satisfying solution.

Studies involving the use of simulation are abstract in two senses. Firstly they are abstract in the representation of their algorithm. Secondly the environments in which the algorithms are being run are abstract. The abstraction of the environment can be close to reality in case of a faithful simulation but this fidelity is not always a requirement.

The present research stands in a hybrid position. It seeks to develop abstract algorithms that are implemented on abstract Turing machines, while these machines live in the real world. Beyond the problem of faithfulness of a simulation (see chapter 3) is the open question of the relevance of abstract models to the solution of real world problems.

2.10 Communication

As stated by Lachmann in [Lachmann et al., 2000] unlike energy sharing, exchanging information is not subject to conservation of total mass. In other words an agent that shares its knowledge with others is not losing any part of itself. This basic consideration makes communication a preferred approach when the aim is to improve the efficiency of a system.

As such it has been mentioned from the beginning as a problem of primary importance in collective robotics [Beni and Wang, 1991] and is still a strong area of interest as stated in Parker [Parker, 1996].

The act of communicating can be deliberate with the agent explicitly invoking a signal transmission. This is referred to as *direct communication*. In contrast *indirect communication* is a consequence of an agent's behaviour and its effects on the environment. In nature both types are used by most species. For instance human communication can rely as much on body language cues as on the actual meaning of the message. In an interesting example, Noble shows how communication between animals can avoid potentially harmful conflicts [Noble, 2000]. For a comprehensive survey of animal signaling and its implications with evolution, see [Endler, 1993].

2.10.1 indirect communication

Introduced by research on social insects, this type of communication relies on interactions with the environment. The agents act on the environmental media as they would in a communication medium. The message that consists of changes in the environment, is delivered when such changes are sensed by another agent, perhaps triggering a special behaviour. It is referred to as *stigmergic* in the literature. Studies on foraging and sorting tasks [Melhuish et al., 1998, Beckers et al., 1994] and distributed building of nest structures [Bonabeau et al., 1999] rely on stigmergic communication.

Another example is the research of Kube and Bonabeau who achieve cooperative object transport without direct communication [C.Kube and H.Zhang, 1994, Kube. and Zhang, 1992]. In their study on multi-robot communication Balch and Arkin show that stigmergy can be sufficient to complete the tasks they consider, but that direct communication is able to increase efficiency [Balch and Arkin, 1994].

Another instance of indirect communication makes use of direct sensing of other robots that can be interpreted as environmental cues. It implies the ability to discriminate robots from other objects, so-called *kin recognition*. Robots in formation usually use this kind of interaction [Reynolds, 1987, Mataric, 1992].

2.10.2 direct communication

Direct communication occurs when an agent sends a direct signal to the other agents, either broadcast or point-to-point. In [Mataric, 1994] Mataric states the potential of directed communication to overcome limited abilities of robots.

Implementations usually involve an exchange of internal states or sensing information. Examples include [Weßnitzer, 2001, Vaughan et al., 2000a, Yanco and Stein, 1992].

Such communication is usually implemented with radio or infra-red devices. As it involves a shared medium some solutions have to be found to avoid collisions and loss of messages. In [Wang and Premvuti, 1994, Wang et al., 1995] a decentralised media access protocol is defined within the framework of mobile robotics.

The factor of shared media is also of primary importance when considering the scalability of solutions. Indeed following Gage [Gage, 1993] in his consideration of groups of "zillions" of robots, it is interesting to study local solutions involving limited range radio devices whose local use of bandwidth enable such scalability. Research in this direction includes the work of Arai et al. [Arai et al., 1993], Yoshida et al. [Yoshida et al., 1998] and Støy [Støy, 2001b, Støy, 2001a].

Støy [Støy, 2001a] further differentiates directed communication into *situated* and *abstract* communication. The latter is the type of communication where the physical signal that transports the message is not considered to carry any meaning. On the other hand situated communication occurs when "both the physical properties of the signal that transfers the message and the content of the message contribute to its meaning". An example is given: when a person says "move towards me" the hearer knows from the direction of the sound and the meaning of the message what to do. It is important to note that the sound alone does not carry any meaning and nor do the semantics alone. The meaning emerges from the combination of both.

2.11 Networks

2.11.1 ad hoc networks

The group of communicating robots considered in this study form a communication network. Because of the assumption of limited communication range and the focus on distributed solutions, this network is a *multi-hop ad hoc wireless network*. In such a scheme when data needs to be sent to a robot that does not stay within range it will be forwarded to a neighbour believed to be nearer to the destination. The process is repeated and the data eventually reaches its destination via several intermediate robots. This process is known as *message routing* and may involve some sort of global knowledge of the network.

The constantly changing topology of the network makes many routing algorithms used on fixed network topologies unfeasible; the bandwidth cost of maintaining accurate routing tables is too high. To circumvent this problem Johnson [Johnson, 1994] proposes to divide the task into two subtasks: route discovery and route maintenance. Instead of keeping up-to-date information on routes to all nodes in the network, the protocol finds one on demand and is then able to maintain it. Another work studies flooding strategies in wireless ad hoc networks [Lim and Kim, 2001].

Because of high level interest from the telecom companies, this area of research is constantly evolving and new algorithms are presented regularly. A comparison of different algorithms proposed can be found in [J.Broch et al., 1998]. All of the algorithms proposed show great sensitivity to the increase in node numbers or node mobility.

In his research on swarm programming Evans [Evans, 2000] expresses doubts about the need for node-to-node routing protocols in such an environment of agents. A possible alternative could be found in the framework of amorphous computing which is described in section 2.11.2.

2.11.2 distributed sensor networks

Man's desire to measure and control different places at the same time is well known and has been investigated at least in fiction (see for instance Philip K. Dick's *Ubik*). Sensor networks are an engineering answer to this need. The aim is to build a group of sensors that would be able to monitor different places simultaneously and therefore measure dynamical properties of phenomena such as underwater currents etc. The natural metaphor being for instance the capacity of an ant to detect an intruder and spread the alarm through the excitation of its neighbours, leading to the mobilisation of the whole colony [Adler and Gordon, 1992, Frehland et al., 1985].

Advances in sensor technology and computer networks will enable the design of large swarms of mobile microsensors communicating through wireless media. These developments will bring new challenges, such as large volumes of data to process, limited bandwidth, limited power resource and the unreliability of the environment. These challenges are comparable to those that can appear in collective robotics.

A review of recent research can be found in [Qi et al., 2001] which divides the problem into four aspects: data processing paradigm, sensor fusion algorithm, communication network structure and optimal sensor deployment strategy. The study of this thesis is concerned with the last two aspects. Other examples include [Bulusu et al., 2001, Hackwood and G.Beni, 1992, Franklin et al., 1995]

Considering the mobility of the nodes it is obvious that the network structure is dynamical and therefore the field of ad-hoc wireless networking is of primary importance. Indeed to gather data to a special, single node will require some sort of routing. One answer could be a point-to-point communication protocol assuring connections between nodes; but Estrin et al. [Estrin et al., 1999a] have presented as an alternative a data-oriented environment which avoids the power consuming maintenance of a connection-oriented network.

The close links between sensor networks and collective robotics underlined above have brought together researchers of both fields, underlying the overlap between the problem of coordinating the robots carrying the sensors and the algorithm that directs the flow of information [Estrin et al., 1999b]. In [Winfield, 2000] Winfield introduces a mechanism to gather data to a collection point despite a constantly changing network topology.

An example of collective sensing and decision making implemented in real robots can be found in [Weßnitzer and Melhuish, 2003].

Related to sensor networks is the project of Amorphous Computing at MIT. It is concerned with programming self-assembling systems of unreliable, irregularly spread out, communication- and processing-limited elements [Abelson et al., 2000, Nagpal, 1999]. As in collective robotics, biology acts as a metaphor for many of the solutions which emerge.

2.12 Biological Examples

As stated in chapter 1, this study is not primarily concerned with mimicking natural behaviours displayed by some social animal. Indeed, as the objective is to investigate the use of radio communication, it is doubtful from the outset that any animal would be realistically mimicked by this research.

Nevertheless the collective behaviours that we set out to demonstrate with the swarm, namely swarming, taxis and global shape control are tasks that are performed by natural collectives. It is therefore not surprising that the solutions developed have biological resonances.

2.12.1 essential genes set

The recent sequencing of whole genomes of organisms has raised the question of which genes are essential for a cell to survive. As the interactions between genes within the cell are very complex and still poorly understood, a way to circumvent this complexity is to inactivate genes one by one in a simple organism having only a few, such as bacteria, and see if the corresponding cell survives. This corresponds to a synthetic or constructionist approach rather more than an analytic one (see chapter 3). Although it is difficult to consider an organism without its corresponding environment, the result of such experiments is a subset of the whole genome that should be essential to any cell, at least in the genes function [Kobayashi et al., 2003, Fraser et al., 1995, Hutchison et al., 1999, Gil et al., 2003].

It is interesting to note that in some examples the resulting subset lacks expression regulator genes [Kobayashi et al., 2003], suggesting that regulation is not needed for processes concerning single cell basic machinery but rather for more complex mechanisms such as homeostasis or differentiation. This finding strengthens the plausibility of the evolutionary framework presented in chapter 6, where the aim is to achieve more pattern complexity from a base consisting of an amorphous connected swarm.

Figure 2.2: cyanobacteria (spirulina)

2.12.2 bacteria

In a paper and later in a book, Shapiro stresses the fact that bacteria have always been studied in pure colonies because of the needs of medical bacteriology. Potential pathogens are identified by showing that pure cultures grown from a single individual are responsible for the given disease. This has led to complete neglect of the complexity of natural bacterial colonies that have just begun to be thoroughly studied [Shapiro, 1988, Shapiro and Dworkin, 1997].

For instance, colonies of *Bacillus subtilis* show strikingly different patterns of growth according to changes in the agar or nutrient concentration [Matsushita, 1997]. These patterns can be modelled by random growth models [Ben-Jacob and Cohen, 1997]. More specifically the cells situated in the inside of colonies growing in soft agar with high nutrients show erratic motion that is not precisely brownian. But cells situated on the edge of the growing colony are rather still. It is conjectured that the colony changes the medium allowing the bacteria inside to swim.

Another example, *cyanobacteria* (figure 2.2), show spatial patterns of differentiation between individuals to conduct differential metabolic functions. Individuals seem able to dedifferentiate to optimise the distribution of heterocysts. And the exchange of nutrients appears to be direct rather than through the medium [Adams, 1997].

Myxobacteria colonies are able to move as *"an intact unit [and] [w]hen an individual cell moves a few microns beyond the edge it quickly pops back into place as though drawn by an elastic thread"* [Shapiro, 1988]. When two cells of *Myxococcus virescens* meet, they either align themselves or rub alongside each other before separating. *Myxococcus xanthus* form spherical colonies that engulf prey in order to digest them.

These examples strongly resonate with the swarm behaviours developed in this thesis (see chapters

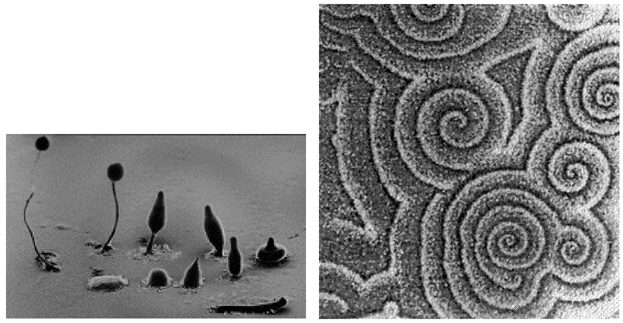


Figure 2.3: development of dyctiostelium discoidum and patterns of cAMP concentrations

4, 5 and 6) and are a motivation for the evolutionary algorithm presented in chapter 6.

2.12.3 slime molds and morphogenesis

Although biomimetics was not the aim at the start of this work, it appears that the aggregating and moving swarm (see chapter 4, 5 and 6) shows comparable properties with cellular slime mold *Dyctiostelium discoidum* (figure 2.3). One of the most widely studied organisms because of its relevance to morphogenesis, it provides a model for studying developmental processes such as chemotaxis, cell sorting, pattern formation and complex behaviour. More particularly, it shows a striking behaviour:

When food becomes scarce in the substrate that the unicellular amoebae inhabit, they aggregate and form multicellular migrating slugs, that eventually stop to form a fruiting body of spore cells on top of a stalk. When aggregating they typically form streams towards the center.

[Savill and Hogeweg, 1997] present a model that accounts for aggregation, cell sorting, slug formation and slug migration by presetting three different cell types and with the help of only three processes: production of and chemotaxis to cAMP and differential cellular adhesion. The present study is related to this work in trying to make the swarm aggregate, move and keep a global shape; and the algorithms developed can be seen as analogous to differential cell adhesion. Interesting work in the field of robotics that is also related to amoeba can be found in [Takahashi et al., 2001, Takahashi et al., 2004]. But it relies on physical bounds that prevent morphogenesis.

This fits in the more general problem of *morphogenesis*. Pioneering work from the english mathematician Alan Turing and his intuitions on diffusing *morphogens* [Turing, 1952], together with the advent of molecular biology, has led to results in embryology showing that gradients of a few molecular signals are able to organise the development of the fly embryo [Nusslein-Volhard, 1996].

These results have been applied to various examples in the field of Artificial Life [Eggenberger, 1996] [Eggenberger, 1997] [Bongard and Pfeifer, 2001] [Bongard, 2002] [Bongard and Paul, 2000]. Also the relatively new field of Amorphous Computing investigates self-organisation of computer "morphology" through the gradient diffusion metaphor [Abelson et al., 2000, Nagpal, 1999] [Nagpal and Coore, 1998].

2.12.4 social insects

Observations of the impressive achievements of social insects are at the core of research in swarm robotics. In these examples, the complexity induced by the collective cannot be explained by the capacities of the individual, despite the complexity of a single insect compared to a robot. Many examples with their potential applications can be found in [Bonabeau et al., 1999].

Examples specially relevant to this work are to be found in various species of ants. The organisation of ant colonies shows dynamical specialisation of individuals into groups performing definite tasks. Removal of individuals from one group results in the readjustment between the others to carry on the task. In polymorphic species (*Pheidole* for instance), this reorganisation occurs even between morphologically specialised groups. A direct application to robotics can be found in [Krieger and Billeter, 2000].

Weaver ants (*Oecophylla*, figure 2.4) form chains with their own bodies (figure 2.4) in order to cross wide gaps and fold leaves to build nests. The explanation of such behaviour cannot rely on centralised command and control and stands therefore as an existence proof supporting the plausibility of the present work.

The study of patrolling ants suggests that they are capable of effectively covering the region around the nest and, whenever needed, of spreading an alarm signal through direct encounter [Adler and Gordon, 1992, Frehland et al., 1985]. Study of the ant *Lasius fuliginosus* shows that the rate of encounters is kept relatively constant as density changes. It is conjectured that they do so in order to keep network connectivity. The framework of sensor networks rely on the same concept.

Examples of chorusing in crickets have shown their potential to achieve flocking and control of the group size [Greenfield, 1994, Melhuish et al., 1999a].

Also relevant, though not studied in this work, is the potential of virtual foraging ants to main-

Figure 2.4: weaver ants

tain accurate routing tables in dynamic network topology. In [Bonabeau et al., 1999] are presented comparisons with powerful algorithms for possibly faulty static networks. But the same framework could be of interest in dynamic situations.

2.13 Analysis

As the presented investigation is primarily concerned with designing emergent behaviours in complex systems, analysis becomes difficult. Indeed dealing with non-linear non-deterministic dynamics, makes formalisation able to give analytical results rapidly intractable. Moreover, in order to prevent the swarm from becoming trapped in a dead-end pattern, the behaviours governing the robots include a random choice of heading, which introduces further difficulties for analysis.

To overcome this limitation there have been several propositions:

- *computational models*: In order to be able to study more deeply what is happening to the robots, behaviours and interactions are simplified [Gazi and Passino, 2003], sometimes to a level that seems to prevent actual relevance for robotics. These examples are related to more abstract study in complex systems [Lachmann et al., 2000, Delgado and Sole, 1997]. Also related to this topic are the *active walker models* that consider brownian particles that change the field governing their motion, as would a moving mass with the gravitational field. For an interdisciplinary review see [Lam et al., 1997, Lam and Pochy, 1993].

- *time series analysis*: Using techniques from experimental physics, it is possible to analyse the behaviour of the robots through the examination of time related measurements [Schreiber, 1999, Wright et al., 2001, Penn, 2002].

- *probabilistic models*: By considering the behaviours of the swarm as probabilistic events, it is possible with simple geometric considerations to extract probabilistic equations that faithfully represent the behaviour displayed by real robots or more detailed simulations. Perhaps the most developed and the most promising in the field of robotics is thanks to the work of Martinoli [Martinoli and Mondada, 1998] [Martinoli et al., 1999c] [Martinoli et al., 1999b] [Martinoli et al., 2003] [Hayes et al., 2000] [Billard et al., 1999b] [Trianni et al., 2002] [Lerman and Galstyan, 2001].

For analysis purposes, this study defines several metrics that are defined in chapter 3. In his remarkable study of the behaviour of discrete dynamical networks, Wuensche is able to dissect their dynamics and to display preimages or basins of attraction [Wuensche, 1998], but he considers only static networks and the computation involved is likely to prevent applying the same analysis to dynamic networks. On the other hand this technique could be applied to the results of the genetic regulatory network introduced in chapter 6. In this case, the genome codes for a static dynamical network that influences the behaviour of a robot, which then has an influence on the expression of the genes by diffusion across the network.

Chapter 3

Methodology

"Most people don't realise that I use chance as a discipline"

John Cage.

"knowledge only is not enough. True understanding comes from experimentation."

Seymour Papert

In the young field of robotics the way to assert a scientific discovery is not yet commonly agreed. The difficulty of mathematically formalising the notion of emergence and the cost of real robot experiments in terms of time and money has led researchers in the field to state experimental guidelines that remain far from a solid scientific proof [Bisset, 2003].

It is now widely agreed that real robot experiments are needed to confirm results from simulations, however accurate they may be, because of the obvious impossibility of representing all of the interactions that can have an influence on the problem considered. The idea coming from the new roboticists such as Brooks [Brooks, 1991] is that the complexity of the real world is best represented by itself.

As a result, most studies rely on the following methodology : the early research work is done through the help of simulations presenting a limited degree of accuracy, but the confirmation of the results is sought by building a real experiment. The *reality* of such experiments is of course the closely controlled reality of a robotic laboratory as the robots, unless designed specifically for it, will not cope with the uncertainties of the outside world.

An interesting complementary research tool is the relatively new application of probabilistic modeling to the field, for instance in [Martinoli and Mondada, 1998, Lerman and Galstyan, 2001] that tries to bypass the formalisation problem by considering the interactions as probabilistic events and producing either macro- or micro-level update equations. This stands at the mid-point between formalisation and simulation and can bring some insight into the dynamics involved in the experiments.

Our research follows the above mentioned practice in presenting results from simulation and then confirming them, in part, through real experiments; while the use of probabilistic modeling is left for further research.

3.1 Constructionism

As stated in chapter 1, this research can be seen as an instance of the constructionist approach to the increase of knowledge. Instead of using an analytic approach that would consist of observing an existing system in trying to understand a given phenomenon, we try to *build* robots or simulate them using computers to gain understanding in the potential of minimal communication between large groups of robots. It is also referred to as a *synthetic* approach [Pfeifer and Scheier, 1999, Braitenberg, 1984]. Constructionism, as introduced by Seymour Papert, derives from the constructivist theories of Piaget [Papert and Harel, 1991, Resnick, 1999].

3.1.1 constructivism

In trying to understand how children perceive their reality, learn and develop, Jean Piaget initiated the constructivist theories in stating that knowledge does not exist by itself but is instead *constructed* in the mind of the learner [Piaget, 1954]. More specifically, knowledge is rarely transferred intact from the mind of a teacher to its student, as in a downloading operation, but the student has to reinterpret and reappropriate this knowledge to be able to assimilate it. In some more radical strands, the knowledge is always altered by the transfer.

To draw a parallel into philosophy of science, the constructivist theories relate to the relativist approach, that challenges the objectivism and positivism of the 19th century and places the observer as an essential component of the scientific experiment. In the words of the radical constructivist von Glasersfeld, [von Glasersfeld, 1995]: *"Objectivity is the myth that an observation can be made in the absence of an observer"* .

3.1.2 constructionism or learning by building

After collaborating with Piaget, Seymour Papert of MIT refined the constructivist view. In his own words : *"Constructionism - the N word as opposed to the V word - shares constructivism's connotation of learning as "building knowledge structures" irrespective of the circumstances of the learning. It then adds the idea that this happens especially felicitously in a context where the learner is consciously engaged in constructing a public entity, whether it's a sand castle or a theory of the universe"* [Papert and Harel, 1991].

While Piaget is interested in how pure cognitive processes and logical thinking appears in a child's development, Papert stresses the importance of context and external tangible artifacts in the process of learning. Constructionism in emphasizing the concreteness of knowledge, is more *situated* [Ackermann, 2001], in a way analogous to the discussion of chapter 2. In Piaget's view cognitive growth is *"a progression from egocentric beginnings to a final "formal stage" when propositional logic and the hypothetico-deductive method "liberate" intelligence from the need for concrete situations to mediate thinking"* [Turkle and Papert, 1992].

In "Epistemological Pluralism and the Revaluation of the Concrete" [Turkle and Papert, 1992], this view is challenged and the focus on general stages of human development is shifted to the study of individual or culturally related learning styles: *"formal reasoning is not a stage but a style"*. In this article, Turkle and Papert ask for an epistemological pluralism in accepting and encouraging concrete knowledge to challenge the *"epistemological elite"* of those who master formal thinking.

In the same article, Turkle and Papert remind us that *"there is a tradition of scientific epistemology that sees the essence of science in objectivity and the essence of objectivity in a distanced relationship with the object of study"*. They later mention the case of Barbara McClintock, molecular biologist who *"came into increasing conflict with formal, "hard" methods of molecular biology"* and was later awarded the Nobel Prize when formal investigations subsequently came to the same conclusions that she derived from her "soft" investigations.

Also they stress the fact that in spite of being the archetype of a logical machine, *"the existence of anything but an analytic approach"* in computing makes the computer a perfect tool for pluralism. Accordingly, in an area where formal thinking has yet shown to be of little use, Mitchell Resnick uses computers and a constructionist approach to render decentralised phenomena more familiar to students [Resnick, 1999]. In his book, he presents his inquiry situated at a crossroad between science and teaching, and the thoughts that emerged from it.

Turkle and Papert emphasize that decentralised phenomena, or emergent AI, present *"a possibility for new alliances between computation and the theorists as well as the practitioners of a science of the concrete"* and that concrete thinking should become *"an object of science in its own right"*

3.1.3 constructionism in science

As stressed by Turkle and Papert in the case of McClintock, many scientific researchers follow a non analytic way of discovery and it is only at a second stage that analysis and formalism come to strengthen the argument. In fields where a general theory is lacking, a lot of research goes on through experimentation trying to delineate the contours that this hypothetical theory should encompass while also work to try and formulate such a theory. This is the essence of the *synthetic approach* introduced in section 2.3.

In fields where strong non-linear interactions predominate this approach is mostly seen. As reduction into smaller problems would lose the interaction and as the formal approach can cope with only a small number of variables, statistical models or computer simulations are a widely used tool for prediction.

A striking example to illustrate this assertion is the recent work in molecular biology on the *essential genes set*. Following the sequencing of whole genomes of organisms, the question has been raised "which genes are essential for a cell to be alive ?" Because of all the unknown interactions that remain to be discovered, the method that researchers use to understand the system is not by first theorising about the functions needed for a cell and then defining the corresponding gene set. Instead they take a living cell with a small genome, a bacteria for instance, and then infer a plausible essential genes set, either through comparison with known sequences of related organisms [Fraser et al., 1995, Gil et al., 2003], or by inactivation of all genes one by one and testing each mutant for survival - enormous and painstaking work [Kobayashi et al., 2003, Hutchison et al., 1999]. Indeed the discovery of these genes is of considerable interest. The functions are likely to be found in each living cell and could by themselves constitute a definition of life. And the discovery of essential genes of unknown function can lead to new understanding of cell life processes [Kobayashi et al., 2003].

Actually being synthetic, this project is an instance of the constructionist approach. Indeed we are facing a very complex and strongly non-linear system that remains poorly understood and constructionism provides a viewpoint to grasp the problem in which non-linearity strongly limits the explanatory potential of the classical reductionist approach.

Another example is the current trend in developing humanoid robots that, despite sometimes being a publicity-oriented engineering effort, can be another way to investigate cognition and its relation with a body and an environment [Knoll et al., 2001, Ziemke, 2001]. Another is the so-called *"constructive biology"* which is *"biology motivated by the desire to understand how biological systems actually are constructed by nature and develop over time rather than just to obtain a descriptive understanding"* [Nehaniv et al., 1999].

3.1.4 this research is constructionist

In investigating multi-robot behaviour linked with local communication, this research steps into the world of non-linear interactions. In the experimental set-up presented here robots communicate only with their nearest neighbours and this communication is either possible or not possible, like a step function without intermediate values. The robots' behaviour is affected by this communication, and the behaviour itself has an influence on the availability of communication. There is no linearity as the behaviour of a single robot cannot be investigated and then extrapolated to the group. The behaviour of the group is not the sum of the behaviours of its individuals by the very fact of the coupling of their behaviour by means of communication.

The method for investigating such phenomena in this research is to build the communicating components of the group, first virtually in simulation, and then in reality with physical robots. By this means emergent behaviours are demonstrated that the lack of formalism would make impossible to predict theoretically. The unavailability of theory is bypassed by building *"a public artifact"* [Papert and Harel, 1991].

In this respect this research is constructionist. We as researchers, are engaged in a process of learning a discipline. But as it is scientific research the teacher is not available. Still we teach ourselves through artifacts and, as Papert, it is believed that learning should *"happen especially felicitously"* (op.cit.).

Following Harvey [Harvey, 2000], we draw attention to the fact that the phenomena constructed actually resist the traditional methodology of reduction into smaller problems, because of strong non-linear interactions. Arguing that a reductionist approach is impossible in this case is not the point here, but to present an alternative approach that might bring a constructionist insight into such non-linear phenomena.

3.2 Simulation Details

The research follows a method of investigation that consists of, firstly, designing and implementing a distributed control algorithm for the swarm, in simulation, to gain insights into the dynamics involved and the problems that may arise. It must be borne in mind that the results are highly dependent on the degree of accuracy of the simulation and represent therefore only partial results. This is the reason that the resulting algorithms are then, secondly, implemented on real robots, taking full advantage of the early simulation work.

As developing a simulation that takes account of every interaction contributing to the dynamics would be impractical, a simple approach is implemented which tries to take into account those variables relevant to the problem considered. In every choice that has been made while designing this simulation care has been taken to always bear in mind that the algorithm will eventually be transferred onto real robots hence, hopefully, avoiding impractical solutions.

In this simulation noise is not modeled accurately, but instead introduced as false sensor readings or random message loss. The reason for this approach is that the best way to simulate sensor noise is to choose a particular sensor, then record and use readings in all possible situations such that sensor values in the simulation will be readings from these measurements. Here the purpose is to test the robustness of the algorithm itself to white noise and not the robustness of the algorithm on a particular robotic platform. This is the reason why such measurements were not made on the linuxbots (see next section).

The probabilistic modeling described in [Martinoli and Mondada, 1998, Martinoli et al., 1999c, Hayes et al., 2000] and [Lerman and Galstyan, 2001] is not used here as a design tool. In fact the result of such modeling is a simulation that is certainly computationally less expensive than a more realistic simulation but is not descriptive of the actual behaviour of the robots. Therefore it seems that until further development, the probabilistic modeling methodology is more interesting as an analysis tool, giving greater insight into which variables are actually relevant to the problem considered. To actually design and refine the reactive behaviour rules requires more of a descriptive simulation. This is especially the case when dealing with emergent behaviours.

However, in this thesis, probabilistic modeling will not be used for analysis either. Firstly because the approach of Martinoli et al. considers problems in bounded environments: in these cases, it is possible to state a probability for a robot to be at a given place within the environment. This is not possible in the present work, because of the initial requirement of an unbounded environment. Secondly, the alternative approach of Lerman and Galstyan is also not applicable as such, for this technique requires the multi-agent system considered to be *markovian* - *i.e.* the agent's future state should depend only on its present state. The algorithms developed in this work rely precisely on action according to the confrontation of past and present local configurations of the network.

Nevertheless, we believe that a probabilistic modeling of the behaviour of the swarm at the level of the network is possible and a preliminary step towards the characterisation of swarm spread on the base of adjacency matrix information is presented in section 4.4.4. Such a measure would be needed in order to confirm that the behaviour of modeled swarm corresponds to the one observed in more detailed simulations.

3.2.1 robot architecture

The robots that will be used in the real robot experiments are the Linuxbots developed in the IAS lab in Bristol[1] [Winfield and Holland, 2000]. They are wheeled differential drive robots with a circular body that can be equipped with several sensors. In the simulation, it is assumed the robots are able to move forward and turn on-the-spot with a precision ranging from perfect to errors of 10 % on the distance or angle travelled, that they have infra-red avoidance sensors, are equipped with a limited-range radio device and that they carry an omni-directional light sensor able to detect whether a robot is illuminated or not.

There are three infra-red proximity sensors, one on each side of the front of the robot (L/R), and the third at the back (B). All sensors have the same range, smaller than the range of the communication device. Note that this does not take into account the variation in performance that real sensors show. An idealised representation of sensor coverage is shown in figure 3.1.

The robots' control system is designed as a *finite state automaton*. A robot will be in one of several mutually exclusive states and switch between them according to environmental cues. In this study

[1] Intelligent and Autonomous System Laboratory, University of the West of England, Bristol

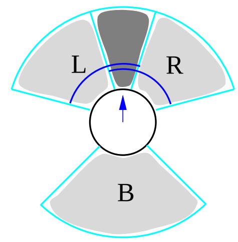


Figure 3.1: idealised coverage of avoidance sensors

the messages coming from other robots are considered as an actual part of the robot's environment. This is a direct consequence of the framework introduced by [Støy, 2001c] as *situated communication*. In this paper Støy emphasizes that communication is actually tightly linked with a situation and in certain cases cannot be separated from it without losing the meaning of the message. Robots that are limited in their communication range actually wrap their messages with an implicit meaning that the recipient can interpret as "this robot is in range of communication". This thesis' work starts from the assumption that this implicit meaning can have an influence on the behaviour of the robot, that is to consider it as an environmental cue.

Other environmental cues that are considered by the robot to alter its behaviour are infra-red reflections from potential obstacles (robots or inert obstacles) and light from the light beacon. Again it is not attempted to model the physics involved but instead a random possibility of binary false reading is introduced that can be tuned to see the influence of a certain level of noise.

The control architecture can be divided, following [Brooks, 1986], into two layers: the *avoidance* behaviour and the *communication* behaviour (see figure 3.2, with the transitions triggered by time (T), proximity sensors (S) or β-algorithm (β)). It is an instance of the subsumption architecture with two loosely coupled behaviours, one having a higher priority (avoidance) than the other 2.7.1).

Firstly, the goal of the avoidance behaviour is to steer the robot away from any obstacle that it detects in a Braitenberg fashion [Braitenberg, 1984]: if it detects an obstacle from one side only, it will command the motor from the opposite side to slow down (SPIN) and hence cause the robot to turn in the opposite direction. If it detects an obstacle from both sides but none at its rear it will move backwards (REVERSE) for a while and then turn randomly (SPIN again). Finally if all three of its proximity sensors are on, the robot will stop until this situation changes (STOP).

Secondly the goal of the communication layer is to analyse the contents of incoming messages from neighbours and decide to either choose a random direction if the robot assumes it has rejoined the group (RANDOM) or to turn 180 degrees because it needs to reconnect (BACK). The subtleties of this layer will be discussed in chapter 4

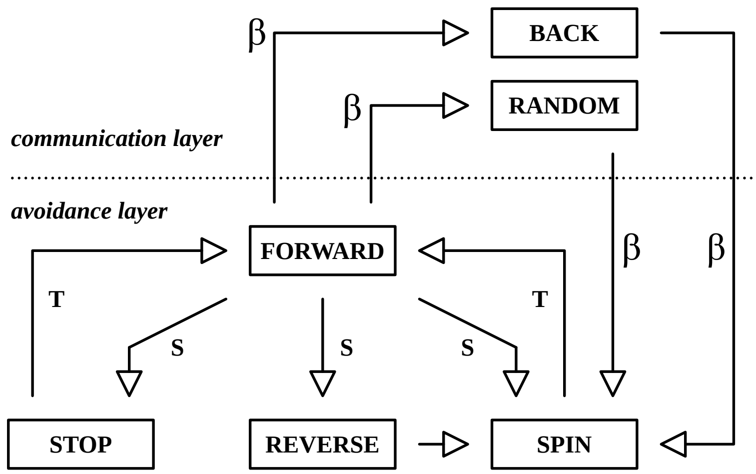


Figure 3.2: State transition diagram of a robot

What is of importance here is the fact that when the robot is performing actions dictated by the avoidance layer such as spinning, reversing or stopping, it does not take account of information from the communication layer. In Brooks' terminology, the avoidance layer *subsumes* the communication layer, because of its priority for the robot's safety.

In addition to these states, each robot keeps the following data structures:

- *Time counters*: to control the robot in real-time, we implement a general clock as a counter in the control loop. This gives the possibility of triggering behaviours in a time-related manner. Several other counters are needed to give the robot some persistence in its behaviours.

- *Local connection information*: as the upper control layer relies on information about connections between robots, the individual robot keeps a possibly inaccurate map of the connections between its neighbours. The reason that such information is critically needed will be discussed in chapter 4.

- *Receive buffer*: in order to keep the incoming messages from neighbouring robots before processing.

3.2.2 communication

Following the preliminary work of Winfield, an idealised model is used to simulate radio communication between the robots. Firstly it is assumed that the wireless antenna is omnidirectional and that the receiver is not able to detect the message if the transmitter is located further away than a distance R, the *communication range*. These are sound assumptions as the communication device is likely to use modes of modulation that experience the "threshold effect", *i.e.* the sudden inability of the receiver to demodulate the transmitted signal when the signal strength falls below the threshold. Phase Shift Keying (PSK) and Frequency Shift Keying (FSK) are modulation schemes that are likely to be used on mobile robot platforms and exhibit this effect. Secondly it is assumed that the wireless network will enable the robots to exchange information through the same medium without compromising the first two assumptions by using time, frequency or code diversity. Direct Sequence (DS) or Carrier Sense Multiple Access (CSMA) are examples of such strategies to share the same radio medium without interference [Winfield, 2000].

As a result it is not attempted to simulate buffer overflow or any other real phenomena occurring such as signal decay and it is considered that a robot can send a message to all neighbours that are within distance R. Noise is modeled as loss of the entire message with a constant probability. This probability was chosen to range from zero to 10% (which represents a very poor signal-to-noise ratio).

The situation depicted in figure 3.3 shows a group of robots together with their range of communication. In this example robot 1 can communicate with robot 2 and the latter with robot 3. Robot 4 cannot communicate with the other robots.

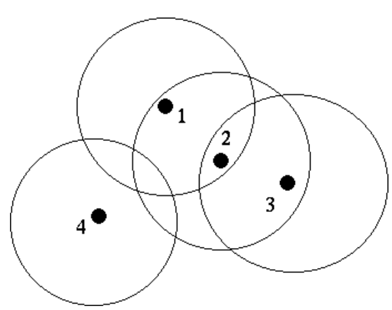


Figure 3.3: range of robots in 2 dimensions

This situation can be formalised using an *undirected graph* $G = (V, E)$ where V is a set of points, the *vertices*, representing the robots and E is a set of lines, the *edges*, connecting one vertex to another, representing the possibility for the two robots to communicate. It is possible then to draw the graph G and hence the situation of figure 3.3 can be formalised as in figure 3.4.

In the simulation user interface the robots are represented as small circles with their ID in the center and a small vector for their direction. When two robots are in range, a line is drawn between them, as can be seen in figure 3.5. It is worth emphasizing here that the position information is only known to the simulated environment; while the simulated robots themselves do not possess any such knowledge.

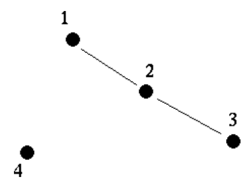


Figure 3.4: undirected related graph

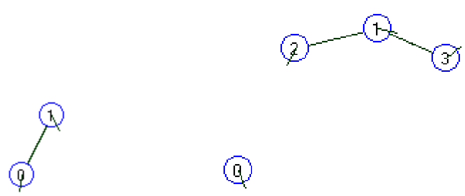


Figure 3.5: Screen examples

3.2.3 environment

As previously stated in chapter 1, the primary interest here is to study global behaviours induced by local interactions within the robot swarm. More precisely we shall seek to make the swarm attracted by some environmental cue or adopt a distinctive global shape. With these aims in mind, we introduce into the modeled environment a source of such a cue, a *beacon*. Its decay and occlusion through line-of-sight obstruction are modeled and again the design was dictated by its possible realisation, the metaphor being a bright light beacon (see figure 3.6). Noise is also introduced with probabilistic false sensor readings. The simulation also includes the option to introduce, into the arena, some occlusive obstacles in the path of the swarm, as can be seen in figure 3.7. This allows us to test the behaviour of the swarm in a more adverse environment.

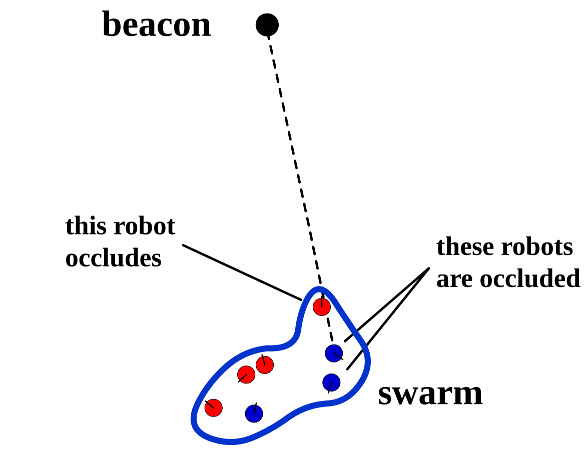


Figure 3.6: Line-of-sight obstruction

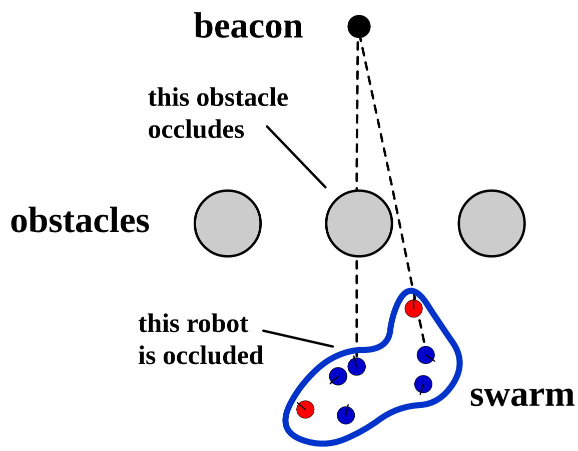


Figure 3.7: Set-up with swarm, obstacles and beacon

3.2.4 artifacts

At a human scale, the real world is a continuous time space where things do happen in parallel. But to model it, sequential and discrete time computers are used. The discrepancy between the reality and its model can lead to problems with the accuracy and efficacy of the simulation results. Indeed some behaviours of the robots observed in simulation might actually depend on the sequencing of the model or the unrealistic absolute synchronicity. These behaviours are called artifacts.

Obviously it is not possible to state with certainty that a particular behaviour is not an artifact and this is the principal reason that real robot experiments are needed. Nevertheless some design pitfalls have been identified in the literature [Harvey, 2000, Jakobi et al., 1995] as being common. One such pitfall arises from sequentiality. If robots' states are updated in sequence, this can lead to the unrealistic preference of some states over others. To avoid this in simulation,the states of the robots are updated following a pseudo-random sequence that changes at each step of the simulation.

Another artifact comes from synchronicity. However accurate they are, robots do not run synchronously unless specifically designed to do so - and this can be very time and resource demanding. Because we seek for a minimalist algorithm, asynchronous robots will be simulated. Hence the internal clock state of the robots is periodically randomly altered to avoid any unintended synchronicity.

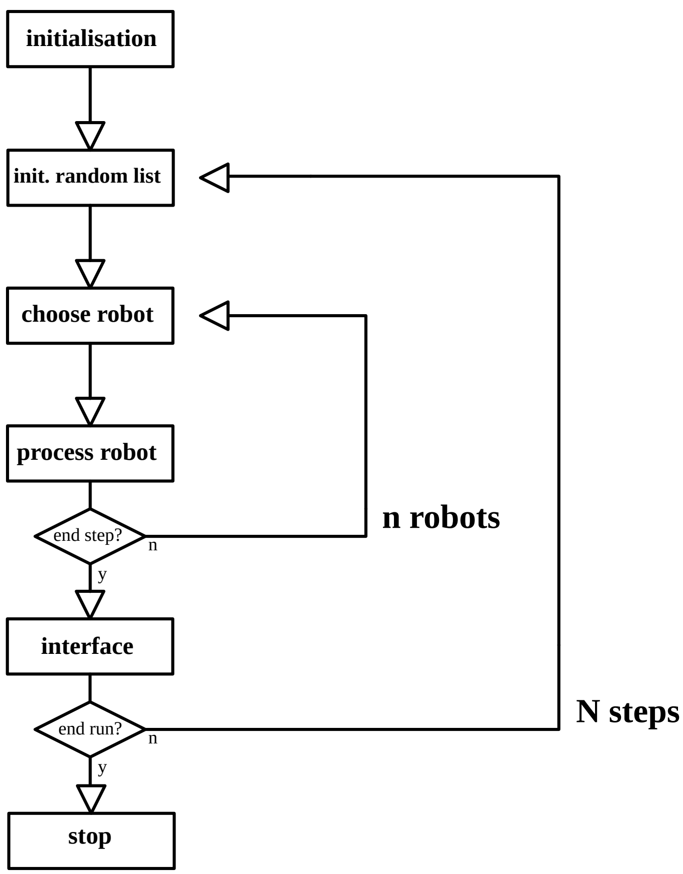


Figure 3.8: flowchart of the simulation

3.3 Real Robot Experimental Details

3.3.1 the robots

The robots used for the experimental work, the linuxbots (see figure 3.9), have two wheels independently controllable to steer the robot. They can be fitted with several sensors such as Digital Video camera, light and infra-red sensors. Robot control is carried out by an embedded PC using the IEEE PC/104 standard with a pentium class CPU. The robots have the advantage of running Linux as in a normal PC, with all its compilation and networking facilities onboard. Fitted with off-the-shelf Wireless Local Area Network (WLAN) technology, they are able to use the full potential of the TCP/IP protocols [Winfield and Holland, 2000].

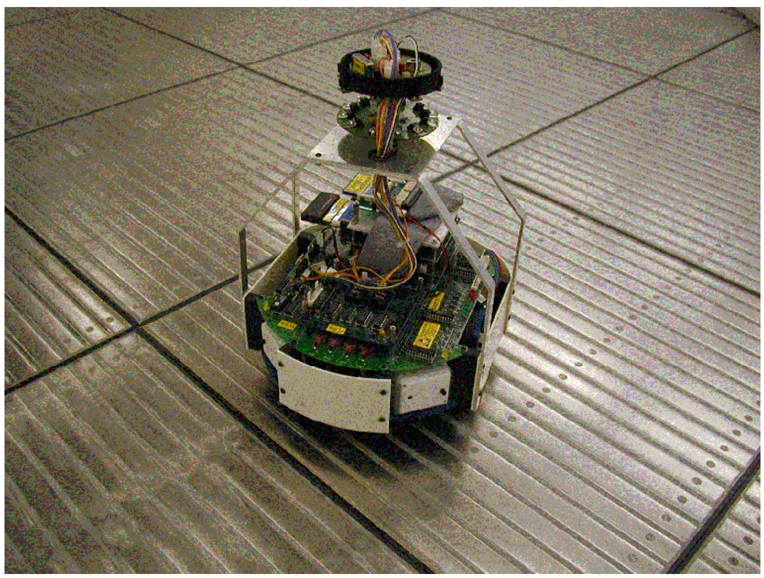


Figure 3.9: linuxbot with infra-red tower

As a result, we have a powerful research tool that can be handled as a normal remote UNIX computer through the help of telnet shells. Control programs can be written in C/C++ and compiled on board. It is a good tool for investigating behaviour-based robotics since different behaviours can be represented as different processes running concurrently and independently.

More specifically, for the experiments the robots are fitted with three active infrared sensors on the chassis for obstacle avoidance; and with a tower developed in the lab for distance measurement, consisting of an array of eight active infrared sensors (see figure 3.10). This tower is needed because of the design choice not to develop a dedicated short range radio device but instead to simulate the locality of the radio communication by taking distance measurements; then robots are able to communicate if the measured distance is below a set threshold. Reasons why such simplification of short range radio communication is realistic enough can be found in section 3.2.2.

The obstacle avoidance infrared sensors are situated as follow: two symmetrically at the front, and one at the back. They have a rather small range depending on the color, angle and texture of the reflecting surface (note the white plastic body of the robot). On average a robot detects the white surface of another if it stands perpendicularly within 50 centimeters. Unfortunately, these sensors share the same frequency and this can lead to a sensor on one robot misinterpreting an incoming beam of a sensor on another robot as a reflection of its own beam. In this particular situation a robot is detected as an obstacle with a considerably larger range.

This study investigates the potential of localised communication as opposed to global communication. The WLAN used to network the robots has a range in the order of a few hundred meters. To be able to conduct experiments with a real possibility for the radio links to be disconnected would need an arena the size of a cricket pitch. As the lab arena is 9 meters wide, a range of the order of one meter is required in order to avoid the interference of the arena's edges with the experiment. As it is not possible to tune the WLAN range by reducing the transmission power or building a Faraday cage around the antenna, we therefore faced a choice: either design a specific radio device with the desired short range or simulate this requirement by measuring distances between robots and allowing them to communicate only if within range. This latter method is referred in the literature as *virtual sensors* [Billard et al., 1999b, Billard et al., 1999a]. Indeed it reduces design and testing time as well as costs, but of course bears the inevitable disadvantages of any simulation. As previous work using local radio communication [Watson et al., 2002, Watson et al., 1999] experienced problems of limiting the range to such small distances, it was chosen not to attempt a specific design. The actual design of a truly limited range radio device is a demanding work that was regarded as beyond the scope of this PhD.

To simulate the locality of radio communication, another choice had to be made, between a solution external to the robot such as a camera above the arena with an image processing tracking software, or to use a device fitted on the robot. It was the latter solution that was pursued, firstly because this device was already available in the laboratory thanks to the work of Weßnitzer [Weßnitzer, 2001, Weßnitzer et al., 2001], and secondly, despite using virtual sensors, it still presents desirable characteristics such as: using a noisy device; creating information subjective to the robot; functioning on the same level of dynamics, and it does not involve additional communications over the network.

The infra-red tower consists of a circular array of IR emitters under a circular array of IR receivers protected from unwanted reflections by a black cardboard ring (see picture 3.10). Because of the too great range of the WLAN radios, measurements are taken through the IR tower to simulate the required locality. Here each IR tower can be set to emit at different frequencies to identify each robot and avoid the abovementioned problem of interference. However another problem is faced in the fact that, as can be seen on the picture, the design use several emitters and receivers around the tower to be able to pick up the signal in an omnidirectional manner. Because of the inevitable differences in characteristics found in electronic components, if a robot senses another at a certain range and then performs a small turn such that another receiver is facing the other robot, the measured range is likely to change. This problem can be improved by suitable calibration but consider the opposite situation: imagine that instead of the receiving robot turning, it is the transmitting one. In this case it is another "emitter" that is facing the measuring receiver and again there is no guarantee that the range measured will not change. However in this case no calibration is possible as the measuring robot has no information about the orientation of the measured robot. This will have an impact on the results of the real robot experiments.

To perform the calibration we proceed as follow: the robot to be calibrated (robot A) is placed together with another one (robot B), separated by the chosen range distance. Robot B is set to spin while the calibrating robot A takes several measurements to construct a mean of the variations in B's transmitter signal strength. Then the calibrating robot A spins for 45 degrees to construct another mean related to the next of the 8 receivers on its tower. Each possible pair of robots were engaged in this calibrating dance, resulting in a robot specific array of thresholds values indicating whether a neighbouring robot is in range or not.

In order to maintain the same ratio between avoidance range and communication range as in the simulation, the robots were calibrated with a range distance of 1.2 meters. With such range, swarms with 7 robots were already very close to the edges of the arena.

Unfortunately it appears that the IR tower used is actually very noisy and the virtual sensor result does not correspond with the basic assumptions the simulation was built upon (see section

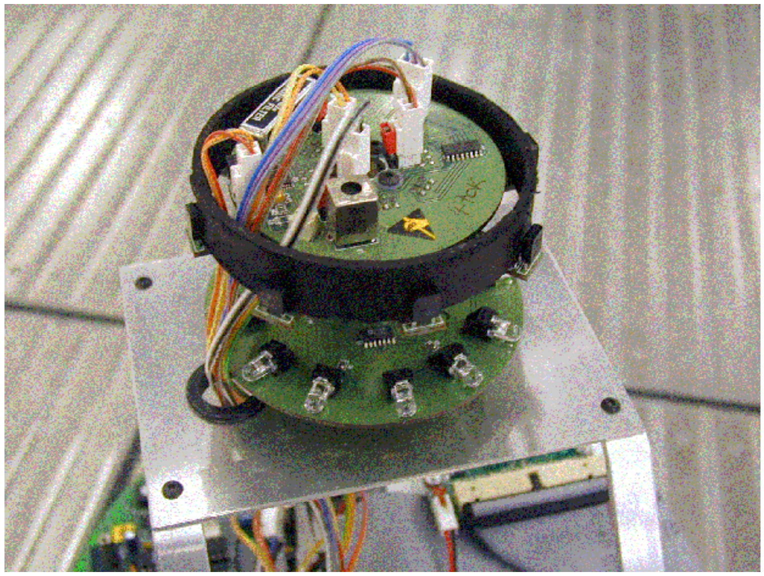


Figure 3.10: close-up of infra-red tower

3.2.2). Indeed the IR signal is not omnidirectional as the range is different in asymmetrical directions around the robot, which result in a non-circular area of communication. It is difficult to say what would be the result if a video tracking system had been chosen to simulate the locality of the radio, but we believe that such problems in symmetry would not arise and we might be in a better position to test the real behaviour of the simulated algorithms. However it is still highly valuable to study the response of the algorithm under such non-ideal conditions.

3.3.2 experimental set-up

In all experiments the environment is the IAS laboratory arena, which is a 9 meter wide octagon delimited by white edges suitable for IR reflection. The tiled floor is covered with steel strips in order to power the robots for long experiments without having to change batteries. Each robot is equipped with power pick-ups that enable it to extract the power from the floor by direct contact, gliding on the floor (see pictures 3.11 and 3.12). A previous example of such an arena can be found in [Martinoli et al., 1999a]. These gliding pick-ups exhibit some difficulty in absorbing the level differences between the tiles of the floor (black edges on the picture). Because of this some robots experienced short stops or deviations which placed extra noise on the experiment.

Each experiment started with the robots grouped near the center of the arena and lasted until

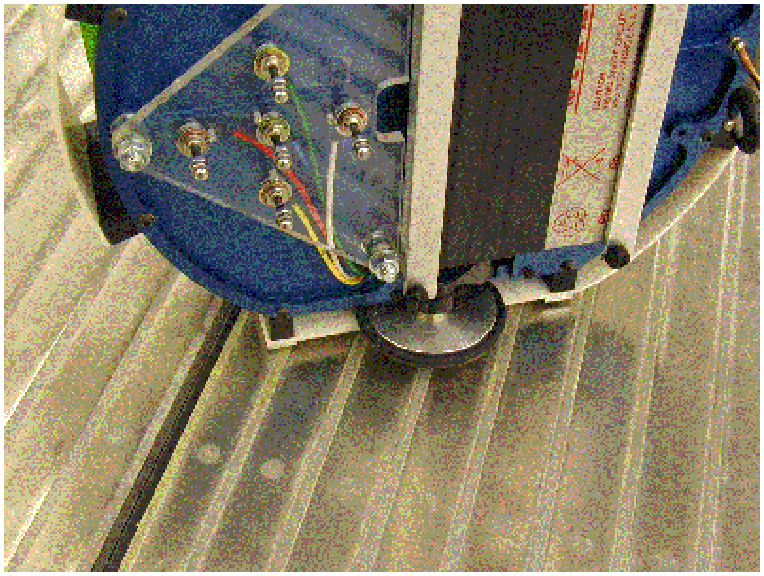


Figure 3.11: bottom side of a linuxbot showing power pick-ups (on the plexiglas triangle)

the group was obviously disconnected (see pictures 3.12 and 3.13). We recorded odometry, local connection information, state changes and video recorded experimentation runs. The starting positions were not fixed but random. The results of these experiments will be discussed in chapter 4.

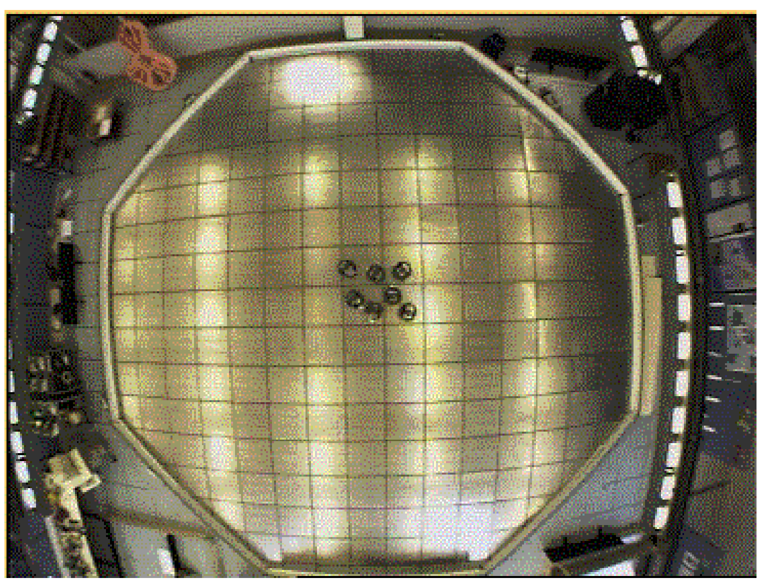


Figure 3.12: start of an experiment

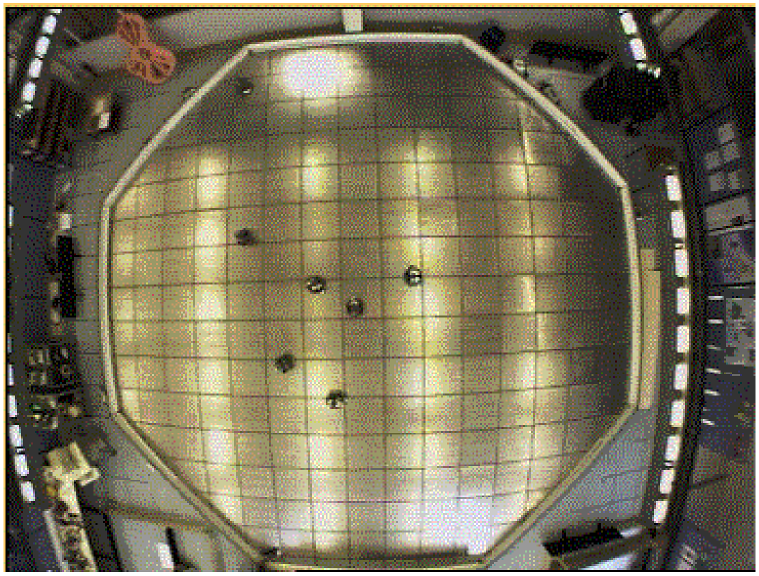


Figure 3.13: end of an experiment

3.3.3 robot processes

The real robots use the same control program architecture as in the simulation. But because the real exchange of messages happens through the local area network, there is need to use TCP/IP protocol to send packets. Although it would have been perfectly possible to establish a socket connection between each robot and send reliable TCP packets, because of the size of the arena and the range of the WLAN radios, we chose to use the unreliable User Datagram Protocol (UDP) (as the purpose is to study unreliable communications likely to show packet loss and non-reciprocity of data exchange). UDP thus more closely reflects the type of message exchange underlying this work. As a result the control software is transparent to the conditions of total connectivity present in the laboratory. This also simplifies dramatically the control software design.

Each robot concurrently runs two processes :

- the *receiver process* that catches incoming UDP messages from neighbours and feeds them, via an inter-process communication queue, to the motor control process.

- the *motor control process* that processes information from neighbours coming out of the inter-process communication queue, checks distances of neighbours, sends UDP messages to neighbours if within range, updates the internal states and controls the motors.

This division is needed because messages can arrive at any time and must find a listening process. Figure 3.14 shows the Yourdon schematic representation of the processes running on two different robots and the flow of data between them [Ward and Mellor, 1985].

It is worth emphasizing that this control algorithm architecture can be readily implemented in real applications thanks to the choice of UDP protocol.

3.4 Goals for a Connected Swarm

Following the focus on minimalist design of Melhuish [Melhuish, 1999c] and the preliminary work of Winfield [Winfield, 2000], the aim is to keep the robots as simple as possible. In this case the constraint of using radio communication is already present, so simplicity will be sought in reducing the amount of information exchanged. It is believed that coherence is achievable only with a radio device with limited range for communication and proximity sensors for avoidance. One of the primary goals for this research is to design algorithms that use a minimum of information exchange and present a control algorithm that is as simple as can be. The aim is to investigate the minimum core set of abilities for the robots to achieve a connected swarm. This can be summarised as the goal towards **minimalism**.

Another important goal in this study is to design algorithms that do not depend on a special number of robots, in other words **scalability** is sought. The solution developed needs to be efficient even in the face of an increased number of robots. This concern has a huge impact on what information is exchanged and how.

We also seek algorithms that are not prone to failure if one of the robots does not function properly or if the communication quality is poor. This is referred to in the literature as **robustness**.

As a result of the general goals mentioned above, a low-level goal in this study is **swarm coherence**, that is a group of robots that forms a network consisting of a single connected component. For the purpose of inter-robot communication to be possible at any time, a unique stable swarm is needed. A loss in this interconnection has to be constrained to a limited time in order to minimise loss of messages. Moreover the swarming mechanism does not rely on information such as relative position to allow a lost robot to reconnect. This is the reason a loss of connection for more than the time needed for a robot's return can result in a complete and irremediable loss of the robot, the focus

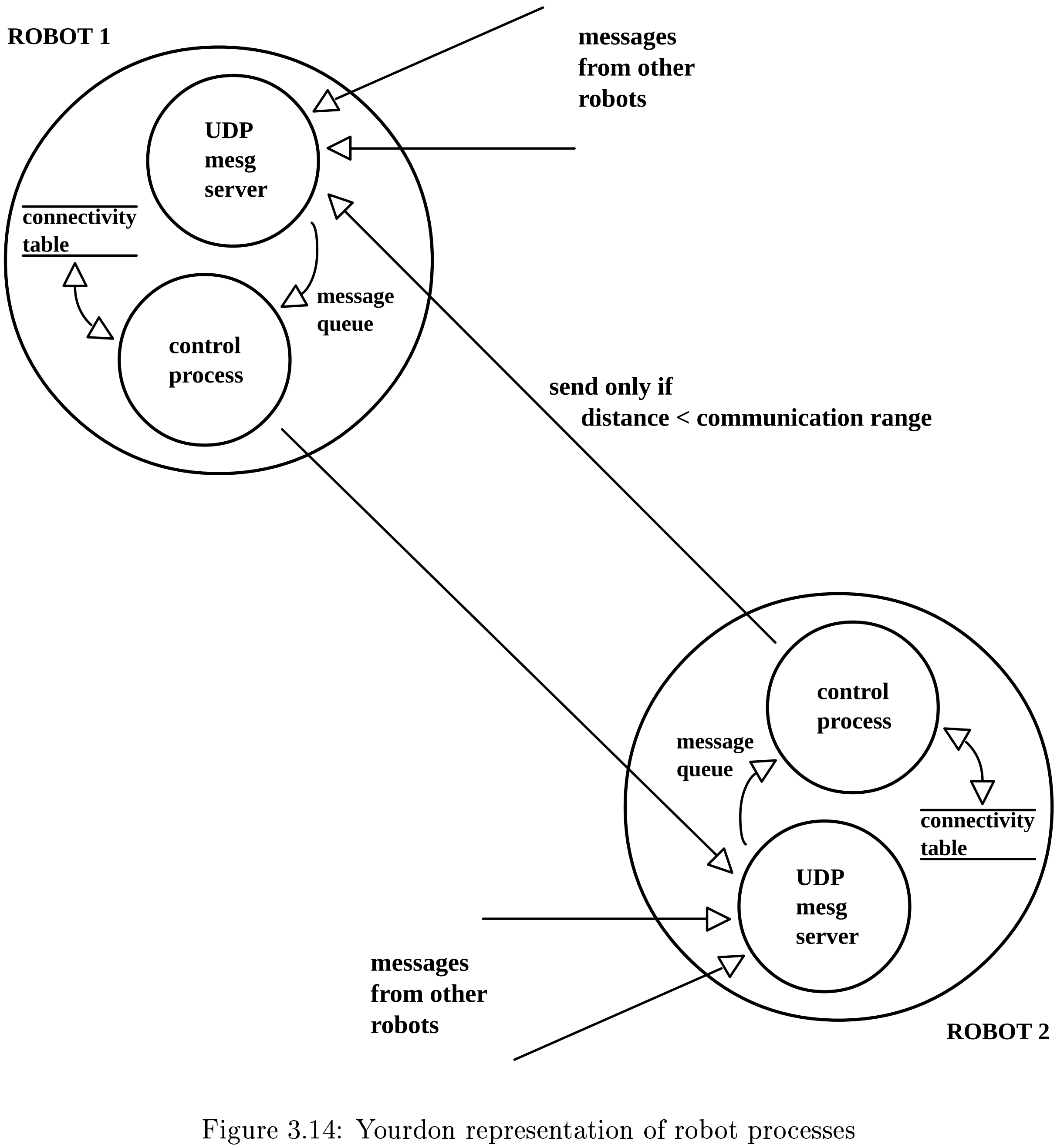


Figure 3.14: Yourdon representation of robot processes

being on unbounded environments. Coherence is therefore an absolute requirement for the swarm. This coherence is also required when the swarm is attracted by a beacon or has to avoid obstacles (see chapters 4 and 5).

Due to the underlying motivation on sensory network applications and for the sake of minimalism, we also seek to minimise the communication overhead that is necessary to achieve coherence. This is to retain maximum bandwidth for data gathering.

The second low-level goal: **maximising the area covered** by the robots derives from the underlying motivation of distributed sensing, whose principle is to take measurements in parallel at different places. Hence an over-stable group clumped tightly together would not be desirable. What is sought is the most widely-spread yet stable swarm. Therefore the area covered by a swarm is measured to quantify this goal and the influence of noise and increase in the number of robots will be investigated in chapter 4.

In the task of being attracted to a beacon presented in chapter 5, a low-level goal is the **distance traveled by the centroid of the swarm** during a certain time in the presence of obstacles or not. The influence of noise and the number of robots will also be investigated.

In chapter 6, the possibilities for minimalist control of the shape of the connected swarm are studied and the low-level goal is **to make the swarm adopt a linear shape**. Again robustness to noise and scalability are studied.

3.5 Measures

3.5.1 measure of coherence

In this study, the first feature that has to be measured is *swarm coherence*. But as the loss of a connection is a discrete event, the quantity that needs to be measured is not obvious. Should we restrict ourselves to detect when the swarm becomes disconnected, or should we look for finer criteria?

A definition of coherence would be:

> The swarm is *coherent* if any break in the overall connectivity of the network lasts less than the time constant C.

The constant C depends on the range of communication, the periodicity of the calling messages (cadence) and the speed of the robots. In this study we have chosen $C = 10$ times the cadence (see figure 3.15). After such time, the swarm is declared disconnected and does not contribute to the computation of results.

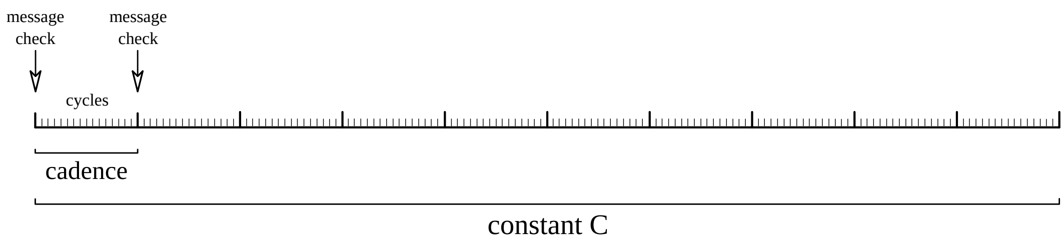


Figure 3.15: schema of cadence and C constant

But this definition is very crude and it would be of more use to have a gradation in levels of coherence. This can be provided by the graph theory concept of *n-connectivity* and *n-edge-connectivity*:

> Firstly, a graph G is *connected* if any two of its vertices can be joined by a path. Then if G is connected and, for some set W of vertices or edges, $G - W$ is disconnected, then we say that W *separates* G.
>
> For $k \geq 2$, we say that a graph is k*-connected* if either G is a complete graph K_{K+1} or else it has at least $k + 2$ vertices and no set of $k - 1$ vertices separates it. Similarly, for $k \geq 2$, a graph is k*-edge-connected* if it has at least two vertices and no set of $k - 1$ edges separates it.
>
> The maximal value of k for which a connected graph G is k-connected is the *connectivity* of G. Analogously the maximal k for which G is k-edge-connected is the *edge-connectivity* of G [Bollobás, 1998].

These two values are related by a corollary of Menger's theorem to the number of *independent*[2] and *edge-disjoint* paths [Bollobás, 1998] and therefore have a meaning in terms of robustness of the network to connection or node failure respectively. Therefore, following the underlying concern for sensor networks, the relevance of such a quantity to the ability of the network to propagate messages is another motivation for measuring it.

[2] paths from s to t that have only vertices s and t in common

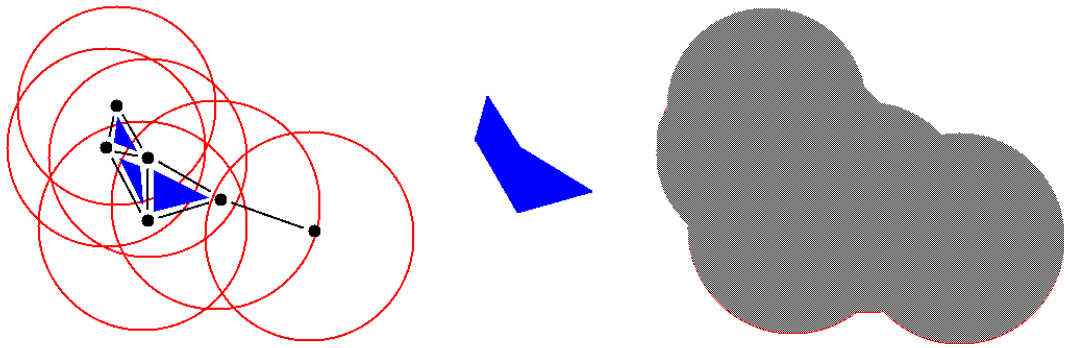


Figure 3.16: Area covered by the swarm

The algorithm used to compute both values is found in [Even and Tarjan, 75], similar computations such as number of disjoint spanning trees can be found in [Hobbs, 1989].

3.5.2 measure of area coverage

Another quantity that is important to measure is the *area coverage* of the whole swarm. This was done primarily by triangulation of the bounding polygon of the graph [Nembrini et al., 2002], as can be seen in figure 3.16 (middle). However, this metric appeared to be too sensitive to connection loss; as in the example of figure 3.16, where the removal of an edge can potentially halve the measured area coverage. Area coverage has therefore been redefined as the area covered by the radius of communication of each robot within the swarm, without overlap (see right hand part of figure 3.16); and robots disconnected from the swarm do not contribute to the measured area. This quantity is measured with an approximation by square bisection, allowing for an error of less than 5 percent of a single robot coverage, because of computational complexity. With this new definition of area coverage, a smoother behaviour of the metric is achieved.

The underlying concern on sensor networks, possibly engaged in a monitoring task where area coverage is critical, and where each agent has a local perceptual range, is another argument for the redefinition of the area coverage metric.

Area coverage can also be approximated through the distribution of degree of connections over the whole graph. This distribution depends on the radius of communication and is therefore highly correlated with area coverage.

Appendices A and B develop simple geometrical ideas in order to compute an upper and lower bound of the area coverage (see sections 8.1 and 8.2). For the upper bound, the idea is to consider the maximal area possibly covered by a group of robots where every robot is connected to all of the others. In mathematical terms the graph formed by such a group of robots is called a *complete graph* where each vertex has the same degree. Computing this area and dividing by the number of robots gives the maximal contribution of a single robot to the whole area associated with its degree of connection. By summing these contributions according to the current distribution of degrees gives an upper bound to the area covered. As each robot has some information about the distribution of degrees in their neighbourhood, such an approximation could be computed locally and become a cue to trigger behaviour.

Using simple geometric properties we compute the contribution of a single robot to the area in all possible cases of degree of connections. Then the upper bound will be given by the following formula:

$$A = \sum_{i=1}^{Nrobots} \alpha(k_i)$$

where k_i is the degree of connections and $\alpha(n)$ is a function that returns the maximal contribution according to degree n (see section 8.1).

For the lower bound, the idea is to consider that the avoidance behaviour makes the robots stay away from each other. Thus, the minimal area can be computed considering groups of robots placed on the equilateral grid of edge length R, where R is the range of the proximity sensors.

Again we compute the minimal contribution of a single robot to the global area in all possible cases of degree of connections. Then the lower bound is given by the following formula:

$$A = \sum_{i=1}^{Nrobots} \lambda(k_i)$$

where k_i is the degree of connections and $\lambda(n)$ is a function that returns the minimal contribution according to degree n (see section 8.2).

The area coverage is typically a measure that requires global knowledge but here are presented means to get a local approximation.

3.5.3 other measures

To measure success of taxis with or without obstacles, we compute the distance of the centroid of the swarm to the beacon. This is to be able to study the scalability of the algorithm as we increase the number of robots, as well as the influence of noise on the speed of the swarm.

In the case of a reactive morphology change, we measure the ratio between the added square distance of each robot to the line starting from the beacon passing through the centroid and the added square distance to its perpendicular (see figure 3.17). This gives an idea of the shape of the swarm as would the ratio of the two radii of an ellipse (chapter 6).

$$ratio = \frac{\sum_{i=1}^{N} (d_{beacon}(R_{x_i}) - R_{y_i})^2}{\sum_{i=1}^{N} (d_{perpendicular}(R_{x_i}) - R_{y_i})^2}$$

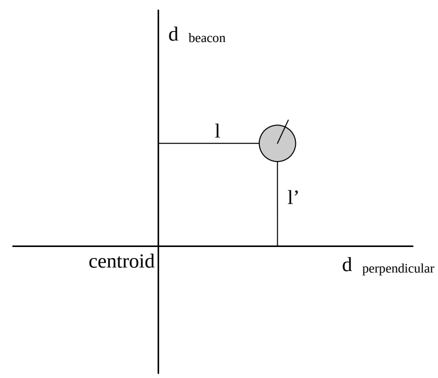


Figure 3.17: reactive morphology change measure

When the swarm is separated into heterogeneous groups meant to self-organise (see chapter 6) the group mean distance to the whole swarm center of mass is computed to measure potential differences between them.

Also the mean minimum distance between individuals in different groups is introduced to measure the distribution of the different groups (see figure 3.18). More formally it is defined as follows:

$$d_{min} = \frac{1}{|G|} \sum_{r_i \in G} \min_{r_j \in G'} (d(r_i, r_j))$$

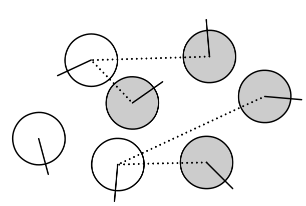


Figure 3.18: minimal distances between two groups

Chapter 4

Coherence

"(...) to shift the responsibility from making choices to asking questions."

John Cage.

In the real world, unbounded environments seem as ubiquitous as bounded ones. If it is sometimes possible to restrict the environment or assume it is bounded, this is certainly not true in general. Consider only fluid environments such as the atmosphere or the oceans. In trying to keep a group of robots together in such an environment there are two approaches: either give the robots an ability to come back towards the group or design a swarming algorithm that will not allow any robot to lose the group.

The first approach involves sensing that will generally be limited in its range, or worse that will rely on external information, such as GPS for instance, with the danger of making the swarm centralised and non-autonomous, but will in any case increase the complexity of the robots. Following the desire for minimalism presented in chapter 2, the second approach will therefore be pursued: throughout this chapter the aim is to show the capacity of the algorithm developed to achieve stable swarming of the robots. It will be referred to as *coherence*.

Extending the work of Winfield [Winfield, 2000], our investigations will be constrained to the study of the potential of communication alone to achieve coherence. Solutions will not rely on positional information either absolute or relative, also because such information is notoriously difficult to obtain.

Actually it will be shown that short range communication alone is able to generate sufficient information to guarantee the coherence of a swarm. Furthermore the influence of noise and other parameters on the overall behaviour of the robots will be investigated and the robustness and versatility of the coherence algorithms demonstrated.

4.1 Starting Assumptions

Initially the research based itself on the work of Winfield that presented a study of data gathering by a group of communication range-limited robots [Winfield, 2000]. In this research the robots move randomly in a bounded space, collect a data sample at a certain time and then send this information through the ad-hoc wireless network to a collection robot. The purpose was to investigate the influence of a completely random and constantly changing topology of the ad-hoc network. Winfield was able to show in simulation that a certain density of robots is optimal to most efficiently gather the information.

Following this work came the idea that the behaviour of the robot could be influenced by the communication information in order to keep the robots together: in the case of range-limited communication, keeping together means staying within range of each other. The goal of coherence, as defined here, is then to guarantee that the communication network forms a single connected component. As a result, in achieving such an aim, the swarm would be able to operate in an unbounded space.

Hence the starting assumptions for the research are :

- the robots operate in an unbounded space.
- they are equipped with range-limited omnidirectional communication devices as well as proximity sensors to prevent collision.
- the robots do not have any other sensing ability but proprioceptive.
- communication is two-way (duplex) but, importantly, any single message is unsolicited and does not require an acknowledgment, *i.e.* it is *broadcast*.

The preliminary aim is to be able to guarantee the coherence of the swarm using an information exchange as restricted as possible to minimise the algorithm's overhead.

This is a rather strong set of assumptions that relates to the work on minimalist robotics and distributed sensing networks. The most dramatic consequence of these assumptions is the fact that in an unbounded space a robot losing connection with the rest of the swarm and not reacting in an appropriate manner will not be able to return, lacking any long-range sensing ability. Therefore the coherence of the swarm is an absolute prerequisite for the following research of this thesis, and the present chapter describes work towards this particular aim.

4.1.1 minimalism

Minimalist robotics as presented in chapter 2 investigates the potential of robots that have strongly restricted sensing and acting abilities, and/or follow a set of minimalist behaviours. One long term aim of minimalist robotics is to find ways to build and control a large number of cheap and unreliable robots, that despite their limitations and unreliability, are still able to complete the task. Hence the present research fits well into such a framework. Other examples that follow this direction have been cited in section 2.8.1.

One could argue that communication in itself is not a "minimal" ability in terms of, for example, energy consumption or the complexity of devices involved. Our opinion is that this view only reflects the current state of the art and that examples of "minimal" chemical communication in nature lead us to think that truly minimal communication may become feasible. An early example of engineered chemical communication between bacteria can be found in [Weiss and Knight, 2000]. Alternatively, the RFID technology or the newly released ZIGBEE communication protocol could already represent "minimal" communication solutions [Artaud et al., 2004].

4.1.2 distributed networks

If this research is to be applied in the field of distributed sensing, it is of course sensible to restrict the communication overhead. In this case the bandwidth available is principally needed to distribute the sense data and a control algorithm that minimise bandwidth is certainly an advantage. The use of range-limited radio devices also limits interference in the medium due to signal decay. And of course the possibility of operating in an unbounded environment is a crucial advantage. There is a potential weakness in the sense that no positional information is available to tag the samples, but as presented in section 4.5, there is potential for an alternative way to gain some confidence in the general relative position of the robots.

The field is rather developed because of its obvious applications and includes research that concentrates on the problem of data gathering such as the work of Estrin at USC as presented in section 2.11.2 or which investigates the networking problem if the nodes are unreliable, for instance see [Bulusu et al., 2001, Foreman et al., 2003].

It is worth noting that these examples mainly look at networks consisting of fixed nodes, the dynamics being a consequence of their unreliability. The investigation of moving active nodes is studied by fewer groups, among which is the researches of Honary and Wessnitzer at the IASLab. Honary is interested in monitoring underwater currents by analysing the deformation of a swarm of diving robots [Honary and McFarland, 2003] and Wessnitzer uses information dissemination to make a group of robots collectively catch a prey or, as a sheepdog, oblige another one to move to a target [Weßnitzer and Melhuish, 2003]. Other examples include [Hackwood and G.Beni, 1992, Franklin et al., 1995]

These examples, while they do not rely on absolute positioning through the use of a GPS or equivalent device, use relative positioning or directional information to complete their tasks. Actually, most studies on coherence for swarm of agents have relied on positional information; see the well-known example of Reynold's boids for an early result and also the work on formation keeping, such as [Balch and Hybinette, 2000] An interesting exception is the work of Melhuish using chirping robots for swarming [Melhuish, 1999b], where the intensity of the chirping is used as a cue for the robot to decide whether it is going in the right direction or not. But in this case the loss of a robot is not considered critical to the success of the algorithm.

In the present research the only relative positional information is the interplay of the avoidance behaviour and the range limitation of the radio-communication. It gives a robot a very approximate idea of the location of its neighbours by telling it whether they are close, too close or not-at-all. The next section will be show how the exchange of messages between the robots is critical to transform this crude data into sufficient information to reliably achieve coherence.

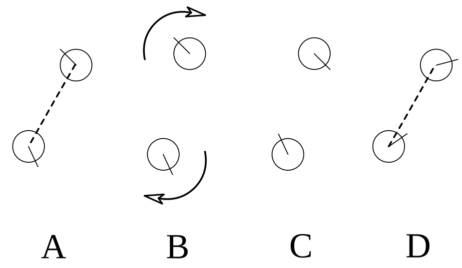


Figure 4.1: Basic algorithm

4.2 Algorithms

4.2.1 basic algorithm

This study restricts itself to use only the information on connections between robots. In other words whether a particular robot is receiving a signal from another or not. The omnidirectionality of the radio device implies that there is no positional indication about where to go in case of disconnection. To transform this limited information into good use, the ability of the robot to turn on the spot with reasonable precision is exploited: in the default state the robot moves forward. As soon as the control algorithm detects a loss of wireless connection, the robot assumes it is going in the wrong direction and turns 180 degrees to go back.

For simplicity let us restrict ourselves to the case of two robots (see figure 4.1):

Assume that the robots are initially in communication range, moving forward with random headings (A). Unless they have parallel or crossing trajectories, the former situation undetrimental to the connection and the latter being dealt by the avoidance behaviour, they will eventually lose contact (B). In order to check whether this is the case or not, the algorithm uses a send-listen mechanism: with a certain periodicity each robot broadcasts a message containing its ID to the others indiscriminately and then listens for possibly incoming messages. If no message is received within a certain time, each robot assumes it is out of range (B) and reacts immediately by turning 180 degrees in order to reconnect (C). Then as soon as it receives a message it chooses a new random heading (D).

As no global time is implemented the robots should react asynchronously. However each robot has the same range of communication so both reactions should occur within a short time, depending on the periodicity of the calling messages. This periodicity is referred to as *cadence*.

This behaviour leads to the two robots maintaining themselves in range as if they were attached

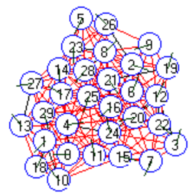


Figure 4.2: N robots aiming for a complete graph

with an elastic band. The choice of a random heading when reconnection occurs makes the pair as a whole follow a random walk. It is important to observe that the reciprocity of reaction, even though it does not have to be simultaneous, is crucial to retaining the connection. Homogeneous robots have equal velocities and the reaction of only one robot could lead to an endless pursuit.

Following Støy, it is worth emphasizing here the fundamental characteristic of this paradigm to achieve swarming. It is not the semantic content of the message alone that matters but the message being tied with an environmental cue, that is the presence -or absence- of a message [Støy, 2001c]. This algorithm thus stands in the framework of *situated communication* where the semantic content of a message is closely linked to an environmental meaning.

4.2.2 connection degree algorithm: α-algorithm

Applying this basic algorithm to a greater number of robots by making each robot react to every loss of connection leads to an over-reactive swarm which clumps together. To react to every loss of connection is equivalent to aiming towards a complete graph where each vertex is connected to every other. This is not the aim. Figure 4.2 depicts the result of applying the basic algorithm to a swarm of N robots.

So the problem for the robot is in choosing which loss of connection not to react to. In order to discriminate between different messages, these must be somehow tagged with the sender's ID. A first suggestion is that robots periodically sending messages containing their respective IDs only, this would enable a receiving robot to count how many others are in range, referred to as the *connection degree* (remember that message collision does not need to be taken into account, see section 3.2.2 for discussion).

A possible algorithm for choosing when to react thus consists of giving each robot a threshold α on the number of connections, and making the robot react if this number falls below α. The pseudo-code of the α-algorithm is set out in figure 4.3. This solution is close to the approach adopted by Støy in [Støy, 2001c]. Actually Støy's algorithm relies on differences in the number of neighbours without the help of thresholds. The algorithm is straightforwardly implemented on real robots (LEGO mindstorms), using very crude infra-red communication and suffering therefore from very unreliable communication channels. It is tested in a bounded office corridor and Støy is able to show that the algorithm makes the robots stay together more than random movement alone would allow, but the coherence of the network is not assured.

measures

The measures used to test the potential of the *alpha*-algorithm are the following (see chapter 3):

- *number of successful runs*, a successful run being a run that ends with a connected swarm.
- *edge-* and *vertex-connectivity*, which represent the minimal number of edges (respectively vertices) that have to be removed to disconnect the network.
- *area coverage*, which is the measure of the area spanned by the communication radius of all robots and *normalised area coverage* which is this value divided by the swarm size.

Each simulation run lasted 50,000 time steps with a cadence value of 100. Hence, each robot sent 500 messages. Each of the measures above was recorded once every 1000 steps and has been averaged over the whole run. For a number of values of swarm size and α, 10 runs have been performed and the result depicted is the mean value over these runs with its standard deviation. When the purpose of the investigation is not to vary them, the chosen values for the remaining parameters are shown in table 4.4.

```
Create list of neighbours for robot, Nlist
k = number of neighbours in Nlist
i = 0

loop forever {
  i = i modulo cadence

  if (i = 0) {
    Send ID message

    Save copy of k in LastK
    k = number of neighbours in Nlist

    if ((k<lastK) and (k < alpha)) {
      turn robot through 180 degrees
    }
    else if (k > LastK) {
      make random turn
    }
  }

  Steer the robot according to state
  Listen for calls from robots in range
  Grow Nlist with neighbours IDs

  i++
}
```

Figure 4.3: pseudo-code for α-algorithm

size	20 or 60
random noise	2%
α	6
steps	50,000
runs	10

Figure 4.4: general parameter values for α-algorithm

results

Figures 4.5 and 4.6 depict the results in edge-, vertex-connectivity and normalised area coverage with increasing values of the α threshold for swarms of 20 and 60 robots. The most striking feature is the drop of connectivity between a swarm of 20 robots and a swarm of 60 (figure 4.5). Also the parameter α show the ability to control the area coverage of the swarm: an increase in α corresponds to a densification of the swarm (figure 4.6). This ability seems to level down as the parameter value increases. It is important here to mention that the results depicted in these figures are only results of successful runs, *i.e.* runs that ended with a connected swarm.

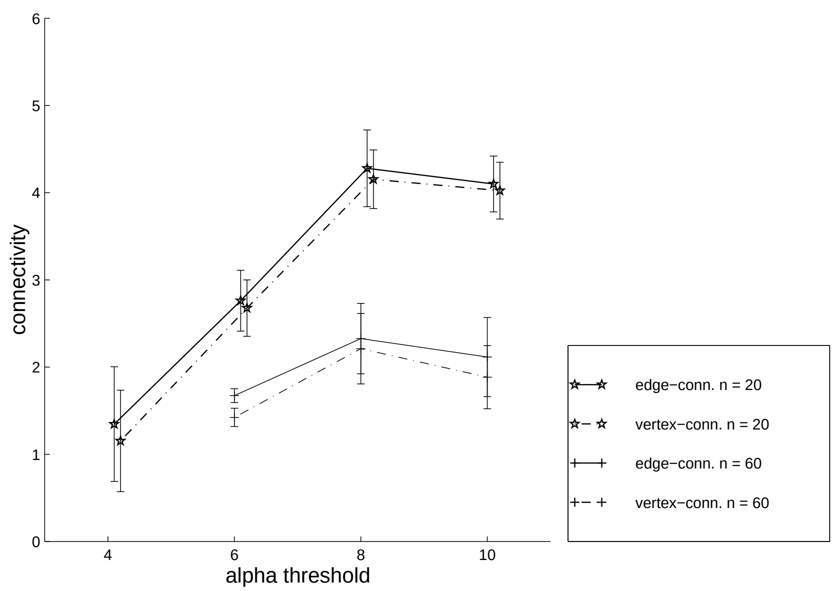


Figure 4.5: edge- and vertex-connectivity for *alpha*-algorithm

The proportion of successful runs for each value of α is depicted in figure 4.7 and shows high

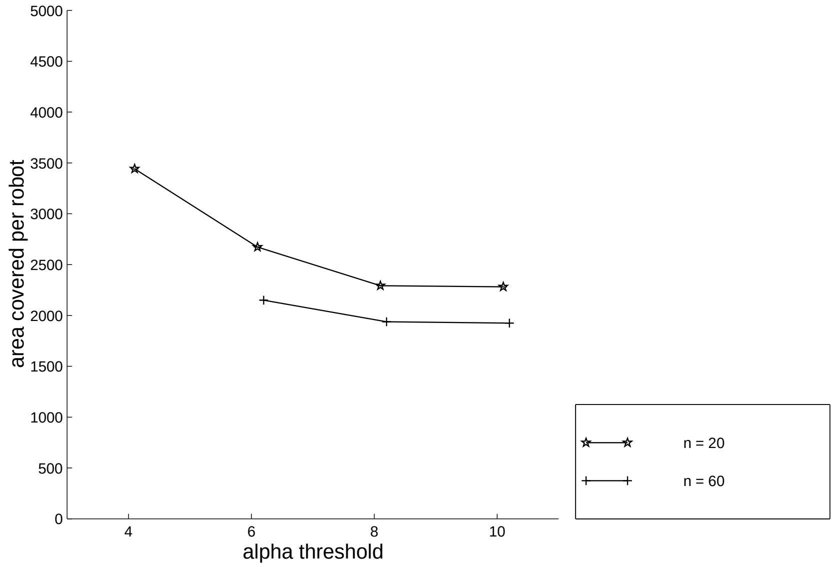


Figure 4.6: normalised area coverage for α-algorithm

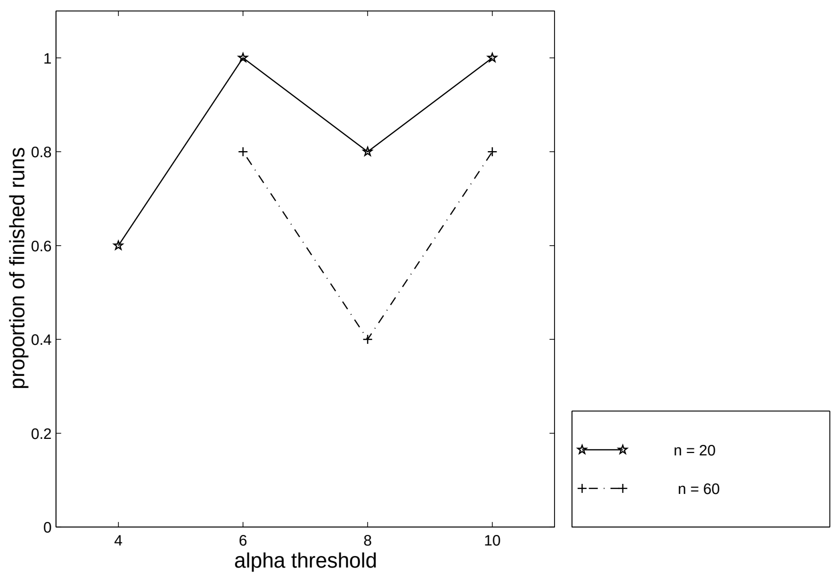


Figure 4.7: proportion of finished runs for α-algorithm

variability, apparently independent of swarm size.

The ability of the parameter alpha to control the area coverage is not surprising. It follows from the fact that increasing the number of connections sought by a robot makes it stay close to its neighbours. As a result the overall swarm density increases. This phenomena occurs up to the point where the avoidance behaviour takes over to prevent further densification.

What is more unexpected is the drop of connectivity between swarm size in figure 4.5. In order to investigate this more closely, the parameter α is fixed to the value of 6 and the size of the swarm varied. The results are depicted in figures 4.8: the drop of connectivity can clearly be followed as the size increases. It is also interesting to note in figure 4.9, that the area covered per individual robot decreases slightly. This is due to the way area is computed: because of overlapping communication area, robots that are within the swarm do not contribute as much as robots that are on the edge. When the swarm size increases more robots find themselves within the swarm, hence the decrease of the area covered per robot. Again these results are from successful swarms only.

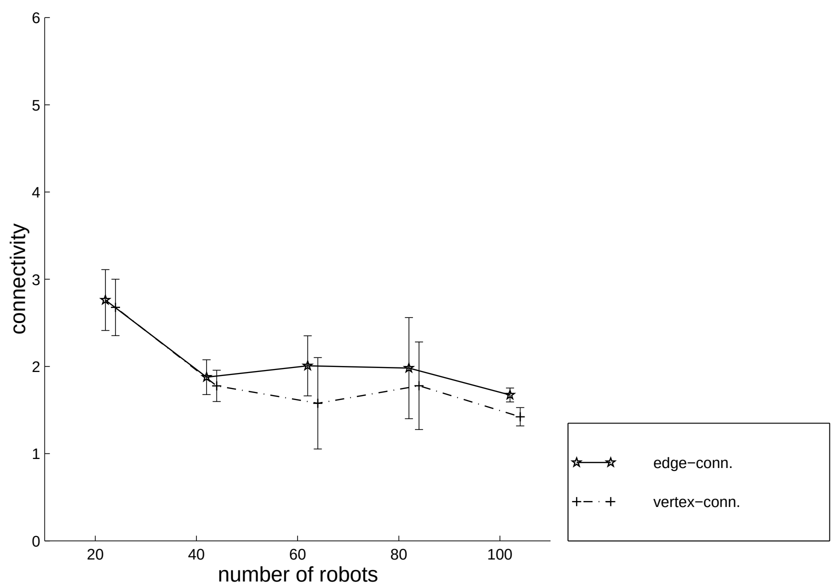


Figure 4.8: connectivity with increasing swarm size, α-algorithm

As can be seen in the result for the α-algorithm, the coherence of the network is more or less maintained for thresholds $\alpha = 6$ or above. But there seems to exist special network configurations that must be avoided in order to assure the coherence of the swarm (see figure 4.10). When a robot (or a group) is linked to the rest of the swarm by a single communication link, the danger lies in the possibility of a robot not reacting to the loss of such a connection essential to global connectivity, because the number of remaining connections is above the threshold. In graph theory an edge representing such an important connection is known as a *bridge*. In the case of a vertex that is essential for connectivity, it is called a *cutvertex*: in such a situation, a robot failure would lead to disconnection (figure 4.11).

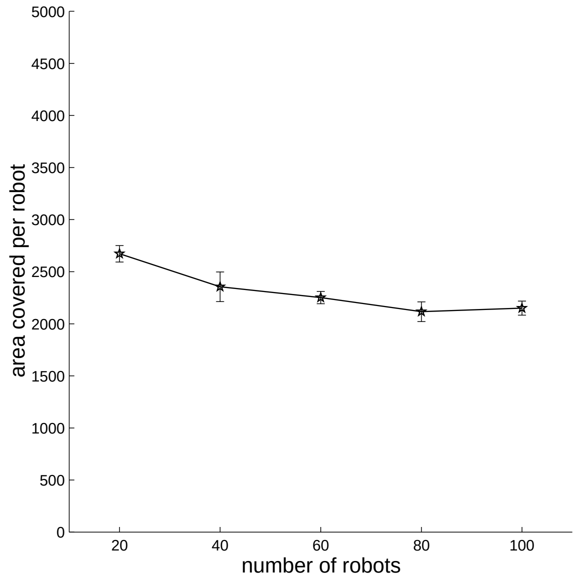


Figure 4.9: normalised area coverage with increasing swarm size, α-algorithm

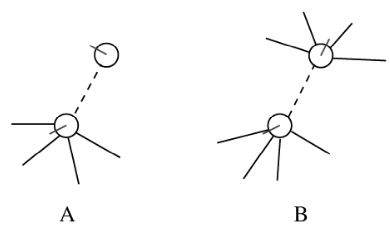


Figure 4.10: extreme states: bridges

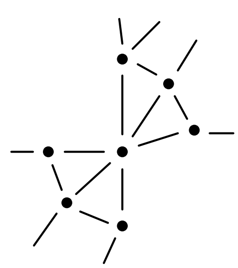


Figure 4.11: extreme states: a cutvertex

This is the explanation of the drop in connectivity observed: the measures of edge- and vertex-connectivity are global measures. This means that a single change in the network topology can potentially lead to a decrease in the connectivity value. The appearance of a cutvertex for instance will drop the vertex-connectivity to a value of 1. With an increasing swarm size, the probability of the occurrence of such situations increases. In fact the connectivity measures represent the resilience of the network to component failure: the edge-connectivity value is the number of connections - regardless which one - that can be lost without disconnecting the network, whereas the vertex-connectivity value represents the number of nodes that can be removed without disconnection (a node removal will happen, for instance, in the case of a robot network failure).

It is worth here reminding the reader that our main concern is the coherence of the swarm. Because of the dynamicity of the swarm, this is a constant danger; whereas, and again for dynamicity reasons, the probability that a robot failure occurs when being in a cutvertex situation is much lower. Therefore the solution presented in the next section deals primarily with bridges and only indirectly with cutvertices.

4.2.3 shared neighbour algorithm: β-algorithm

To avoid the extreme configurations of figure 4.10, we make use of the graph theory concept of *clustering*: instead of considering only its own degree of connection to trigger a reaction, each robot will receive from its neighbours their adjacency table - their neighbours' list - in order to check whether a particular neighbour is *shared* by other ones, that is whether a particular neighbour is the neighbour of other robots.

The algorithm works as follows: for each lost connection a robot checks how many of its remaining neighbours still have the lost robot in their neighbourhood. If this number is less than or equal to the fixed threshold β, the robot turns around and comes back. In parallel if its degree of connections is rising the robot chooses a random heading.

For instance in the situation of figure 4.12, robot A, when losing the connection with robot B, will check its other neighbours and finds that robots C and D share B as neighbour. Hence A will react and turn back only if the threshold β is set equal or greater than two. The algorithm thus makes the robot try to maintain the triangulation observable in the figure, therefore avoiding critical states. The pseudo-code of the β-algorithm for one robot is set out in figure 4.13.

It is interesting to note that the robot tries to maintain β shared connections with each neighbour and one might think that this would lead to over-connectivity, as described earlier. But in fact each connection can contribute to different sharings and such a condensed clustering is never reached. On the other hand if the robot for instance establishes a new link with a robot that does not have other

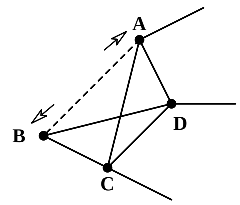


Figure 4.12: Shared neighbour

connections with the robot's surrounding neighbourhood, it will react to the loss of this neighbour until the shared connections are also established. This is precisely the behaviour that is sought.

The β-algorithm also prevents the formation of cutvertices as long as β is greater or equal to 2. Indeed in the situation depicted in figure 4.14 a cutvertex is about to be formed. But the presence of a cutvertex implies that there can only be one shared neighbour. Hence a reaction to prevent a cutvertex providing $\beta \geq 2$.

Simulation confirms that the β-algorithm increases swarm coherence, as the triangulation is perfectly observable and therefore critical states avoided. A value of β equal to one is enough to achieve coherent spread. Of course the communication bandwidth of the whole process is somewhat increased compared to the basic algorithm, as well as the processing power needed for the robot. More sensitivity to the message content (semantics) is also introduced. However, the communication is still situated, and hence message loss or misinterpretation only leads to over-reactivity without loss of robots, as the introduction of noise into the simulation confirms. Also this increase in bandwidth does not affect the scalability of the algorithm as it concerns only exchanges between neighbouring robots and will therefore not be propagated more than a single hop in the network.

If we set M as the maximum number of neighbours, then the length of a sent message will not be greater than M^2 IDs (M neighbours with M neighbours themselves). Considering the robot's body and the communication area, an approximation for M can be computed as follows:

$$M = \left\lceil \frac{A_{comm}}{A_{body}} \right\rceil - 1$$

where A_{comm} is the communication area, A_{body} is the area of the robot body and $\lceil \quad \rceil$ is the upper rounded integer function.

It is worth here stressing that the results obtained with the increase of information exchange needed by the β-algorithm, could not be achieved in the framework of the α-algorithm. Indeed the

```
Create list of neighbours for robot, Nlist
k = number of neighbours in Nlist
i = 0

loop forever {
  i = i modulo cadence

  if (i = 0) {
    Send ID message

    Save copy of k in LastK
    Set reaction indicator Back to FALSE
    k = number of neighbours in Nlist
    Create LostList comparing Nlist and OldList

    for (each robot in LostList) {
      Find nShared, number of shared neighbours
      if (nShared <= beta) {
        Set reaction indicator Back to TRUE
      }
    }

    if (Back = TRUE) {
      turn robot through 180 degrees
    }
    else if (k > LastK) {
      make random turn
    }

    Save copy of Nlist in Oldlist
  }

  Steer the robot according to state
  Listen for calls from robots in range
  Grow Nlist with neighbours IDs and connection info

  i++
}
```

Figure 4.13: pseudo-code for β-algorithm

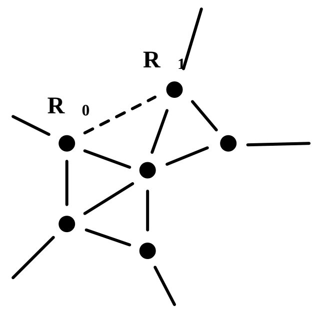


Figure 4.14: a cutvertex about to form

second order information (the neighbours of a neighbour) is crucial to detect bridges and cutvertices before they form. This follows from the fact that the first order information of one robot connected to another is not sufficient to determine the nature of this connection in terms of the connectivity of the neighbouring network.

4.3 coherence

It has been shown that the α-algorithm is not able to guarantee the coherence of the swarm, mainly because of its fundamental conceptual flaws. The following section will show that the upgrade to the β-algorithm is itself able in (possibly noisy) simulations and to a certain extent in real robot experiments, to guarantee this coherence.

4.3.1 simulation measures

The measures used to test the potential of the β-algorithm to achieve coherence are as follows (see chapter 3):

- *number of successful runs*, a successful run being a run that ends with a connected swarm.
- *edge-* and *vertex-connectivity*, which represent the minimal number of edges (respectively vertices) that have to be removed to disconnect the network.
- *odometry* which records the proportion of time spent in the different states (forward, turn, backwards and stop) over the whole run.

Each simulation run lasted 100,000 time steps with a cadence value of 100. Hence, each robot sent one thousand messages. Each of the measures above was recorded once every 1000 steps and has been averaged over the whole run. For a number of values of swarm size, β, cadence and noise, 10 runs have been performed and the result depicted is the mean value over these runs with its standard deviation. When the purpose of the investigation is not to vary them, the values chosen for the remaining parameters are shown in table 4.15.

size	20
cadence	100
random noise	2%
β	2 or 5
steps	100,000
runs	10

Figure 4.15: general parameter values for β-algorithm

4.3.2 simulation results

Because of the interesting features of the β-algorithm, a much more extensive investigation of the parameter space has been conducted, in order to show that the upgrade to the β-algorithm (and the corresponding increase in information exchange) is justified and leads to an effective improvement.

Figure 4.16 shows the variation of the edge-connectivity with an increase in β threshold and swarm size. The drop in connectivity observed in the results of the α-algorithm is no longer seen. In contrast the β-algorithm shows good constancy of connectivity against swarm size increase. Figure 4.16 also shows that the connectivity increases sharply with increasing β, then levels out at β values between 5 and 10.

The reason for this leveling is due both to the avoidance behaviour that comes into action, and the fact that, with high values of β, aggregation takes some time to complete. The typical middlerun configuration is a highly connected swarm with a few satellite robots that need some time to clump with the rest of the swarm. Because of the global nature of the connectivity measure these satellites can lead to a slight underestimate of the connectivity, that we would not observe with much longer

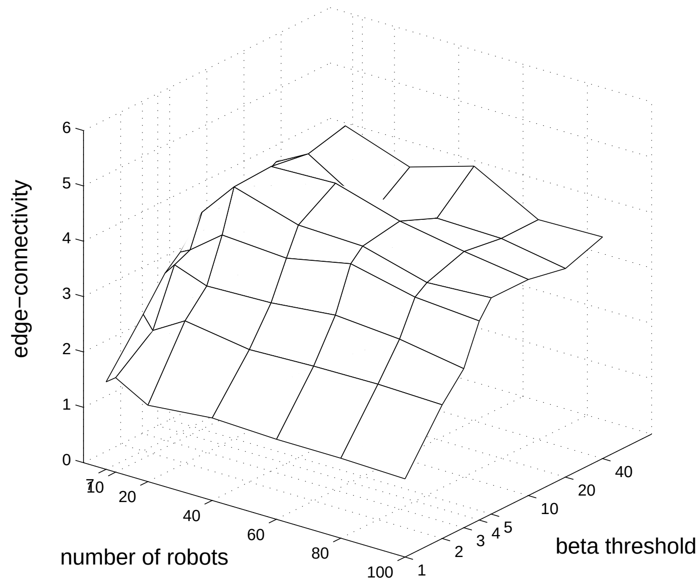


Figure 4.16: connectivity against β value and swarm size

runs.

Also, smaller swarms seem to present a slightly lower performance, especially with higher values of β. The reason is the higher dynamicity of smaller swarms that makes them more brittle in their behaviour (figure 4.17). The vertex-connectivity measure, almost equal to the edge-connectivity one, is not shown for readability purpose. In figure 4.18 the increase of performance due to the β-algorithm is obvious with most of the runs completing with a connected swarm. The value $\beta = 1$, that is keeping one shared neighbour, is insufficient for reliable coherence, whereas a value of 2 is already very good.

Figure 4.19 plots the variation of odometry against swarm size and β threshold. In this picture, the surface for the "stop" state is not shown for simplicity, but it is clear from the others states that the proportion of time spent in the stop state is very low. The higher dynamicity of smaller swarms is clearly seen on the diminishing curvature of the "turn" state surface along the swarm size dimension. This curvature corresponds to an increasing curvature of the "forward" state surface.

The next step is to test whether the desirable qualities shown by the β-algorithm are conserved with increasing levels of noise and the results are depicted in figures 4.20, 4.21 and 4.22. Note that these results have been obtained with increasing level of noise in communication, proximity sensors and actuators simultaneously. Indeed a drop in connectivity with increasing noise can be observed

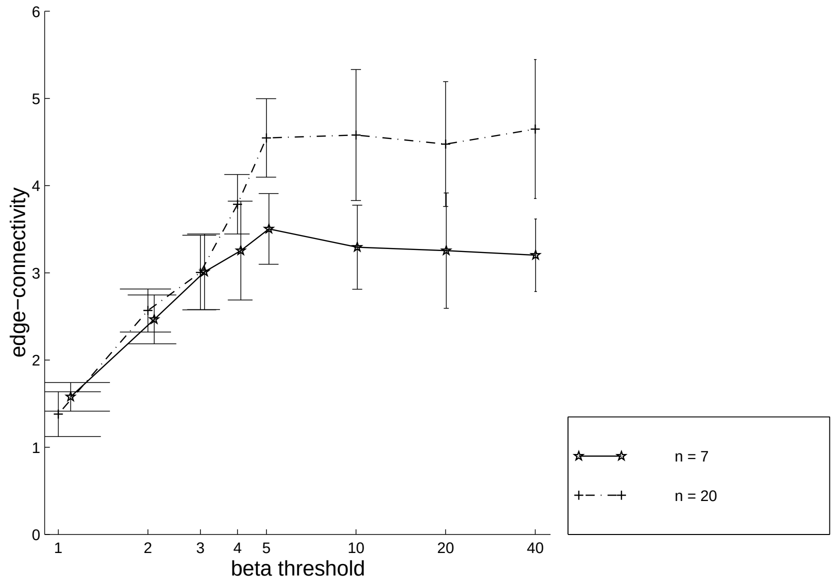


Figure 4.17: edge-connectivity with increasing β threshold

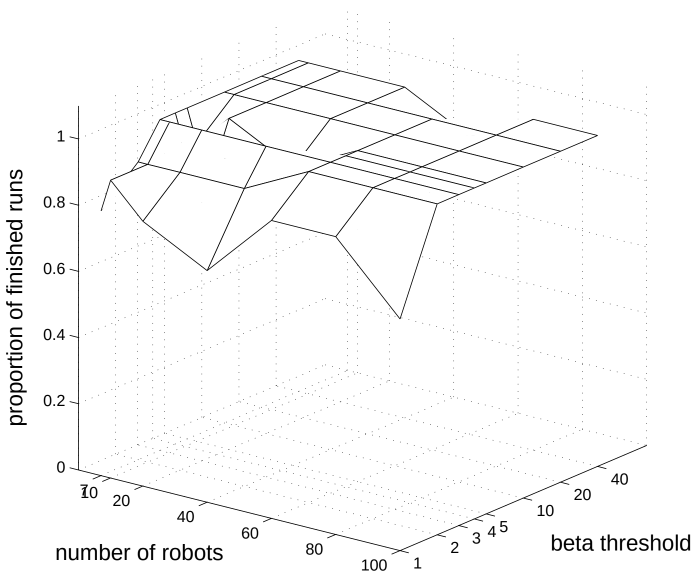


Figure 4.18: successful runs against β value and swarm size

in figure 4.20, but this is hardly noticeable for $\beta = 2$. Meanwhile, there is no difference in the proportion of success (figure 4.21). Taking a look at the odometry in figure 4.22, an almost linear decrease of the proportion of the "forward" state is observable, with a corresponding increase of the "turning" state.

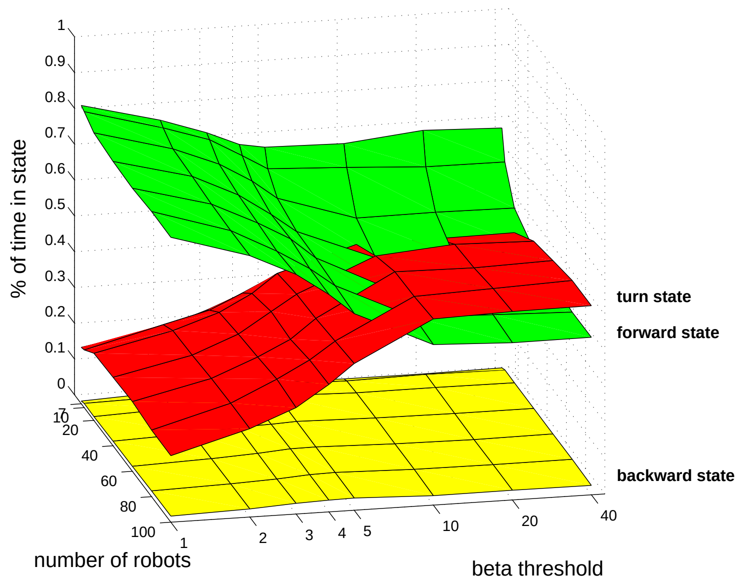


Figure 4.19: odometry against β value and swarm size

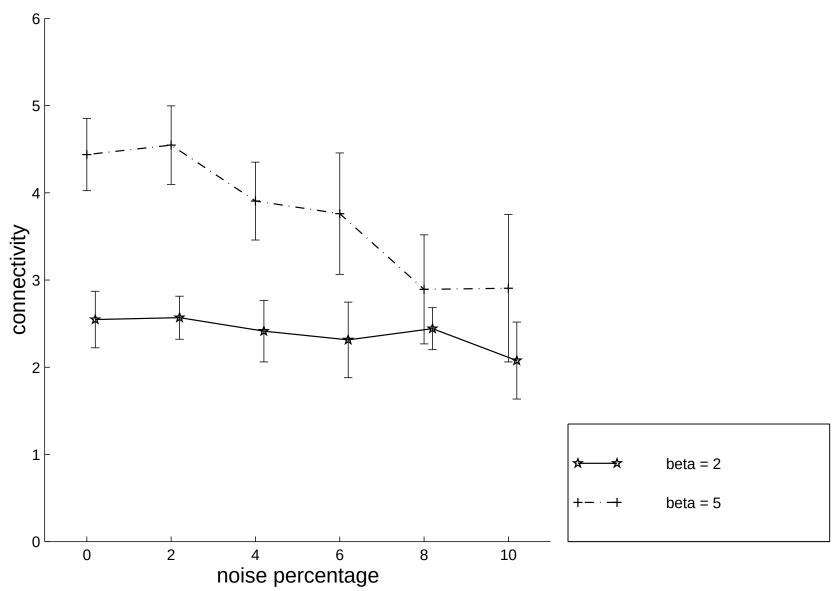


Figure 4.20: connectivity against noise, β-algorithm, (n=20)

The introduction of noise directly decreases connectivity with the loss of messages. Therefore the performance of the algorithm with $\beta = 2$ in maintaining the connectivity at a high value is impressive. In fact, with increasing noise, a robot experiences more disconnection and as a result its reactivity increases. This maintains the connectivity. With its higher threshold value, a swarm

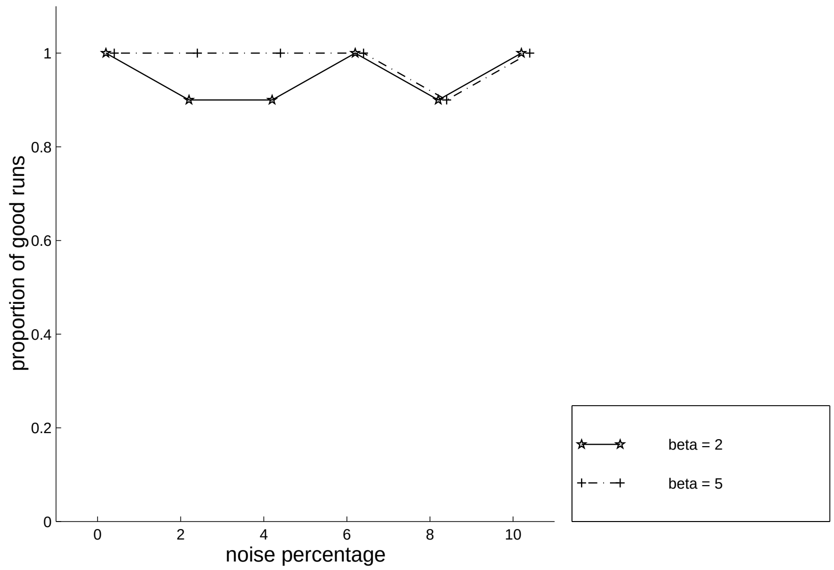


Figure 4.21: proportion of successful runs against noise, β-algorithm, (n=20)

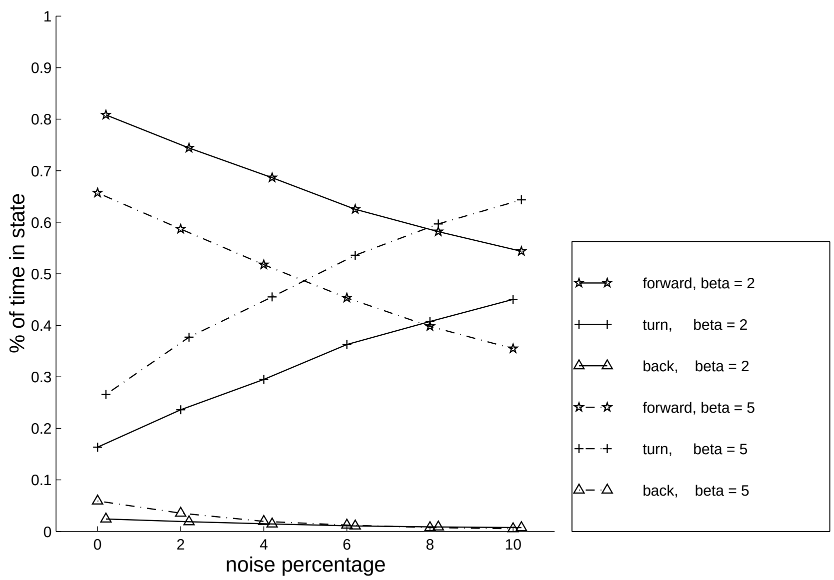


Figure 4.22: odometry against noise, β-algorithm, (n=20)

with $\beta = 5$ is already more reactive than $\beta = 2$, the potential to then absorb the direct decrease of connectivity due to noise is thus lower, and the decrease in connectivity with increasing noise is evident.

Results for area coverage show that the coverage per robot increases slightly for $\beta = 5$ with increasing noise, suggesting that higher levels of noise will lead to disconnection (see section 4.4

figure 4.47, page 122). This is indeed what happens in the real robot experiments. Nevertheless the behaviour of the β-algorithm in the presence of increasing levels of noise is arguably very satisfactory.

Now we investigate the influence of the cadence C in which messages are sent. Plotting connectivity against cadence shows an increase until $C = 150$ where a sudden drop is observed (figure 4.23). Different values of β show the same behaviour.

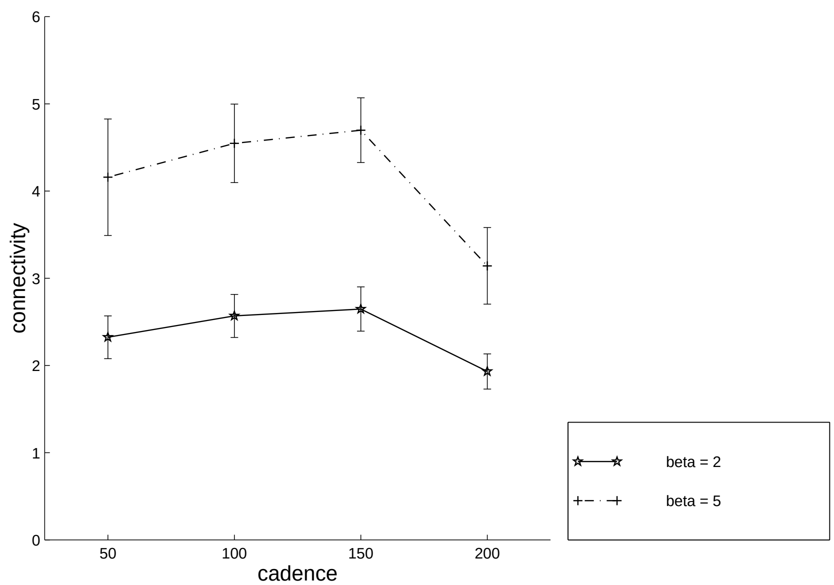


Figure 4.23: connectivity against cadence, β-algorithm

Looking at the odometry in figure 4.24, the proportion of the "forward" state increases as expected: a robot turns to react to the absence of incoming messages after a cadence lapse; if the length of the lapse increases the time spent going in a straight line must increase as well. But such an increase is limited and the figure 4.23 drop corresponds to a plateau.

These two figures suggest an optimum value for the cadence in terms of connectivity. This optimum lies between $C = 100$ and $C = 200$. It is most likely near the value $C = 150$ as this corresponds to the highest value of the connectivity and to the start of the plateau in the odometry.

Now a robot needs 180 time steps to perform a U-turn (one time step per degree). Figure 4.23 shows that connectivity is highest when the value C is below this number, which means that the robot, because of the steering layer priority, is not going to react to the next output of the β-algorithm. Indeed the communication layer has an influence only if the robot is in the "forward" state (see schema 3.2). The value $C = 50$, which allows for ignoring two following outputs of the

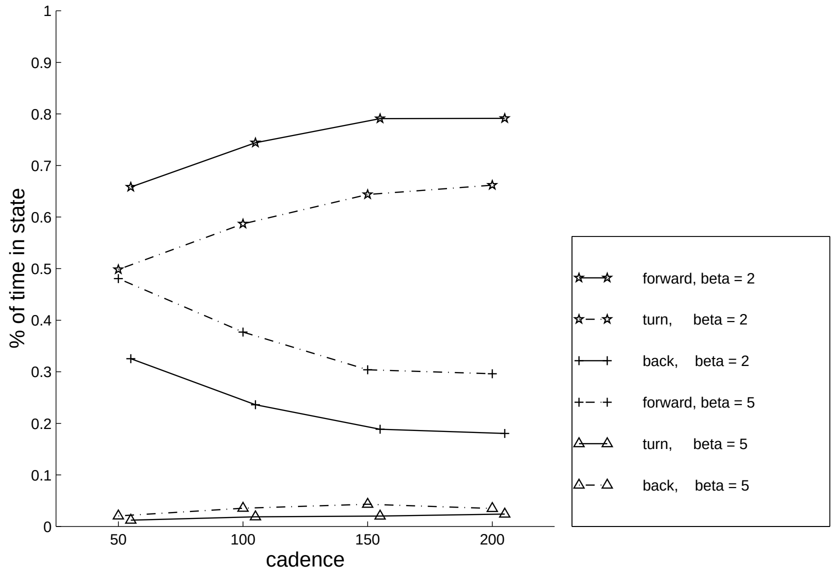


Figure 4.24: odometry against cadence, β-algorithm

β-algorithm, does not perform as well.

Thus the explanation for the drop at $C = 200$ is that immediately after performing the U-turn, the robot gets a new stimulus without having had the time to recover the lost connection that triggered the first reaction. In the worst case a second U-turn will be triggered, that will put the robot back into its initial direction, which is clearly harmful to the connectivity.

It follows from this analysis that the β-algorithm makes good use of a *refractory state*, ignoring one of its outputs after reacting, but that this state should not last more than one network stimulus.

4.3.3 real robot experimental measures

The real robot experiments have necessarily proceeded slightly differently from the simulations. Instead of leaving a run to continue, even in case of unrecoverable disconnections, the run was stopped whenever the disconnection was visually obvious. This was necessary because of the restricted size of the arena, allowing a lost robot to return to the swarm by bouncing back from the arena boundary, negating the assumption of an unbounded arena.

The performance of the real robot experiments was far below the performance observed in simulation in that disconnection always eventually occurred. Hence the measures used to test the performance of the β-algorithm in the context of real robot experiments are modified as follows (see chapter 3):

- *mean length of the runs.* This is a measure of the performance in maintaining connectedness, as the run was stopped when the swarm was disconnected.
- *edge-connectivity*, which represents the minimal number of edges that have to be removed to disconnect the network.
- *odometry* which records the proportion of time spent in the different states (forward, turn, backwards and stop) over the whole run.

Each value was recorded every time a message was sent and has been averaged over the whole run. For a number of values of swarm size, β, and cadence, an average of 10 runs have been performed and the result depicted is the mean value over these runs with its standard deviation. The number of runs for each parameter value investigated can be found in table 4.26. When the purpose of the investigation is not to vary them, the chosen values for the remaining parameters are shown in table 4.25.

The attention of the reader is drawn to the fact that the computation of the connectivity presents a serious conceptual problem: as the real robots are asynchronous finite state real machines, there is no global "tick" to allow for a meaningful sample of the topology of the network. The measures were recorded onboard of each robot and the inevitable drift between the robots' processing cycles weakens the accuracy of the connectivity measure. As will be shown in section 4.3.5, the robots are also heterogeneous, which makes the computed value of coherence even more difficult to determine.

size	7
cadence	100
β	2
length	variable
runs	table 4.26

Figure 4.25: general parameter values for β-algorithm

swarm size	β	cadence	nb. of runs
2	2	100	12
4	2	100	10
6	2	100	15
7	0	100	8
7	1	100	8
7	2	100	10
7	3	100	11
7	4	100	8
7	5	100	9
7	6	100	10
7	2	50	8
7	2	150	9
7	2	200	10

Figure 4.26: number of real robot experiment runs

In the real robot experiments, the robots run the same algorithm as the simulated robots, therefore increasing the cadence value increases the number of program loops between each sent message. Because of the decision to simulate the locality of communication through the help of the IR tower (see chapter 3), the action of sending a message takes much more time in the real robot experiments than in simulation. Thus the variation of the cadence value does not have the same meaning in the real robot experiments. Figure 4.27 plots the real-time periodicity of message sending against

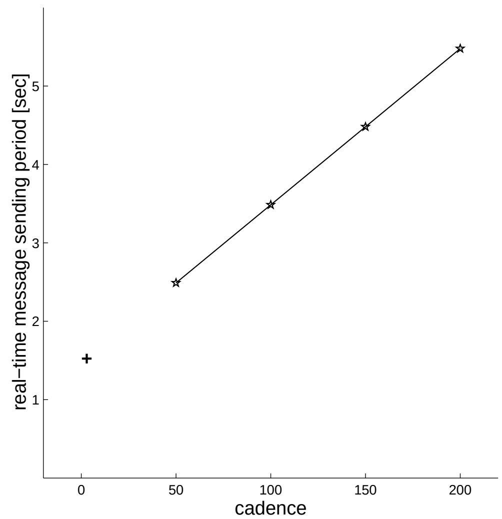


Figure 4.27: time correspondence for cadence values

different values of C (the "+" sign interpolates the cadence value $C = 0$, corresponding to the processing time of the IR tower).

4.3.4 real robot experiments

As mentioned above, the real robot experiments presented a much lower performance than the simulations. The reasons for this drop in performance are investigated in section 4.3.5. The size of the arena and the laboratory resources did not allow experiments with a larger number of robots. Nevertheless the qualitative behaviour seen in the real robot experiments did not differ significantly from that observed in simulation. Figure 4.28 on page 106 shows a frame sequence of two real robots running the basic algorithm.

Firstly we investigate the performance of the β-algorithm with different swarm sizes. The results are depicted in figures 4.29, 4.30 and 4.31. A much lower connectivity than in the simulation with a swarm of 7 robots is observable, even in the presence of 10% noise.

Also the proportion of time spent in different states is very different than in simulation. There is a decrease in the proportion of the "forward" state as swarm size increases, which is not noticeable in the simulation results. It has to be noted that, for larger swarms ($N = 6$ or 7) the robots spend roughly the same time going forward and turning. From the previous discussion of the simulation results, we know that better performance is obtained when robots turn less.

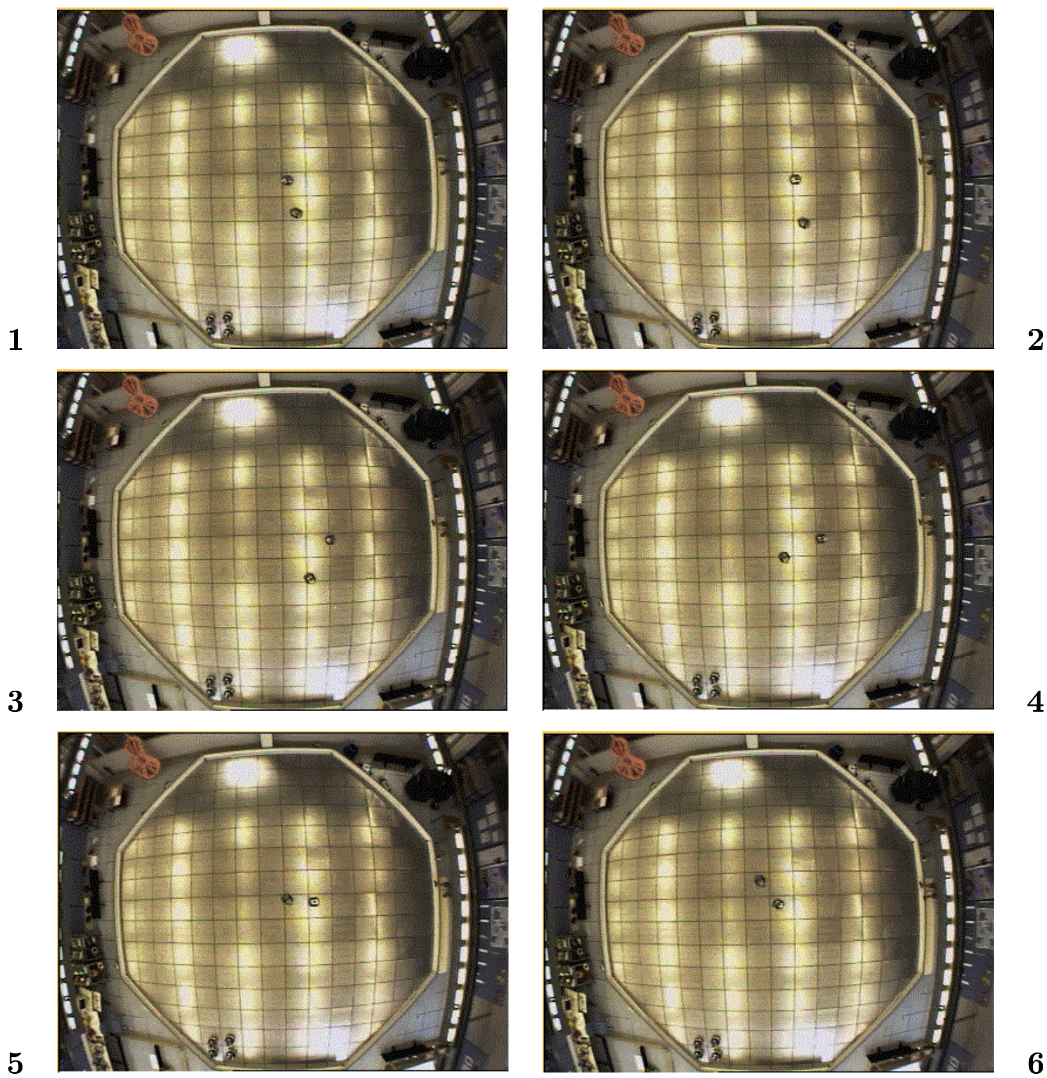


Figure 4.28: real robots running the basic algorithm

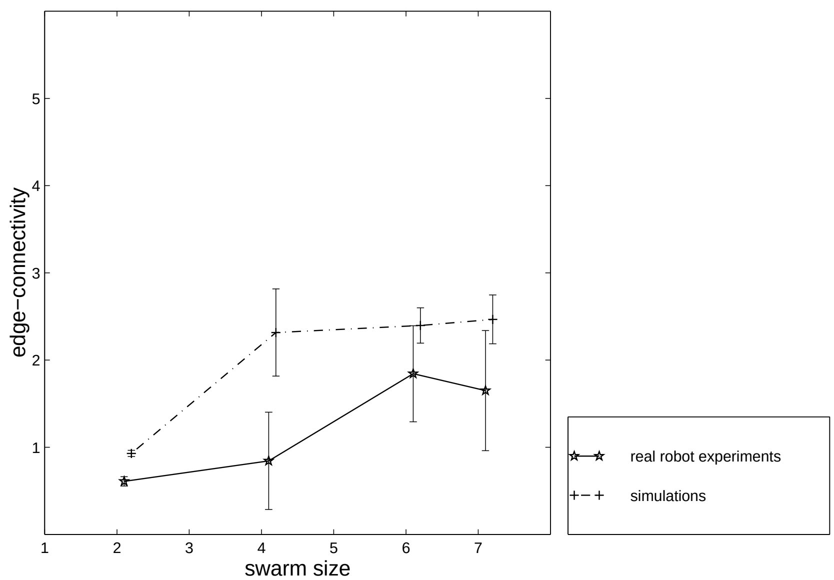


Figure 4.29: connectivity with different swarm sizes

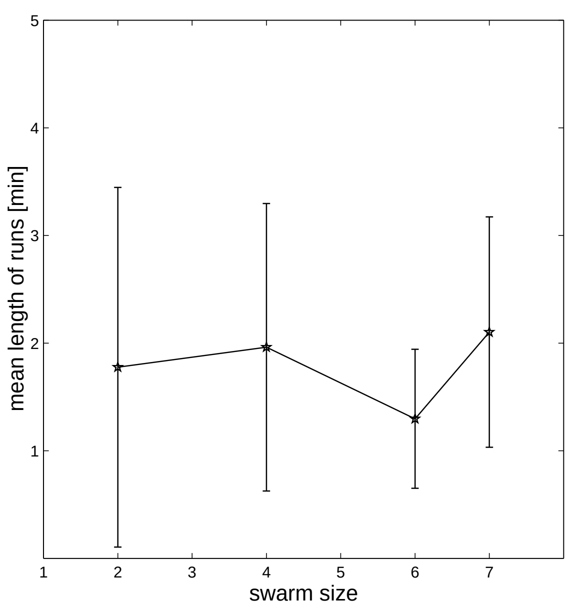


Figure 4.30: run length with different swarm sizes

But it is in the length, and the variability in the length, of the runs that the most striking difference is to be seen. A cadence value of $C = 100$ means a message sending lapse of 3 and a half seconds (see figure 4.27); in the average run (2 minutes) each robot therefore sends an average of only 35 messages. By contrast a successful simulation run meant 1000 messages.

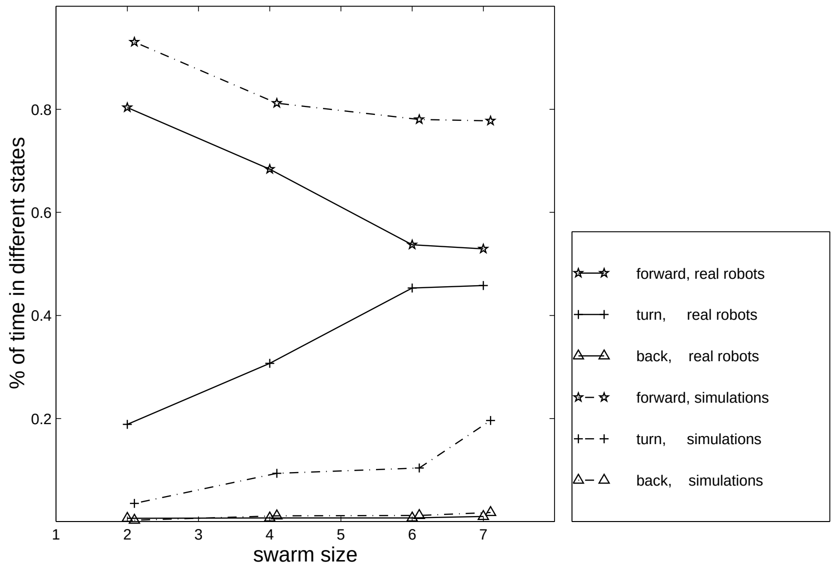


Figure 4.31: odometry with different swarm sizes

When the influence of the β threshold on the behaviour of the swarm is investigated (see figures 4.32 and 4.33), the same features as in the previous discussion, namely very short runs, high variability and differences in odometry are observable. The potential of the β-algorithm to tune the connectivity is nevertheless apparent in figure 4.32. Figure 4.33 shows great correlation of shape; but differences in absolute value are due to the fact that real robots are experiencing a higher level of noise. We know from figure 4.22 (page 100) that increasing noise brings the proportion of "forward" and "turn" states closer together.

Finally the influence of cadence on performance is investigated. The results are depicted in figures 4.34, 4.35 and 4.36 (as there is no exact correspondence with the simulated cadence values, comparisons with simulation results is omitted). Looking at the odometry (figure 4.36), we see that the proportions of the different states vary with different values of C. The proportions for $C = 200$ are quantitatively equivalent to the results of the simulation with $N = 7$ and $\beta = 2$ (see figure 4.19, page 99). A corresponding increase in the connectivity[1] could suggest that this parameter is open for optimisation towards values larger than $C = 200$.

But on the other hand, the mean length of the runs decreases with increasing cadence (figure 4.35). In fact the difference between cadence values in simulation and values in the real robot experiments

[1]the high value for $C = 50$ is due to high spinning of the robots as can be seen in the odometry

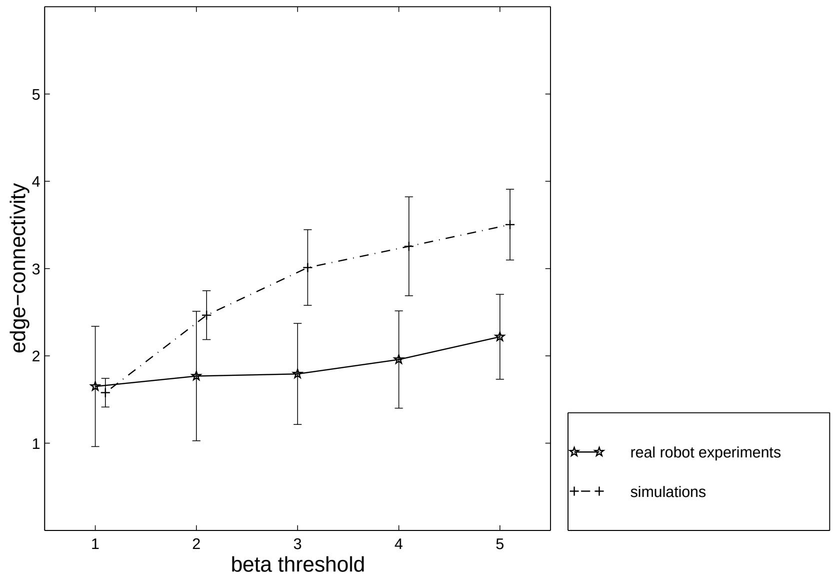


Figure 4.32: connectivity with different β values

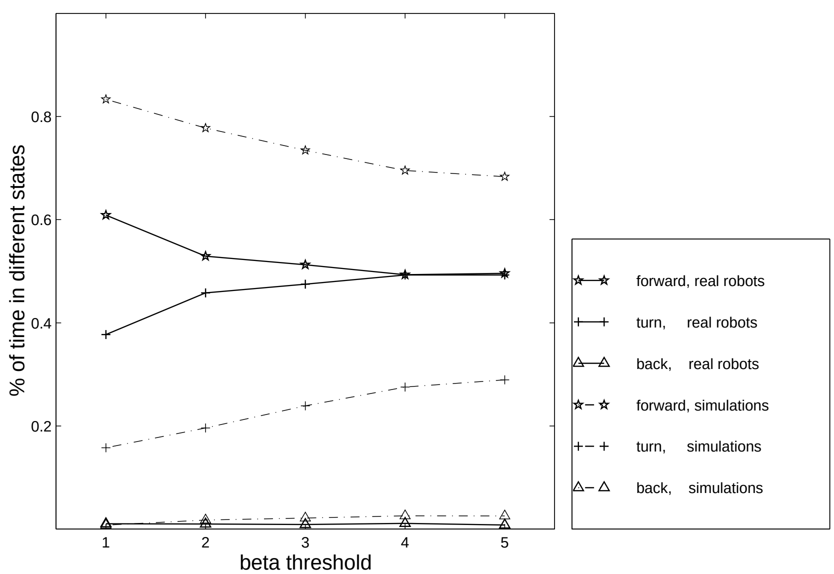


Figure 4.33: odometry with different β values

has an unexpected consequence: as the real robot takes an average of 1.5 seconds to perform a U-turn, the time needed by the IR tower to simulate the locality of the communication leaves no opportunity for the cadence to drop below the turn-time and present the same characteristics seen in the simulation. The pseudo-refractory state that showed better performance in simulation is then out of reach. As this discovery has been made only late in the program of work, new experiments

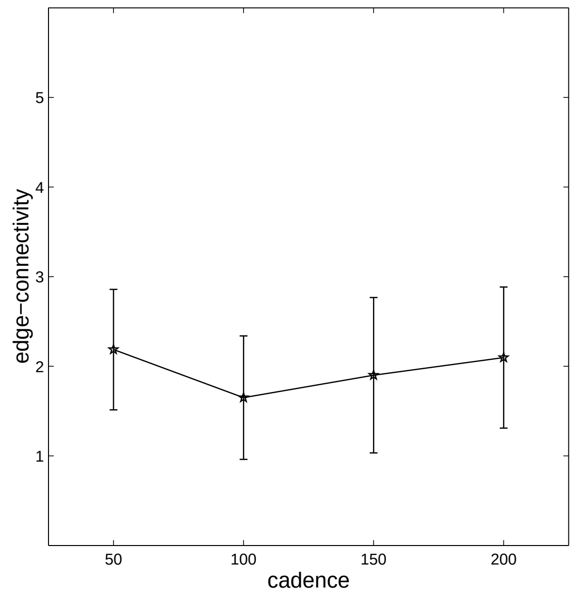


Figure 4.34: connectivity with different cadence values

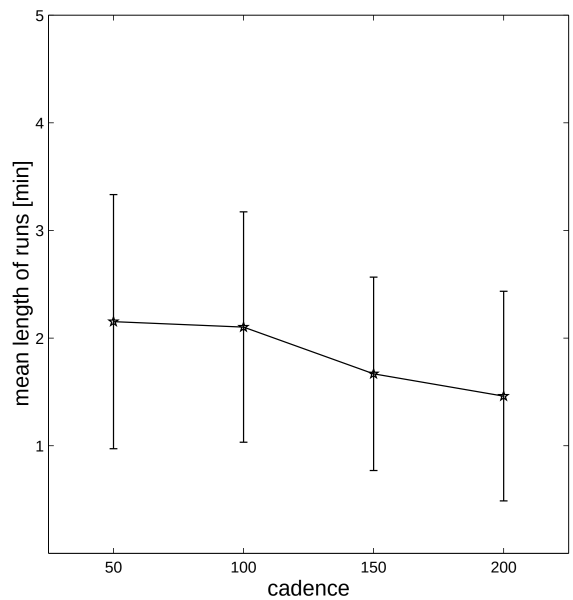


Figure 4.35: run length with different cadence values

could not be scheduled, but it seems very likely that dividing the turning speed of the real robot by 3 or 4 would be beneficial to the swarm coherence.

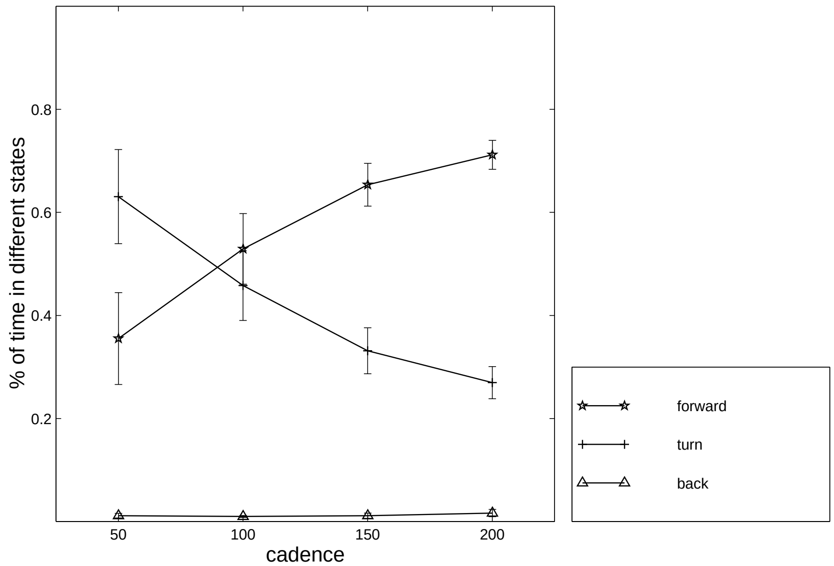


Figure 4.36: odometry with different cadence values

4.3.5 reasons for lower performance

Although clearly showing the desired behaviour, the performance of the real robot experiments is significantly worse than the simulation results. The coherence, that is the stable dynamic connectivity of the robot network, is not maintained for long. The purpose of this section is to question the reasons for this poor performance.

A first reason linked with turning speed has already been mentioned. This concerns the swarm as a whole, but when recording the ID of the lost robot at the end of the run, some robots appeared to get lost more often than others. Figure 4.37 shows the proportion of the runs in which each robot got lost. This proportion has been computed over the 91 runs that involved seven robots (see table 4.26). Note that robot #2 and robot #7 especially show a very low likelihood of getting disconnected (*i.e.* good performance).

An explanation for this variable likelihood of loss is crucial to understanding the low performance of the real robot experiments and the remainder of this section is dedicated to a discussion of the different possible reasons.

A primary cue about where the problem could lie has already been proposed in the discussion of the optimisation of the cadence value. Two successive 180 degrees turns without straight movement has been shown to be crucial for disconnection. But the lack of positive increase in the length of the

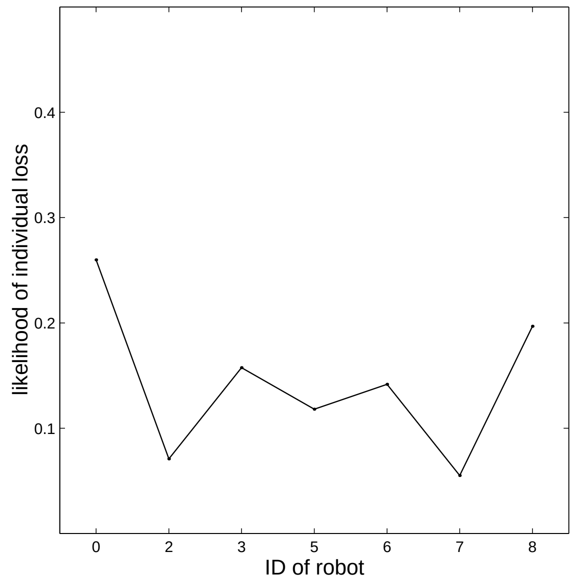


Figure 4.37: likelihood of loss for each robot (note: 7 robots identified 0,2, 3, 5, 6, 7, and 8)

runs with increasing cadence, raises doubts about the unique influence of this factor.

In order to check whether the C parameter has some influence on the distribution of loss across the swarm, the likelihood of loss for each C value is plotted (figure 4.38). $C = 50$ is the only value which presents a somewhat more even distribution. But this value makes the robots turn most of the time until they eventually disconnect, which explains the slightly more even distribution. Although these results are extracted from fewer runs, qualitatively the same distribution is observed.

If the global cadence value is not responsible for differential loss, there might be hardware heterogeneities in the robots that makes them react differently to the same algorithm. Indeed there are and the first one can be found by looking again at the cadence.

a possible reason: heterogeneous cadence

Let us consider the possibility that the heterogeneities are due to the two different generations of Linuxbots within the experimental robot population; one using 486 processor technology and the other 586 (Pentium) technology. Since the cadence value is implemented as a count of algorithm cycles rather than an absolute time value, could variations of processing time be responsible ? Note that although the β-algorithm is very simple in software terms, it hides significant computation in the IR tower subsystem and network communication.

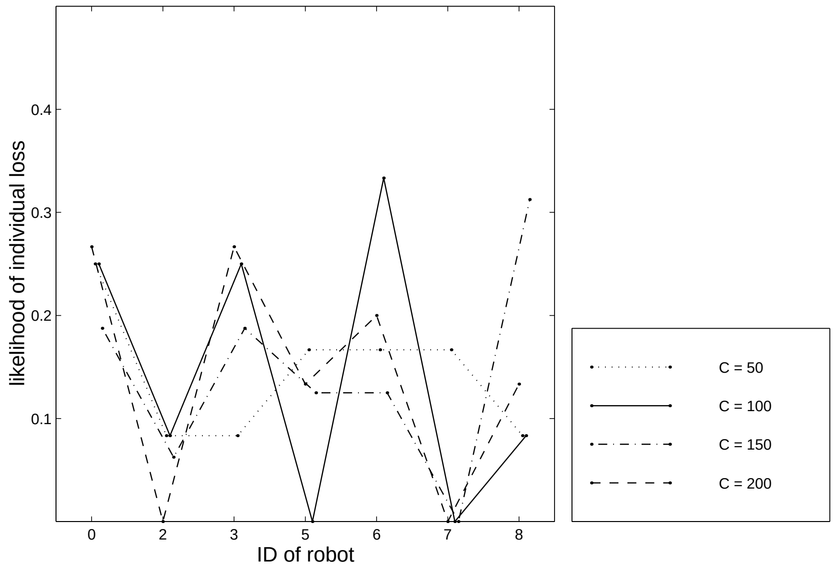


Figure 4.38: proportion of loss for each robot with different C values

Figure 4.39 depicts the comparison of the increase of the time against the number of messages sent between robot #0 and the others for different C values. For robots #3, #5 and #6, the lines don't superimpose. The lines are slightly under the ones for robot #0 for all C values. Robots #3,#5 and #6 belong to the more recent generation and have faster processors. But there is no difference in speed between robot #0 and robot #2 and therefore the difference in likelihood of loss between robot #0 and robot #2 cannot be explained by different platform speeds.

a possible reason: infra-red noise

As differences in processor speed are not sufficient to explain the differences in loss, another possibility for hardware heterogeneities is investigated: the IR tower.

As can be seen in figure 3.10 (page 70), the infra-red tower used to measure the distances between robots in order to simulate the locality of the communication device, is composed of eight IR receivers mounted above eight IR emitting diodes. Each of these has properties that vary around a mean value (see chapter 3, section 3.3.1). In order to compensate for these variations calibration of the IR tower was conducted.

The question is whether the differences in likelihood of robot loss could be explained by the differences in IR towers such as signal-to-noise ratio, etc. Figure 4.40 depicts the "in communication"

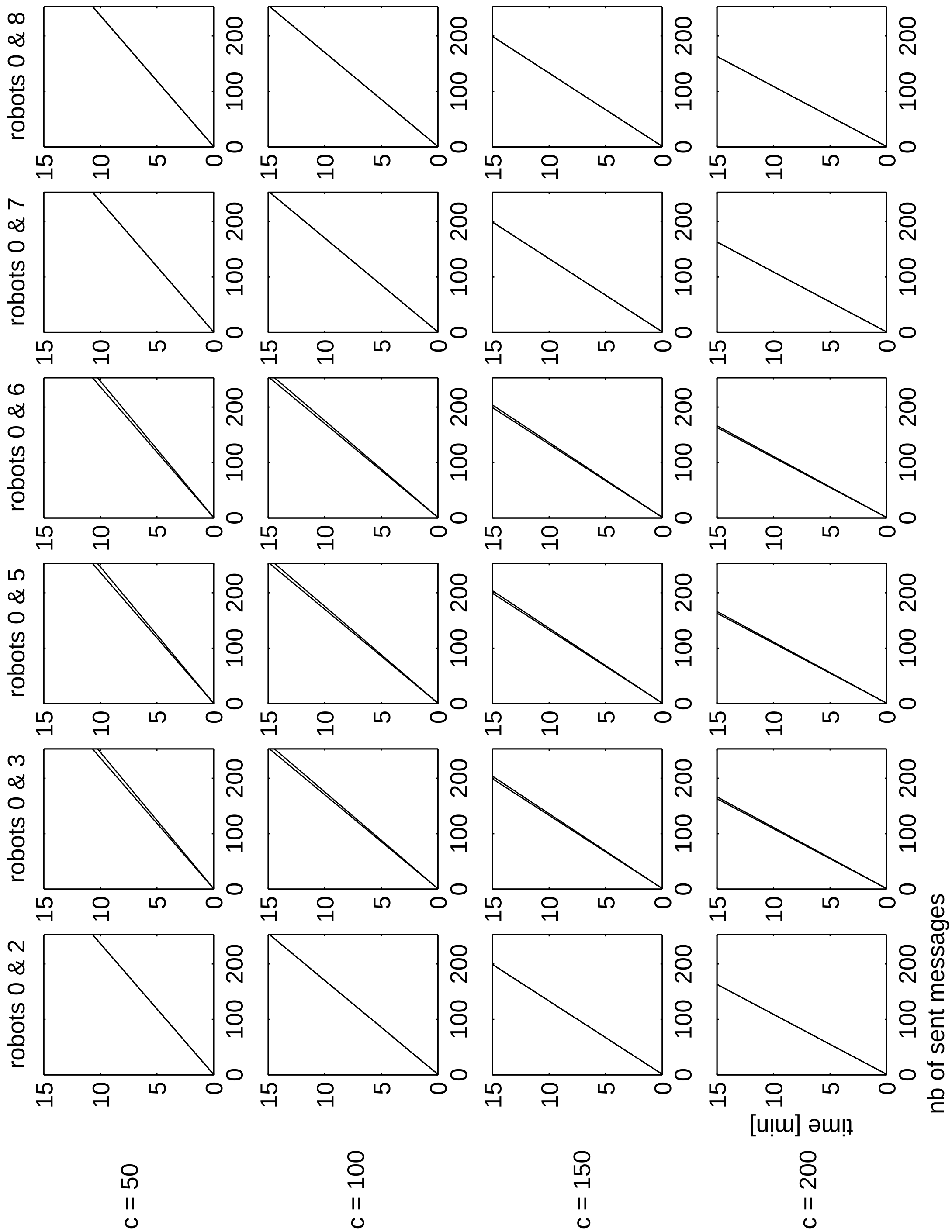


Figure 4.39: robot cadence comparisons

threshold values for each robot against the robot IDs.

In this plot, robot #2 represents a small but noticeable drop in the calibration value compared with the others (see the characteristic drop in each plot of figure 4.40), but this is not the case for robot #7. Alternatively, the signal-to-noise ratio could explain the frequent failure of robot #8, as the threshold values are as a whole lower than for other robots (last plot of figure 4.40) but robot #0, another bad performing robot, does not present the same characteristic.

This discussion suggests that the IR tower cannot be solely responsible for the differential loss of the robots.

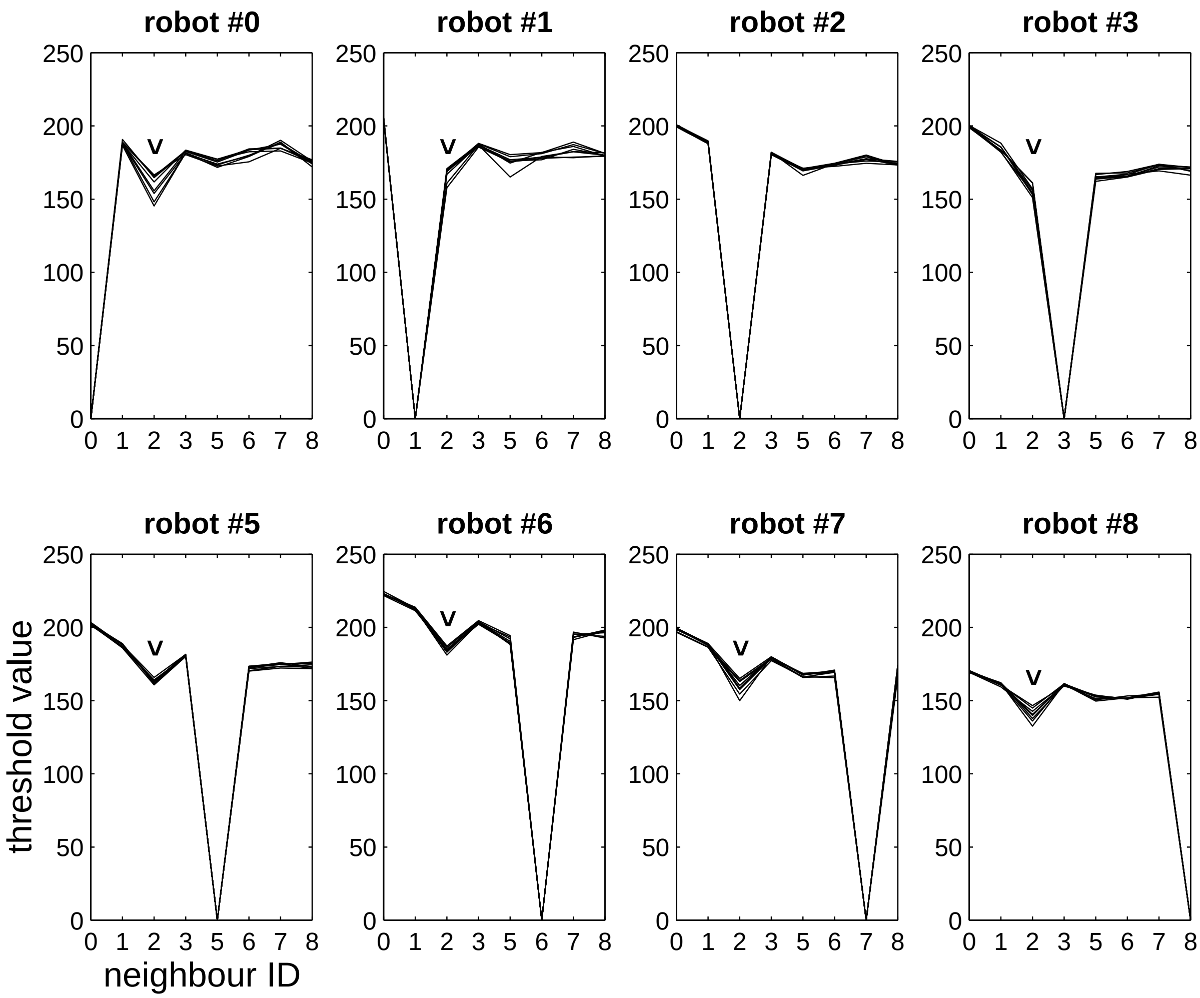


Figure 4.40: calibration against robots ID

a possible reason: avoidance behaviour

The collision sensors used on the robots are IR emitters and receivers and present the same variability inherent to electronic devices as the sensors used on the IR tower. Moreover, all operate on the same frequency. This means that a sensor can mistakenly interpret the IR beam emitted by the sensor of another robot as the reflection of its own beam. The result is a potential interference of the collision sensors at a range of two times the avoidance range, as illustrated in figure 4.41.

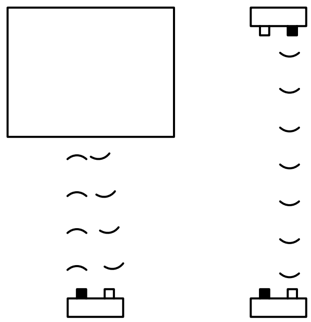


Figure 4.41: avoidance sensors interference

Also, because of the size of the experimental arena and the fact that the β-algorithm needs a large number of robots, the communication range had to be constrained to 1.2 meters to avoid interference from the arena walls. As a result the ratio between avoidance and communication ranges is not the same as that used in simulation: $r = 5$ in simulation and $r = 2.5$ for real robots. This restricted communication range turned out to be actually below the interference range, which sometimes made a returning robot veer away from the swarm before being able to reconnect.

When investigating possible heterogeneities within the swarm, it turned out that the collision sensors have very variable ranges due to possible misplacement of the receiver within the sensor enclosure; a misplaced sensor leading to reduced IR received signal strength, and thus to a reduced sensor range. Again, because these heterogeneities were spotted after the experiments, it is difficult to assert with certainty that the sensors haven't changed platform between the experiments and when these heterogeneities were investigated. Nevertheless, robot #2 and to a lesser extent robot #7 presented sensors with misplaced receivers compared to the rest of the swarm, which makes them more prone to stay within the swarm than the others.

summary of the possible reasons

Because of the delay between the experimental work and the discovery of the role played by heterogeneities within the swarm on the performance, the precise cause of the differential loss cannot be determined with certainty. But the role played by collision sensor interference and heterogeneity seems to be the most likely explanation. It is also possible that a combination of the potential issues discussed here has the power to explain this differential likelihood of loss. Because one of these issues results from an implementation choice, namely the simulation of locality through the IR tower (a choice that would ideally not be repeated), then deeper analysis would not shed greater light on the β-algorithm.

But, the IR tower limitations excepted, this investigation on the potential causes of differential loss has shown several directions for the improvement of the robots: reducing the forward speed of the robots; controlling the cadence more tightly; increasing the size of the arena or, conversely, reducing the range of the avoidance sensors. Also this analysis suggests that the β-algorithm is able to cope with heterogeneities in cadencing, which was not expected.

4.4 Spatial Covering

As one of the underlying motivations for this study is to build distributed sensor networks it is interesting to study the range of behaviours that the implementation of the β-algorithm is able to produce without the aid of any new sensing or processing abilities. The following sections contain a description of work on spatial covering and localization, involving only strictly homogeneous swarms. Again, investigations are conducted firstly in simulation and only the spatial covering algorithm is tested on real robots.

It is of course of primary importance for sensor networks to be able to control the area coverage of sensing components. This corresponds to the control of the granularity of the sensing. The idea is to test the precise role played by the threshold coefficient β and how the area coverage can be controlled through it. Figure 4.42 shows that differing β values indeed have an influence on the swarm coverage.

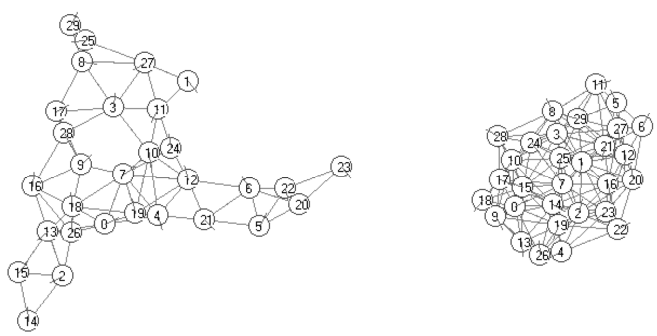


Figure 4.42: examples of area coverage with $\beta = 1$ and $\beta = 4$

Figure 4.43 shows the area covered by a swarm of 40 robots when the β threshold follows a transition from $\beta = 1$ to $\beta = 20$ and vice-versa. We observe that the threshold value can indeed be used to control the area coverage. It is also interesting to note that the spreading is much faster than the aggregation: when β is raised the area coverage slowly reduces to reach an equilibrium in around 100,000 time steps, whereas the swarm only needs 50,000 steps to return to the initial spread. Figure 4.44 depicts the corresponding edge-connectivity, primarily increasing towards a plateau and then swiftly reducing.

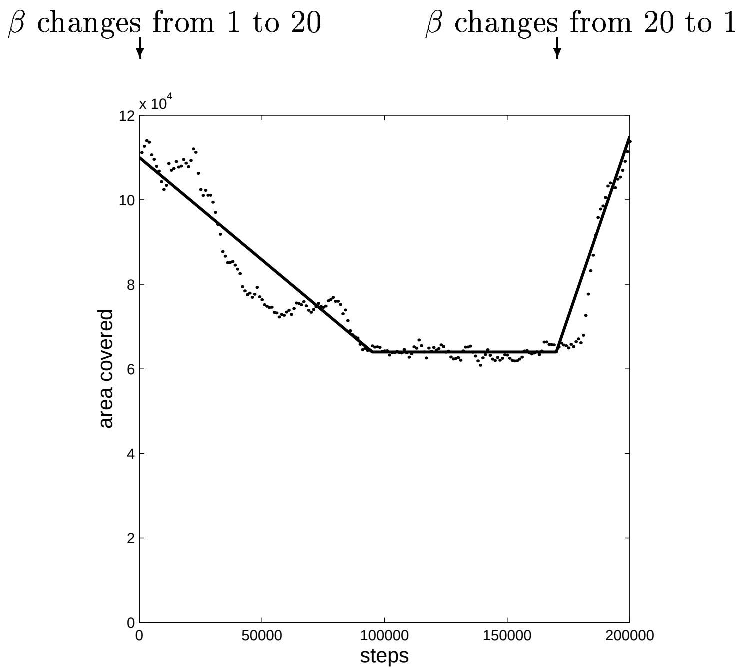


Figure 4.43: area transition from $\beta = 1$ to $\beta = 20$

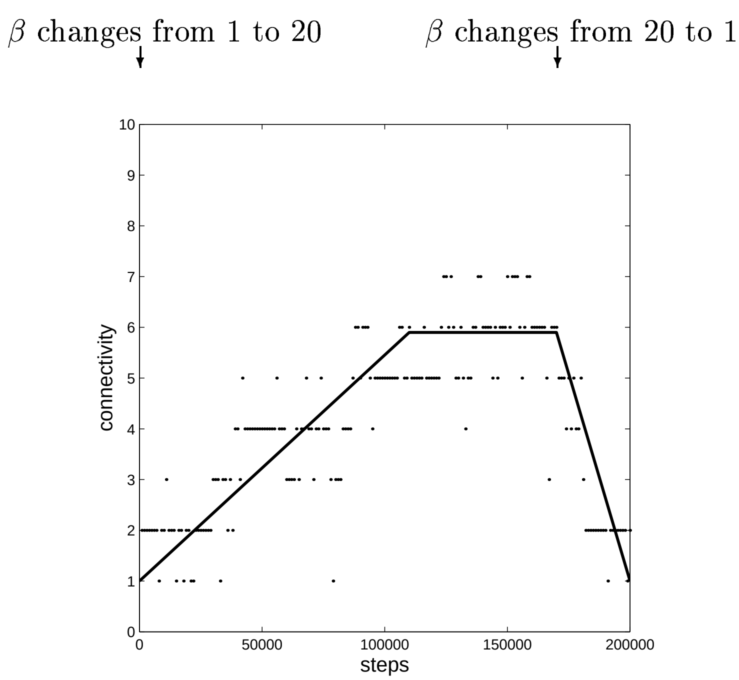


Figure 4.44: connectivity from $\beta = 1$ to $\beta = 20$

4.4.1 spatial measures

The area covered by the robot swarm is defined as all points on the plane that are within the communication radius of at least one robot. We measure this area using an approximation by bisection (see chapter 3). In the simulated case, position information was readily available, whereas positions for the real robot experiments have been assessed by visually following every robot, frame by frame, on the video recordings of each experiment. This means of getting the positions of the robot is prone to errors but an automatic video tracking system was not available at the time of the experimental work.

The measures to record area coverage are as follows:

- *area coverage* and *area coverage per robot*, which are the measure of the area spanned by the communication radius of all robots and the same value divided by the swarm size.

The length of the runs in simulation and in real robot experiments are the same as in sections 4.3.1 and 4.3.3 (pages 95 and 103).

4.4.2 simulation results

When the mean area covered by the robots during the run is plotted against β threshold for different swarm sizes, this gives the family of curves of figure 4.45. In this figure the potential of the β threshold to control the area is clear, and control is effective up to a value of $\beta = 10$. Beyond this value the contraction of the swarm takes longer than the length of the run, and is therefore not measured. But in any case, the contraction is limited firstly by the avoidance behaviour and secondly by the size of the robot body. What is also of primary importance is the linear increase of the area when increasing the size of the swarm. This increase can also be modulated by the β threshold, unharming to the linearity. This shows that there is no leveling of the action of the algorithm and we can therefore potentially control the area of any swarm size.

The area divided by the number of the robots gives the mean contribution of a single robot to the area (figure 4.46). For all swarm sizes, the slope of the decrease of the contribution is strikingly similar. This proves that the amplitude of the β-control is more or less the same with any number of robots. Of course the effect on the whole swarm area is bigger with larger swarms (figure 4.45). The decrease of the contribution for increasing swarm sizes is due to overlapping communication areas: for larger swarms a greater proportion of robots are situated within the swarm. Because of overlaps, these robots don't contribute as much as robots at the boundary to the whole swarm area. As expected, this decrease levels out with increasing swarm size.

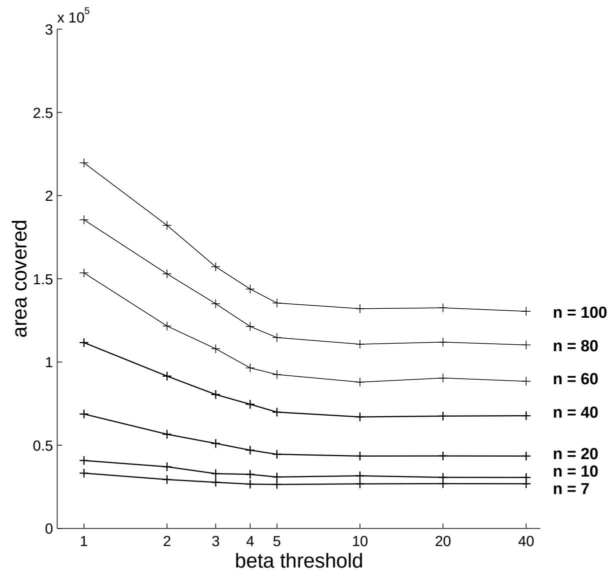


Figure 4.45: area covered against β and swarm size

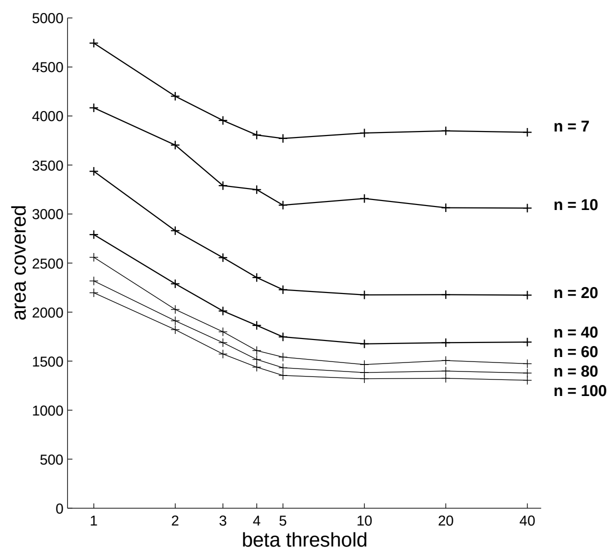


Figure 4.46: normalised area against β and swarm size

Now we investigate the influence of noise on area coverage and the results can be seen in figure 4.47. Again these results have been obtained with increasing levels of noise in connection, proximity sensors and actuators simultaneously. The plot of the normalised area shows that it stays quite constant for $\beta = 2$ and increases slightly for $\beta = 5$. The increase corresponds to a drop in the connectivity and a high proportion of "turn" state. This suggests that an increase in noise proportion could lead to a more brittle swarm. Nevertheless the fact that the area shows such a small variation with a large

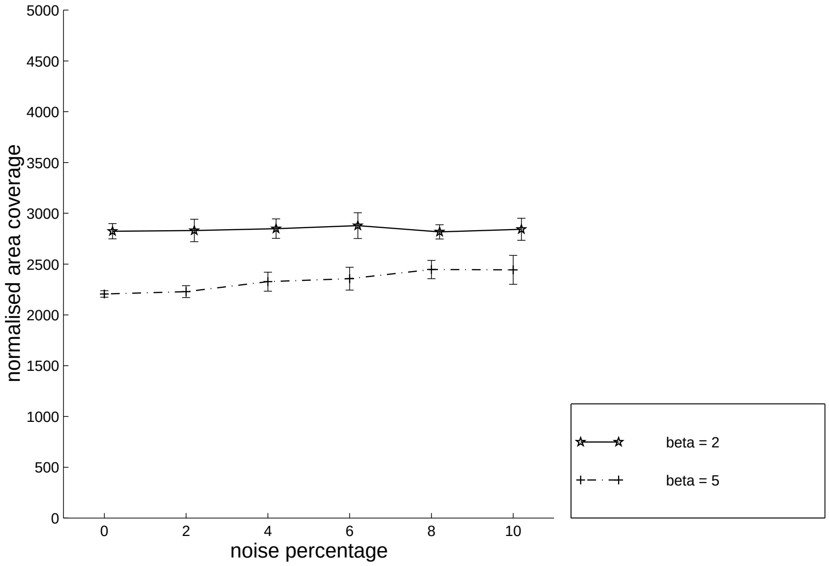


Figure 4.47: normalised area with increasing noise

increase in noise is a striking feature of the algorithm.

Finally the spread of the swarm against an increase in the cadence of sending messages is studied. The results are depicted in the figure 4.48. A very slight increase in the area is noticeable for $C = 200$ and corresponds again to a drop in connectivity (see figure 4.23, page 101).

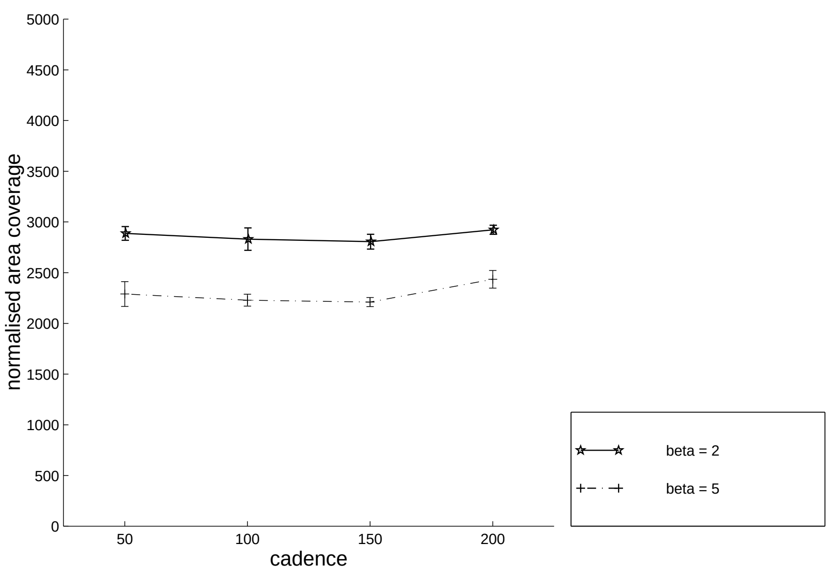


Figure 4.48: normalised area coverage with increasing cadence

4.4.3 real robots experiments

The results of real robot experiments on area coverage are depicted in the figures 4.49 to 4.51. Note that the correspondence in area values between real robot experiments and simulations is computed through a geometrical transformation that does not consider lens distortion. The comparisons in the figures are thus more qualitative than quantitative. An increase in swarm size shows a decreasing curve for the normalised area, similar to the simulation results (figure 4.49). This behaviour is also due to overlaps, as described in section 4.4.2.

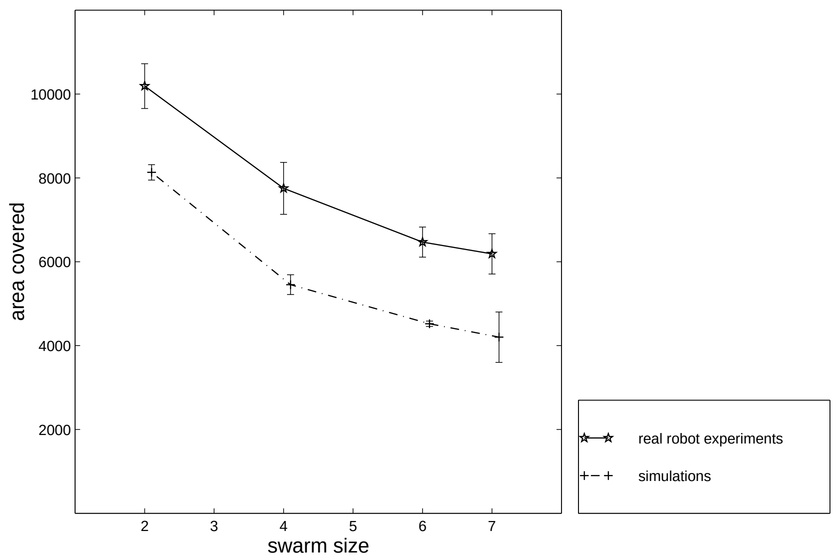


Figure 4.49: normalised area with increasing swarm size ($\beta = 2$)

When parameter β is varied a decrease in the normalised area coverage is noticeable (figure 4.50), although the range of this decrease is not as large as that seen in simulation. The reason could be the lack of connectivity. Indeed such values in connectivity are in simulation translated into larger area coverage. Considering cadence, area coverage shows no sensitivity to an increase but a higher variability (figure 4.51). Figure 4.52 shows typical dispositions of real robots running the β-algorithm with differing β values.

4.4.4 area coverage estimators

Chapter 3 (and Appendices A and B) introduced a method to estimate, from the adjacency matrix of the network, an upper and lower bound for the area coverage of the swarm. Figures 4.53 to 4.57

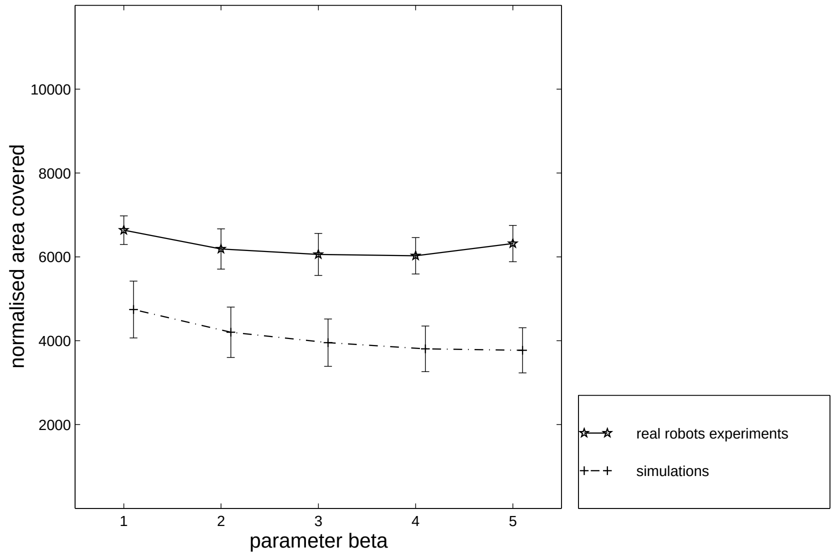


Figure 4.50: normalised area with increasing β threshold $(n = 7)$

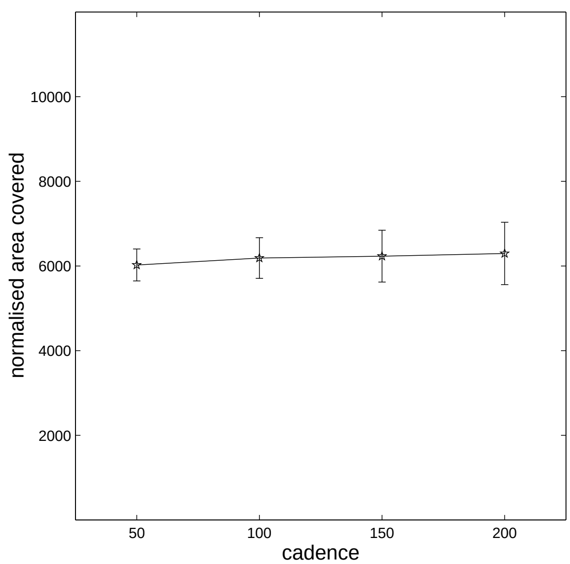


Figure 4.51: normalised area with increasing cadence $(\beta = 2, n = 7)$

present the results of a set of experiments testing this hypothesis. Runs lasted 100,000 steps and 5 runs were needed to compute the mean and standard deviations.

It results from figure 4.53 that the estimates are upper and lower bounds for the area coverage. However, when the normalised values are considered (figure 4.54), we observe that these bounds fail to display the same behaviour as the actual area coverage. More specifically, the decrease in normalised

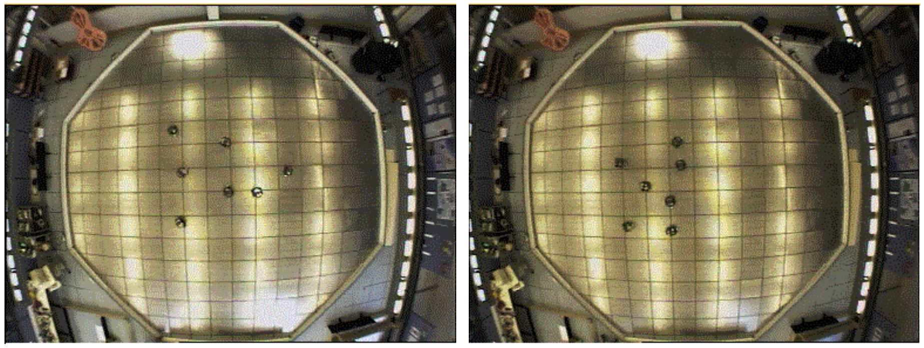


Figure 4.52: real robot experiments with $\beta = 2$ and $\beta = 6$

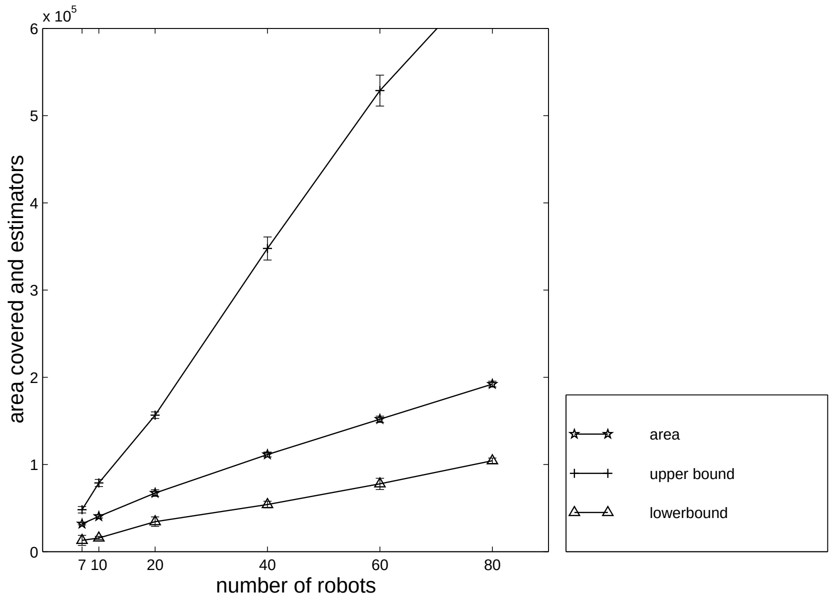


Figure 4.53: area and estimators with increasing swarm size

coverage that occurs with increasing swarm size is very slightly observed in the behaviour of the lower bound, while the upper bound shows a completely opposite behaviour. Also the contraction of the swarm with increasing β values is not observed (see figure 4.55).

The reasons for such inaccuracies are to be found in the fact that the basic assumption for the estimation of the upper bound is placing each robot on a circle in order to measure its contribution to the overall area. This results in every robot contributing something to the area (with the small outermost region, see Appendix A). On the other hand, in a simulated swarm, there are robots situated within the swarm whose area is completely covered by others. And the number of such

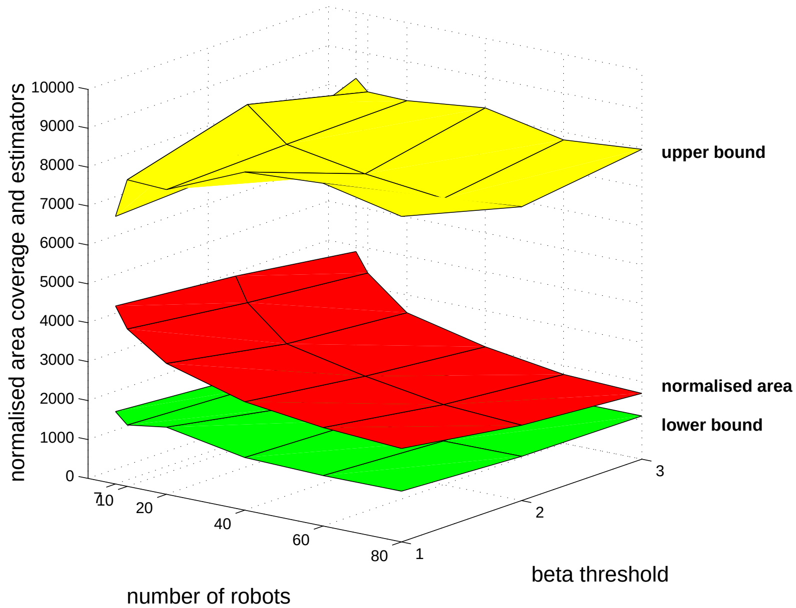


Figure 4.54: normalised area and estimators with increasing size and β values

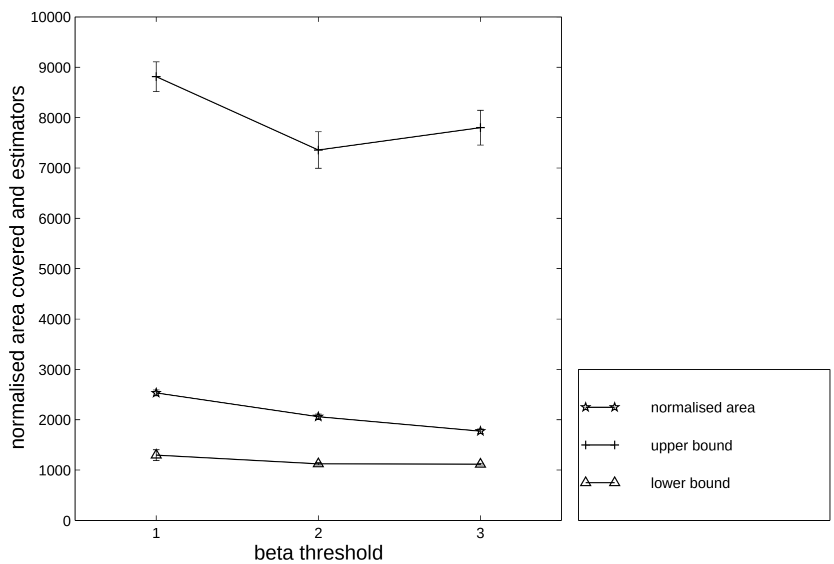


Figure 4.55: normalised area and estimators with increasing β values

robots increases with increasing size.

The example presented in figure 4.56 shows that although being far above and under the area coverage value when averaged over whole runs, the bounds estimators are not strict lower and upper bounds. Indeed the area coverage value occasionally exceeds the upper bound value or finds itself below the lower bound. As non-strict bounds typically are difficult to narrow (usually loosing bounding ability), this feature raises doubts of a possible improvement in the bounds without involving more than just consideration of the vertex degree value of the graph.

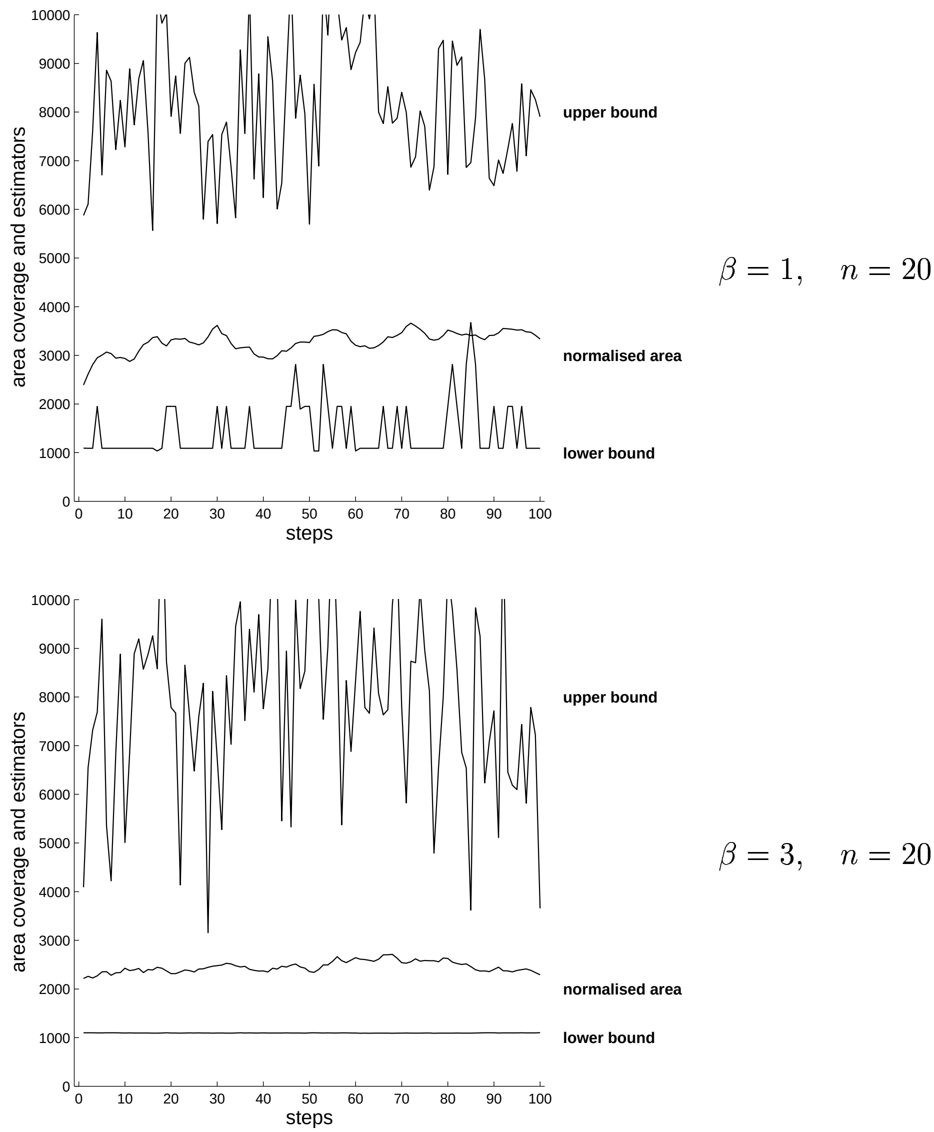


Figure 4.56: normalised area and estimators during a run with differing β values

Figure 4.57 shows the influence of noise on the estimation of the area. The direct influence of noise on the connections does not seem to have a substantial influence on the area estimators. This corresponds well with the behaviour of the area coverage in the presence of noise.

Despite these inaccuracies, the area coverage estimators still present a crude idea of the spread of the swarm and can be of use in different contexts such as, for instance, distributed sensing. It is worth noting here that the estimators proposed did not make any use of neighbour information,

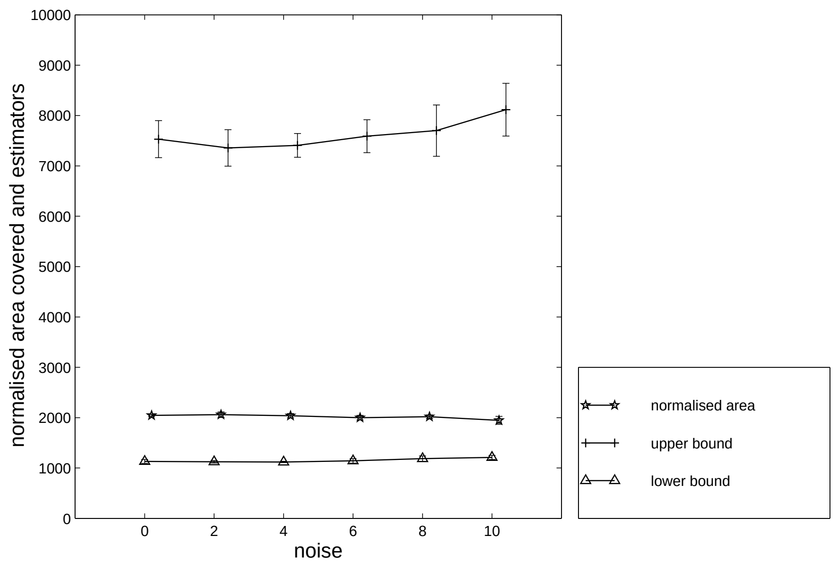


Figure 4.57: normalised area and estimators with increasing noise

which indicates room for improvement within the framework of the β-algorithm.

4.4.5 summary

In this section, the power of the β-algorithm to achieve control of swarm spread has been demonstrated in simulation and this ability is not lost with increasing noise. The relative difficulty of the real robot swarm in demonstrating the same behaviour is linked with practical issues in achieving coherence and not due to some physical impossibility.

Also, a method has been presented that is a means for deducing upper and lower bounds for area coverage from the adjacency matrix of the network, which performs well despite high frequency time fluctuations. This method represent a first step towards the characterisation of an efficient robot swarm through analysis of the adjacency matrix of the network. Such a characterisation would be needed for the assessment of a probabilistic model describing the dynamics of the robot swarm by modeling changes in the network (see chapters 3 and 7).

In the next section a means will be presented for the robot to locally deduce its relative position within the swarm.

4.5 Spatial Selection

One of the initial aims of this research is to achieve coherence without the help of relative positions. The previous sections of this chapter have shown that this is indeed possible. Now it will be demonstrated that the increase of information exchange introduced by the β-algorithm brings a straightforward way to infer a very crude approximation of the relative position of each robot within the swarm.

Nagpal [Nagpal, 1999] describes an interesting solution within the framework of amorphous computing, using triangulation on hop counts to derive an estimate of the position. But this algorithm makes use of a spread of information that will use some of the available bandwidth. Moreover the problem of choosing the three robots needed to initiate the spread is not addressed.

The idea developed here instead makes use of the information already available to the robot by the requirements of the β-algorithm: its own degree of connections and those of its neighbours.

4.5.1 description

At the end of its cadence round the robot compares its connectivity degree to the ones of its neighbours. If its own degree is the greatest it assumes it is located within the interior of the swarm, and if it is the smallest, the robot assumes it must be at the border of swarm. This process is equivalent to finding the local extrema of the connections degree potential over the swarm.

More formally, a robot R considers itself an *insider* if

$$deg_R \geq deg_{R_i} \forall R_i \in Neigh(R)$$

and an *outsider* if

$$deg_R \leq deg_{R_i} \forall R_i \in Neigh(R)$$

and neither if these two conditions are not satisfied.

The justification for this method is simply that the degree K tends to be lower at the border. The reverse argument holds inside the swarm. Of course this approach cannot guarantee to select all of the robots that are actually at the border of the swarm, but that is the price to pay for locality and low bandwidth exchange.

Another drawback is the possibility that a robot believes itself to be at the border although it is not, when considering large groups of robots where local minima can happen within the swarm. But these situations are not stable and depend on low values of β.

The advantage is that this selection is done completely locally and that it is able to cope with any degree of dynamicity as a change in the network is immediately reflected in the selection. A biological example of such a dynamic selection can be found in cyanobacteria, like in the genus *Nostoc*, which forms filaments where 5 to 10% of the individual cells differentiate into heterocysts to produce colony essential nutriments. If this individual is removed from the filament, reorganisation occurs and a new individual differentiates. To achieve this differentiation, intercellular communication is essential but it remains unclear what precise mechanism is at play [Adams, 1997, Shapiro, 1988].

4.5.2 measures

The measures to record performance of the spatial selection algorithm as follows:

- *insider and outsider mean distance to the center of mass* and its *standard deviation*, for both groups of selected robots, the distance and its standard deviation is computed, in order to measure the difference between them.
- *whole swarm mean distance to the center of mass* as defined in chapter 3 for comparison purposes.
- *proportion of insider/outsider robots* in the swarm, to measure the ability of the algorithm to select robots.

The length of the runs in simulation are the same as in section 4.3, namely 100,000 steps.

4.5.3 simulation results

The results of simulation runs are depicted in figures 4.58 to 4.63. Firstly we investigate the influence of swarm size and β value. In figure 4.58, there is a clear separation between the average mean distance to the center of mass of "insiders" robots (surface below), "outsider" robots (surface above) and the swarm's global mean distance to the center of mass. This clearly confirms the ability of the proposed algorithm to spatially select robots.

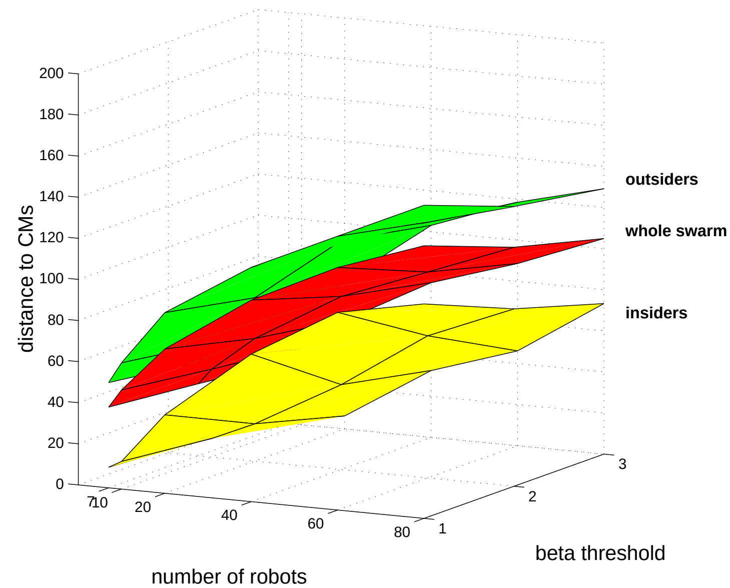


Figure 4.58: distance to center of mass

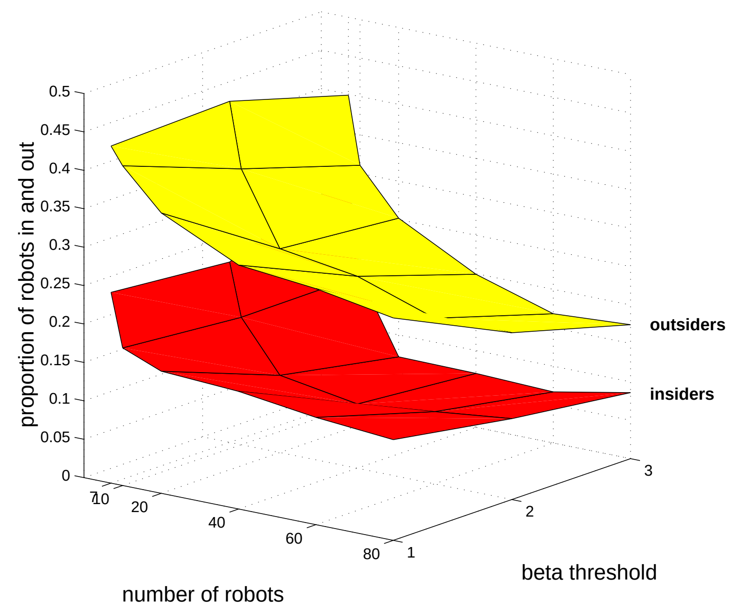


Figure 4.59: proportion of selected robots

When considering the proportion of both groups of robots in the swarm (figure 4.59 “outsiders” surface above, “insiders” surface below) “insiders” happen to be less frequent. The explanation lies in the fact that the number of robots at the boundary of the swarm is, in “healthy” swarms, larger than at the local connection degree maxima. Which means that more robots are potential candidates

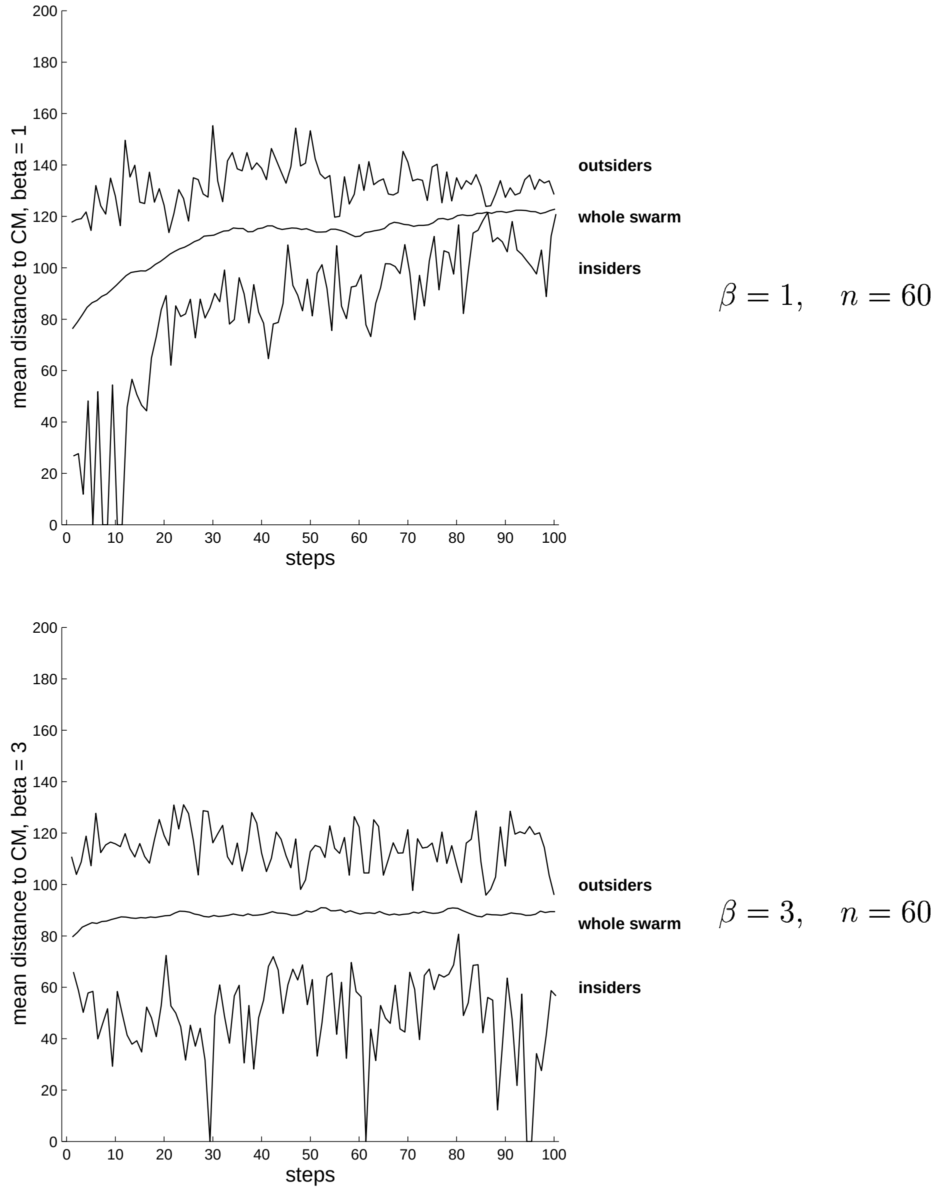
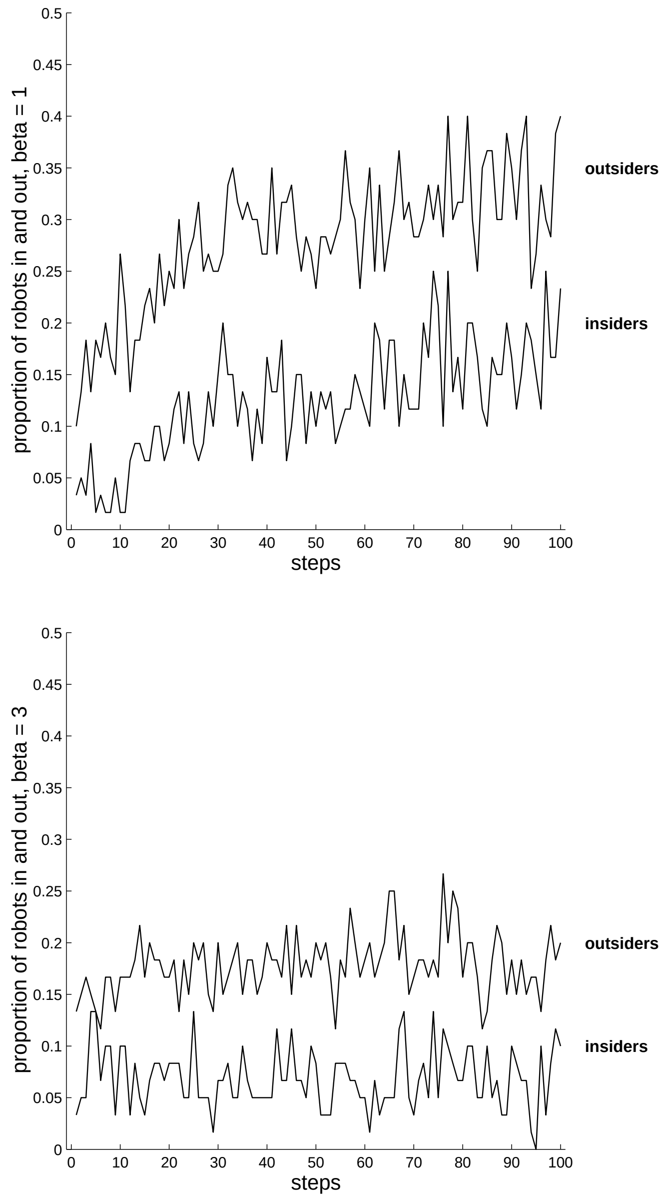


Figure 4.60: distance to center of mass during a run

for becoming "outsiders".

When observing the mean distances and proportions for a larger swarm ($n = 60$) with different β threshold values ($\beta = 1$ and 3) during a run (figures 4.60 and 4.61), it turns out that a smaller β value lowers the performance of the selection. Indeed the difference between distances to center of mass is reduced. This can be explained by considering the potential formed by all connection degree values on all robots. With small β values, the dynamicity of the swarm is high and this potential is typically very hilly, resulting in many different local maxima and minima in large swarms, as can



$\beta = 1, \quad n = 60$

$\beta = 3, \quad n = 60$

Figure 4.61: proportion of selected robots during a run

be seen in figure 4.61. Typically the number of local minima and maxima increases as the swarm expands thanks to its low β threshold. As a result the probability for these local extrema to be located outside or inside the swarm is lowered. Increasing the β value leads to a more even potential lowering the number of local minima and increasing selection performance.

Studying the influence of noise on the spatial selection algorithm (figures 4.62 and 4.63) shows that the proposed approach is globally resilient to noise, which is of remarkable interest considering that connection noise has a direct impact on the degree values of the swarm.

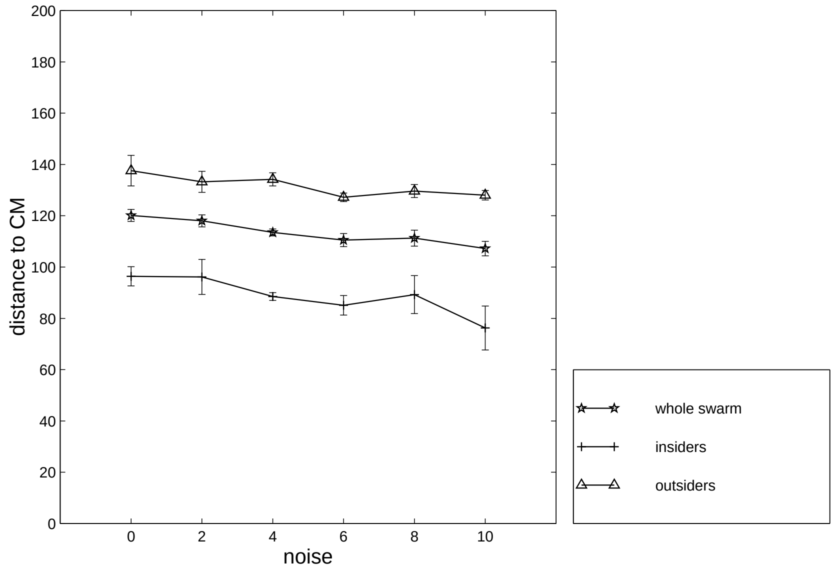


Figure 4.62: distance to center of mass with increasing noise

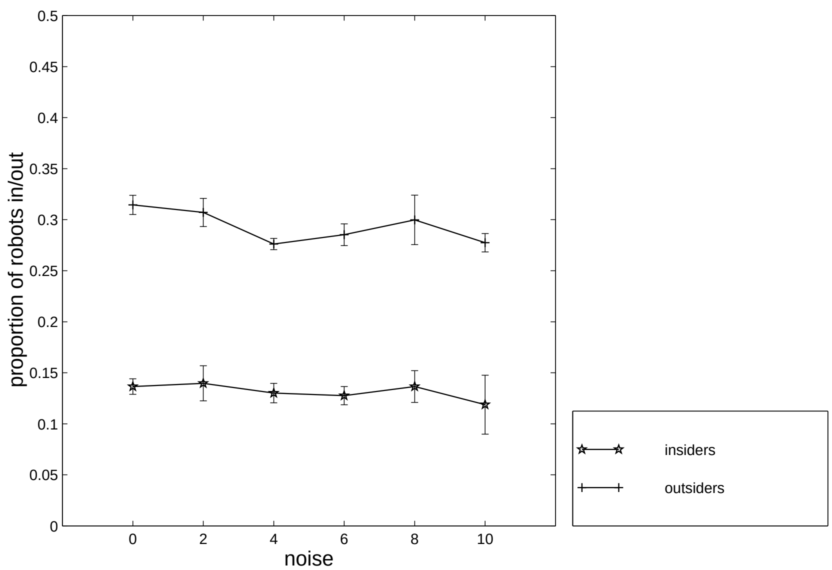


Figure 4.63: proportion of selected robots with increasing noise

4.5.4 summary

This section has presented a method for an individual robot to declare itself near the center or at the boundary of the swarm, with the help of information already available, because of the requirements of the β-algorithm. It has been shown that despite short-term inaccuracies at the individual level, this algorithm is actually very efficient when whole swarms are considered over the run. It is also very resilient to noise.

Indeed, while the information provided by this technique is not as precise as relative or global coordinates, it is nevertheless a reliable information the robots can act upon to trigger specific behaviours necessary to the task being implemented.

4.6 Summary of the Chapter

discussion of α and β variants

Although coherence is greatly improved in introducing the shared neighbour variant the β-algorithm does have some drawbacks.

The main one is the increase of radio-communication bandwidth, as each robot needs to transmit its neighbours' list to its immediate neighbours. This indeed increases the communication overhead; but as the communication is assumed to be local and as the neighbour information is only needed by the robot's neighbours, this isn't a real problem when dealing with large groups of robots. In fact, the β-algorithm displays an almost absolute scalability: its complete locality and distributedness put no limit in the number of the robots involved. There is a slightly more subtle implication, which is the new importance of the identities of the robots. When two robots exchange their list, it is of critical importance that they agree on each others' IDs. One solution is to give every robot a unique identification number. This might have no implication for our experiments as the number of robots will be constrained but it will certainly be a limiting factor if we were to control a very large number of robots; but the example of unique hardware MAC addresses across the internet (around 25 millions of different IDs) shows the potential of the algorithm as it stands.

A solution could be found by dynamically allocating the robots' IDs. A very simple approach consists of a previously unnamed robot in choosing its ID as different from the list of IDs received in the first message. Then comes the problem of changing the ID of the twins that eventually come into contact. This can be done deterministically by setting the new ID as the sum of all IDs received in the last message (modulo some limiting value), through the hypothesis that the neighbourhood of each twin is different. Of course the number of shared neighbours will be erroneously computed until the twins directly send themselves a message but this will affect coherence only slightly.

coherence

The β-algorithm does not make use of relative positions within the swarm. The spatial information needed for the robot to react is embodied by the presence of a message or not. Instead of concentrating on maintaining distances as would be done for a formation algorithm, the idea is to concentrate on the connectivity of the immediate neighbourhood. Global connectivity comes only as a side effect, although an extremely valuable side-effect.

It has been shown in this chapter that the parameter β is able to control the edge and vertex-connectivity of the swarm network. This is of primary importance: these measures are global measures, that is to say they hold for the whole network; if there is on one end of the network a robot with a unique connection to the rest of the swarm then both measures will take the value of 1, disregarding the connectivity of the remaining robots. On the other hand, the β threshold is a strictly local parameter. Therefore its capacity to maintain and tune the global connectivity is of remarkable interest.

area coverage

Summing up the results for area coverage, we can say that the desired control of the swarm is possible by tuning the parameter β. It has to be noted that the amplitude of the control is determined by the ratio between the communication range and the avoidance range/robot body size. With a higher communication range a larger coverage can of course be met with a low β value, while the limit for the smallest area will still be set by the avoidance range and the robot body size.

Also, the coverage is not greatly affected by an increase in noise, which is of primary importance if the coverage has to be guaranteed in noisy environments.

A way to deduce the coverage knowing the distribution of connection degrees has also been investigated and showed interesting results, but the upper and lower bounds created fail to really exhibit the same behavioural features as the actual coverage, especially with the influence of the threshold β.

spatial selection

The β-algorithm provides enough information exchange for a selection of distinctive robots. This selection can be of great interest in the context of distributed sensing: indeed the robots that select themselves as being within the boundary of the swarm are preferable candidates for the garnering of information for they have, by definition, a high degree of connections and should be situated near the center of mass of the swarm. The outside robots on the other hand can be used for the sensing itself for they should present a nicely spread group. Alternatively, the outsiders can be used to communicate the sensed data with some neighbouring external agent.

β-algorithm is similar to differential adhesion

The crucial role played by differential adhesion for morphogenesis has already been mentioned in section 2.12.3. In fact the β threshold can be regarded as a "glue" parameter for the robot, a high value meaning a very sticky robot. Indeed by raising the threshold, the robot will react to more connections and hence will stick more to the others. With a low value, it will move more freely within the swarm. This analogy will be further investigated in chapter 6, where it will be shown that an extended algorithm can demonstrate somewhat typical behaviours of cells during morphogenesis.

β-algorithm is dimension independent

It is interesting to note that the β-algorithm has been straightforwardly transposed to a three dimensional simulation (result not included in this thesis). The ability to tune the spread of the swarm and the possibility to spatially select the robots was clearly observed. These behaviours can therefore be seen as direct consequences of the basic design and not a particularity of a precise dimension. In fact the behaviours presented in this chapter are defined inside the environment defined by the connected swarm and are therefore independent of the dimension of the environment the swarm is embedded within.

noise in simulation and in real robot experiments

In the simulation noise has been introduced as false sensors readings, variability in the actuators or complete message loss in the case of communication. According to this choice, levels of noise were increased from 0 to 10% uniformly. This is by no mean realistic, as there is no reason that the amount of noise at the actuator level should be proportional to the noise at the communication level. In fact, 10% of message loss represents a very poor communication channel by current standards.

This choice is a non-situated implementation of noise: in the real world, noise is linked to a situation. Take for instance the avoidance interference problem described in section 4.3.5. The real robot implementation indeed suffered from such poor representation of noise in the simulation.

However, it also suffered to a great extent from the choice of the IR tower to simulate the locality of the communication device. This choice challenged one of the basic assumptions of the research, namely that the range of the communication device could be accurately described by a circle [Winfield, 2000]. This was not the case on the real robots presented in this thesis, but we believe that effort put into the design of a specific device should provide the desired regular locality, and lead to better results. Nevertheless, the β algorithm still performed well despite this adverse context, which shows its intrinsic high resilience to noise.

In fact, the similarities we observe between the behaviour of the simulations and the real robots strongly validate both implementations of the β-algorithm on these different platforms.

Chapter 5

Taxis

" Nel farsi di ogni avvenimento che poi grandemente si configura c'è un concorso di minuti avvenimenti, tanto minuti da essere a volte impercettibili, che in moto di attrazione e di aggregazione corrono verso un centro oscuro, verso un vuoto campo magnetico in cui prendono forma: e sono, insieme, il grande avvenimento appunto. In questa forma, nella forma che insieme assumono, nessun minuto avvenimento è accidentale, incidentale, fortuito: le parti, sia pure molecolari, trovano necessità - e quindi spiegazione - nel tutto; e il tutto nelle parti. "

L'Affaire Moro

Leonardo Sciascia

The movement of the multitude is all a question of coordination. The individual in a crowd is always confronted by the movement of its peers and the attention needed to avoid collisions makes a persistent destination easy to forget. Ideas to overcome this difficulty without the help of direct control of the individual, which very soon becomes intractable as the multitude grows, involve either the transformation of the environment, drastic constraints on the movement allowed or submission to some "coordinator".

Take road traffic as an example. The centralised framework that underpinned the use of traffic lights to coordinate the flow of cars has been in many cases progressively replaced by roundabouts. In the transformed roundabout environment, traffic coordination emerges from the interplay of the constrained environment and the learned behaviour of the drivers. More general coordination in road traffic relies also on severely constrained behaviours of the drivers - think of the convention on the side of the road used. Also submission to a road "coordinator" implies some confidence in his "insightful" guidance.

Now that the aim of a coherent swarm has been achieved, the question is whether its global movement is condemned to remain random or whether a coherent swarm that displays *taxis*[1] behaviour is possible. In this chapter not only will it be shown possible, but also that the solution does not require transformation of the environment or choice of a leader and involves only a limited constraint on the individual's movement.

5.1 Environmental Incentive

In this investigation of swarm movement, the aim is to show that it is possible to control the movement of the swarm by providing an environmental "attractor". To this end, a *beacon* is introduced into the environment, that is, an environmental cue to break the evenness of the environment. To enable further real implementation, it has been chosen to simulate a source of light with its decaying intensity (see chapter 3). As a result, at each position in the environment we measure a possibly noisy lighting value, a *gradient*, that could be exploited to sense the direction of the beacon.

When following a beacon gradient (chemical, sonic or light...) an obvious solution is to place two different sensors on both side of the robot (see figure 5.1) and then make the robot turn towards the sensor indicating the highest value. This stereo implementation is highly dependent on signal-to-noise ratio as the robot's sensors are not situated far from each other and the two different sensing values are therefore very similar.

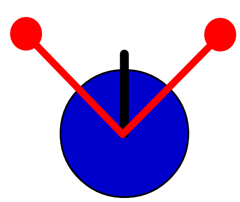


Figure 5.1: a stereo sensors solution

When trying to simulate such a behaviour in the robots of the coherent swarm of chapter 4, it was quickly evident that the changes in direction induced by the stereo beacon sensors were completely overwhelmed by the changes in direction needed by the β-algorithm, that had to have precedence to

[1]from the greek $\tau\alpha\xi\iota\sigma$ that refers to the organisation of battalion on the battlefield

maintain coherence. The random movement of the swarm was, therefore, not altered at all.

It has been shown in chapter 4 that the results of the distributed selection algorithm provides us with a possibility to choose special robots (section 4.5). The answer could then be to make these robots take a sample and somehow share it with their neighbours, generating an idea of the position of the beacon to then make the whole swarm move towards it (see [Nagpal, 1999, Dudek et al., 1993] that present algorithms to generate the position).

Although the problem of localization of the beacon might seem the most difficult to answer, the problem of coordinated movement represents an equal challenge. How is it possible to make a group of robots that individually do not have any precise idea about relative directions or positions follow a precise direction ?

5.1.1 biological examples

The most striking biological example is the behaviour of the slimemold *Dyctiostelium discoidum.* As described in chapter 2, in the context of scarce resources, the amoeba aggregate to form as a whole a moving slug, in order to get back into some resourceful place. It has been shown that this movement is the result of differential cellular adhesion [Savill and Hogeweg, 1997] and actually it will be shown that the solution investigated here is close to this example, at least conceptually (see section 5.5). *Myxobacteria* and heterocystous[2] *Cyanobacteria* show also an ability to move as a whole, but the precise mechanism involved has not yet been discovered [Adams, 1997, Shapiro, 1988].

Social insects are the living existence proof that coherent swarming is possible; bees, ants and crickets to cite just some of the examples [Bonabeau et al., 1999, Greenfield, 1994, Melhuish, 1999c]. The behaviours often involve chemical or sound emission that is relayed by the individuals, inducing a positive feedback. It is known as *secondary swarming* [Melhuish et al., 1999a]. An example of interestingly efficient stereo-sensor taxis is the example of cricket phonotaxis, in which neuronal cross-inhibition has impressive signal filtering capabilities [Webb et al., 2003].

Last but not least, vertebrates such as some fishes, mammals or birds behave as a school, herd or flock. But in these cases the capabilities of the individuals are much more developed than bacteria or insects, and it remains an open question what the mechanisms are to allow them to display such behaviours [Endler, 1993].

[2]that has the ability to differentiate into heterocysts

5.1.2 robotics examples

In the robotics literature, we have several examples of global movement, partly taking inspiration from the biological examples cited above. This research divides into two groups: problems where the shape of the group is of importance - formation - and problems where only the group movement is important - swarming or flocking (see section 2.8.2). Typically, research in the first group often involves a limited number of robots whereas the second one is more concerned with scalable behaviours.

The work on formation concentrates on keeping the formation and the choice of a direction is usually not addressed. For a review of several examples see section 2.8.2. But the avoidance of obstacles is a problem considered as it can be destructive to the formation. Balch and Arkin present a solution based on global position that is able to avoid obstacles while maintaining the formation [Balch and Arkin, 1998]. Fredslund and Mataric present a neighbour-referenced algorithm using only local sensing that is able to reform the formation after the avoidance of an obstacle [Fredslund and Mataric, 2001]. The most interesting example on scalable formation control by Balch and Hybinette also shows very good obstacle avoidance but again the movement is not an exemple of swarm taxis [Balch and Hybinette, 2000]. In [Baldassare et al., 2003] the formation behaviour is evolved by selecting cooperating robots equipped with 4 light sensors. The fitness function biases the selection towards collective behaviour and interesting formations emerge from this evolutionary process. Another example also involving artificial evolution is the work of Trianni et al. on hole avoidance while maintaining formation [Trianni et al., 2004].

Because of its computer graphics flavour, the work of Reynolds on flocking did not address the problem of the movement incentive [Reynolds, 1987]. But several other authors have investigated the task [Mataric, 1992, Melhuish et al., 1999a, Melhuish, 1999a, Hayes et al., 2000]. [Melhuish et al., 1999a] differentiates between examples where the robots happen to be together because they are attracted by the same beacon (*swarming* in the definition of section 2.8.2) and the task where the robots play an active role in maintaining group relationships among themselves (*flocking*).

One possibility to render the robots active in this sense is to let them relay the signal of the beacon by some signaling ability. It is referred as *secondary swarming* [Lewis and Bekey, 1992, Melhuish, 1999a, Hayes et al., 2000] and results in a better performing swarm as the decay of the signal is bypassed by the signaling robots acting as repeaters. In [Hayes et al., 2000] a probabilistic modeling approach to the secondary swarming/flocking task is investigated in order to give more insight into the crucial parameters at play.

Weßnitzer and Melhuish present an interesting algorithm for target hunting where the robots share their perceived progress towards the target within the neighbourhood making the most successful ones keep their direction and the others follow [Weßnitzer and Melhuish, 2003]. It results in the cooperative hunting of the prey with the swarm acting as an active catching net. Because of line-of-sight obstruction, some of the robots cannot see the prey. The robots that do see it then act similarly as a secondary beacon.

An interesting work that stands in between the two groups is the work by Quinn et al. which presents the investigation of the possibility for a small group of very simple robots to move collectively by selecting the controllers through *artificial evolution* (see section 6.3) on their ability to make the group move [Quinn et al., 2002a, Quinn et al., 2002b]. In this case the successful group chooses a formation to maintain, but this choice is only a successful strategy given the parameters, and not a prerequisite of the research.

Work concerned with modeling the behaviour of amoeba is presented in [Yokoi et al., 1998, Takahashi et al., 20 It is interested in modeling the slug part of the social amoeba life cycle in implementing a model based on vibrating potentials. Differential vibration abilities make the whole group move in the direction of the beacon by changing the vibration of the "cells" nearest to the beacon [Takahashi et al., 2004]. The drawback of this approach is that it needs physical bonds to convey the vibration that then make a change in configuration difficult.

5.2 Taxis Behaviour

5.2.1 description

As stated above taxis can be decomposed into a localisation task and a movement task. But this decomposition raises the problem of adequacy between both tasks: the localisation should provide the robot with information that it can make use of in order to perform a sensible move. Clearly in the two beacon sensors example the directional information is useless as the robot changes direction constantly because of the β-algorithm.

In fact, instead of considering the questions of localisation and movement as two different problems to be solved separately, the answer seems to be in binding these two problems together.

localisation

Firstly the robots are equipped with an omnidirectional light sensor. This sensor is put on the robot body such that the presence of another robot on the line-of-sight to the beacon occludes the illumination (see chapter 3). It is assumed that the signal-to-noise ratio is high enough for a robot to distinguish between sensing the beacon directly and the ambient light.

The result is a swarm that senses the beacon only - with some noise - on the side that is nearest to it, see figure 5.2. It has to be stressed here that there is no measure of signal strength that gives an estimate of the distance. Real robots should react to a threshold value that discriminates between ambient light and direct beacon illumination. Nevertheless the illuminated side of the swarm provides as a whole a direction to follow. It will be shown that this restricted information is sufficient to allow the swarm to move towards the beacon.

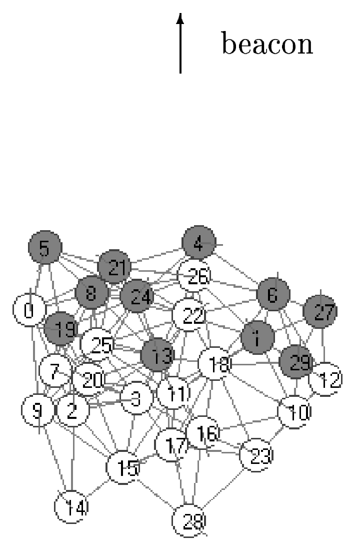


Figure 5.2: an illuminated swarm

movement

The idea is to make an illuminated robot become special to its neighbours by entering a new state - referred to as the "red" state. A layer is then added on top of the shared neighbour behaviour that always triggers a robot's reaction if the lost connection involved a red robot; this holding whatever the number of required shared neighbours β. The information of being "red" needs to be broadcast to the neighbours to allow them to react to it. The available bandwidth will only be slightly affected as the increase per message corresponds to a number of bits equals to the number of the robot's neighbours.

It has been shown in chapter 4 that swarms with higher β threshold tend to remain in the same place while smaller β values enable for more random movement. In fact the special state "red" corresponds to setting the β value to infinity for distinctive robots.

This results in the red robots trying to build complete graphs among themselves, reacting to each loss, clumping together and therefore restricting their global movement. Meanwhile the others are drawn towards the red ones, surrounding them and hence getting themselves into the light. As the current red robots build their complete graph, they occlude the red ones that happen to stand inside. This leads to a configuration similar to that at the start, but - importantly - the restricted movement of the red part of the swarm has attracted the other robots slightly in the direction of the beacon. The process then repeats itself and a slow movement proceeds towards the beacon (see figure 5.3).

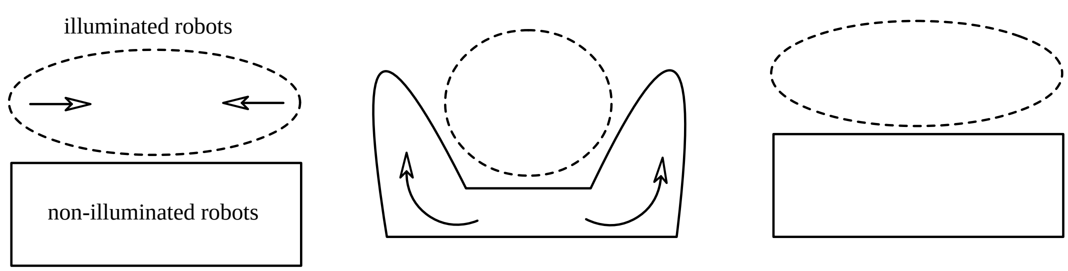


Figure 5.3: schematic suite of the taxis process

It must be noted that if the explanation above appears to dissect the process into distinctive phases, this does not correspond to the actual phenomena which occur continuously and dynamically. Such phase separations are made only for the sake of clarity.

As a result, the taxis process is not implemented through direct coding. It is a behaviour that emerges from the interaction of the illuminated and non-illuminated robots. What is introduced in the code is a difference in behaviour that translates itself into a *bias* for the swarm towards the beacon. Figure 5.4 presents the modification to the pseudo-code needed for the implementation of the taxis behaviour.

5.3 Experimental Set-Up

This chapter only presents investigations in simulation. Indeed the taxis process of this chapter needs long runs to show meaningful results, as will be shown in the following sections. Because the real robot swarm is not quite successful in maintaining coherence, it was impossible to implement taxis on real robots without improving their performance running the coherence algorithm.

Another reason not to run real robot experiments is the relative lack of behaviour difference between real robot swarms running differing β values. As the taxis algorithm relies on differentiation, it is very likely that no movement will ever take place.

5.3.1 environment

Until now the environment has been considered as completely neutral. In order to give the necessary tropism to the swarm, a beacon is introduced in the environment north of the starting area. It is situated at a distance of 1000 units from the center of this area.

In order to put the movement of the swarm under environmental pressure, three circular light occluding obstacles are introduced in another set of experiments, one directly in line between the swarm and the beacon and the two others on both sides of the first one. After an initial random spread the beacon is sensed by a few robots that attract the swarm through the obstacles. This setup slows down the global movement and it seems that a minimal aperture between the obstacles is needed in order not to overwhelm the emergent bias of the taxis. These obstacles are situated on a straight horizontal line 150 units north of the center of the starting area. They have a diameter of 80 units and the gap between them measures also 80 units (see figure 5.5). In chapter 3, figures 3.6 and 3.7 present schematically both experimental set-ups.

```
Create list of neighbours for robot, Nlist
k = number of neighbours in Nlist
i = 0

loop forever {
  i = i modulo cadence

  if (i = 0) {
    Send ID message

    Save copy of k in LastK
    Set reaction indicator Back to FALSE
    k = number of neighbours in Nlist
    Create LostList comparing Nlist and OldList

    for (each robot in LostList) {
      Find nShared, number of shared neighbours
      if (nShared <= beta) {
        Set reaction indicator Back to TRUE
      }

  // this is new ...
      if (color of robot == red) {
        Set reaction indicator Back to TRUE
      }
    }

    if (Back = TRUE) {
      turn robot through 180 degrees
    }
    else if (k > LastK) {
      make random turn
    }

    Save copy of Nlist in Oldlist
  }

  Steer the robot according to state
  Listen for calls from robots in range
  Grow Nlist with neighbours IDs and connection info

  i++
}
```

Figure 5.4: pseudo-code for β-algorithm

Figure 5.5: a swarm oozing its way through the obstacles

5.3.2 measures

The measures used to test the performance of the taxis algorithm are as follows:

- *progress towards beacon.* The Y coordinate of the position of the swarm's center of mass at the end of the run gives measure of the speed of the taxis.
- *number of successful runs*, a successful run being a run that ends with a connected swarm.
- *edge-connectivity* which represents the minimal number of edges that have to be removed to disconnect the network, giving a measure of coherence.
- *odometry* which records the proportion of time spent in the different states (forward, turn, backwards and stop) over the whole run.

Each run lasted 1,000,000 time steps with a cadence value of 100. Hence, each robot sent 10000 messages. Each value was recorded once every 10000 steps and, except for the progress measure, has been averaged over the whole run. For a number of values of swarm size, β and noise, 5 runs have been performed and the result depicted is the mean value over these runs with its standard deviation. When the purpose of the investigation is not to vary them, the chosen values for the remaining parameters are shown in table 5.6.

size	20
cadence	100
random noise	2%
β	1 or 2
steps	1,000,000
runs	5

Figure 5.6: general parameter values for β-algorithm

5.4 Results

The results of the simulation runs are depicted in figures 5.7 to 5.20. It is firstly shown that there indeed is a movement of the swarm in the direction of the beacon and that obstacles can be avoided. The influence of noise is investigated afterwards.

5.4.1 results with and without presence of obstacles

Figure 5.7 shows the progression of the swarm towards the beacon with increasing swarm sizes and change in β parameter. The first observation is that a value $\beta = 1$ is not enough to guarantee the cohesion of the swarm as almost all runs finish disconnected (see figure 5.8, where most values for $\beta = 1$ are empty). Therefore the good performance of the few runs that completed cannot be considered as conclusive. On the other hand with values $\beta = 2$ or 3, the proportion of good runs increases and a net movement in the direction of the beacon is observable. The curve of the results suggests the presence of an optimal swarm size near $N = 20$ and an optimal value $\beta = 2$.

The difference of performance when changing the β value is explained by the fact that the process relies on differentiation between illuminated and non-illuminated robots. By raising the β value for the non-illuminated robots the differentiation is lowered and hence the taxis performance degrades.

The lower performance of smaller and larger swarms suggests that the proportion of differentiated robots within the swarm is an important factor. In small swarms almost all robots are illuminated while this proportion is inversed in larger swarms.

But any discussion on performance is somehow missing the point as the speed of the swarm is tremendously slow. Indeed the length of the run is 1,000,000 steps and in this number of steps, a

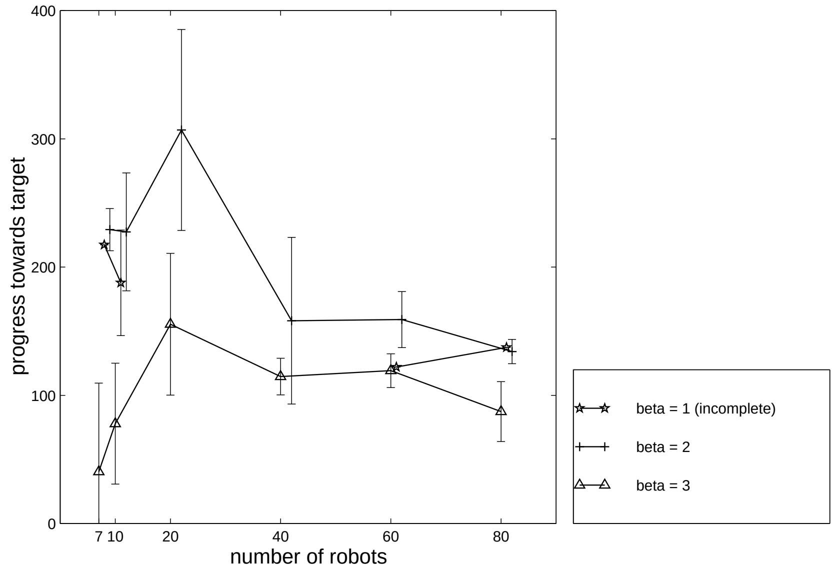


Figure 5.7: progression for taxis behaviour

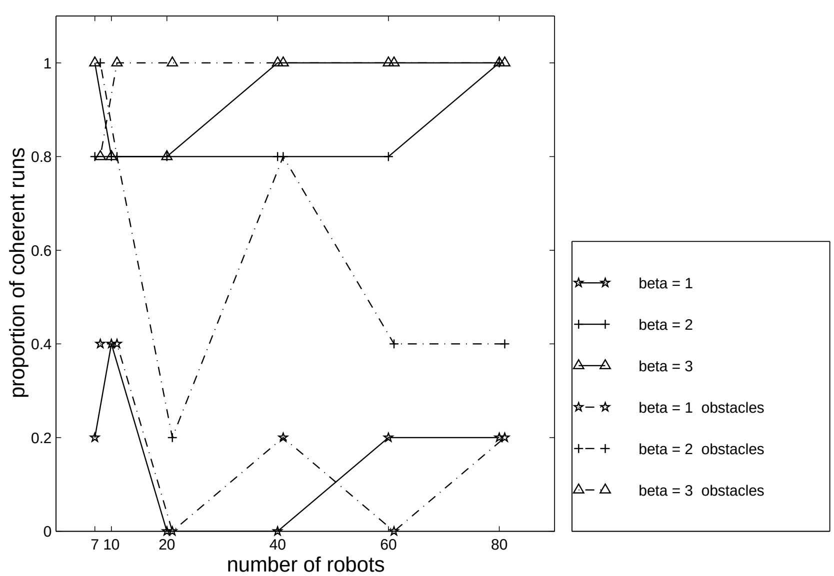


Figure 5.8: successful runs for taxis behaviour

lone robot going in straight line can travel 10,000 distance units. Which means in the best case that a robot spends roughly one twenty-fifth of its time moving towards the beacon. This can never be regarded as good performance.

In fact, the interesting feature of this algorithm is not the speed of the movement but the fact that movement is taking place at all, and this without the help of a directional sensor. The reduced performance with larger β values or larger swarm sizes is counterbalanced by confidence that the movement is such that the swarm is eventually going to reach the beacon.

In the presence of obstacles, this confidence does not hold in all cases, as can be seen in figure 5.9. While a value $\beta = 1$ shows the same brittleness, it can be seen that swarms of 10 and 20 robots with $\beta = 2$ are able to move through the obstacles. But in the case of $\beta = 3$ for size greater than $N = 7$ for instance, the requirements on connectivity induced by the β-algorithm does not leave enough malleability on the swarm to let it ooze through the obstacles. In this case, the disposition of obstacles represents a trap for the swarm as the attraction draws the swarm towards a gap it cannot go through. This impossibility is of course dependent on the size of the gap between the obstacles. Nevertheless the ability of the swarm to find the way between the obstacles is quite impressive, sometimes showing very interesting behaviours as in figure 5.11.

Also larger swarms with $\beta = 2$ tend to disconnect themselves in the presence of obstacles. Indeed the attraction on the part of the swarm that is already gone through the obstacles is stronger that the connectivity glue of the β-algorithm (see figure 5.8). Figure 5.10 shows a disconnected swarm of 60 robots experiencing this problem.

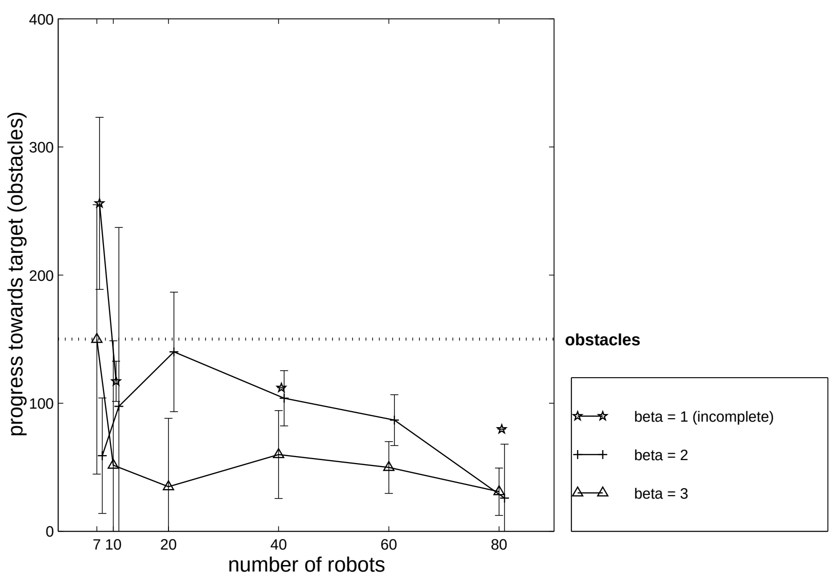


Figure 5.9: progression for taxis behaviour with obstacles

The case of $N = 7$ with $\beta = 3$ is interesting as it shows better performance with than without the obstacles. An explanation is that the obstacles prevent the deleterious over-illumination of the

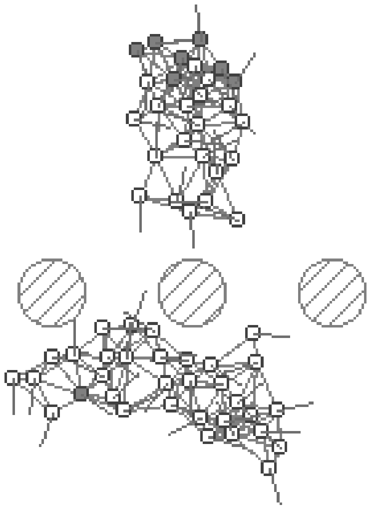


Figure 5.10: a broken swarm of 60 robots

swarm and that the size is too small to stop the swarm going through them.

Taking a look at the edge-connectivity (figure 5.12) shows that the taxis algorithm does not affect its tunability, indeed differing values of β induce differing connectivity values. These values are comparable to the non-moving case. Also connectivity does not seem to be affected by the presence of obstacles.

The comparison of the odometry plots (figures 5.13 and 5.14) shows that the local behaviour of the robots is not sensitive to the presence of obstacles, meaning that the time spent turning to avoid the obstacles is negligible in comparison of the turns required by the β-algorithm.

5.4.2 influence of noise

To measure the precise influence of noise on the taxis process requires two steps. Firstly noise will be increased on all possible sources of noise simultaneously, namely noise on actuators, avoidance sensors, communication device and illumination sensor . Secondly noise on communication will be fixed at a level of 2% and only the remaining sources will be varied. The results are depicted in figures 5.15 to 5.17 for the first step and 5.18 to 5.20 for the second step.

When noise is increased on all sources at the same time the degradation of performance is quite strong (figure 5.15), with a result at 10% of noise that shows a possibility of negative movement, that is movement away from the beacon. Nevertheless with intermediate levels of noise, movement is still

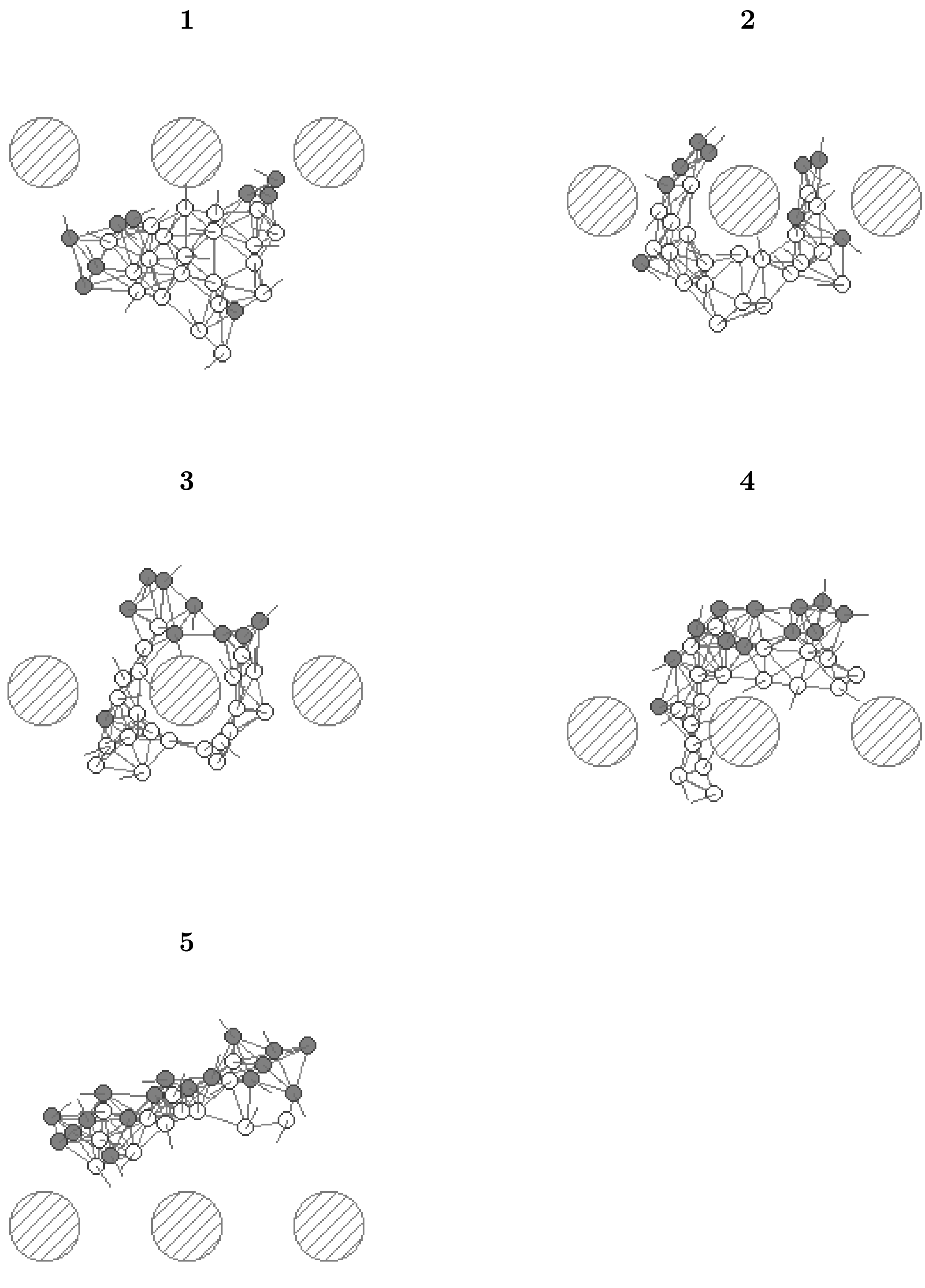


Figure 5.11: progression through the obstacles for a swarm of 30 robots

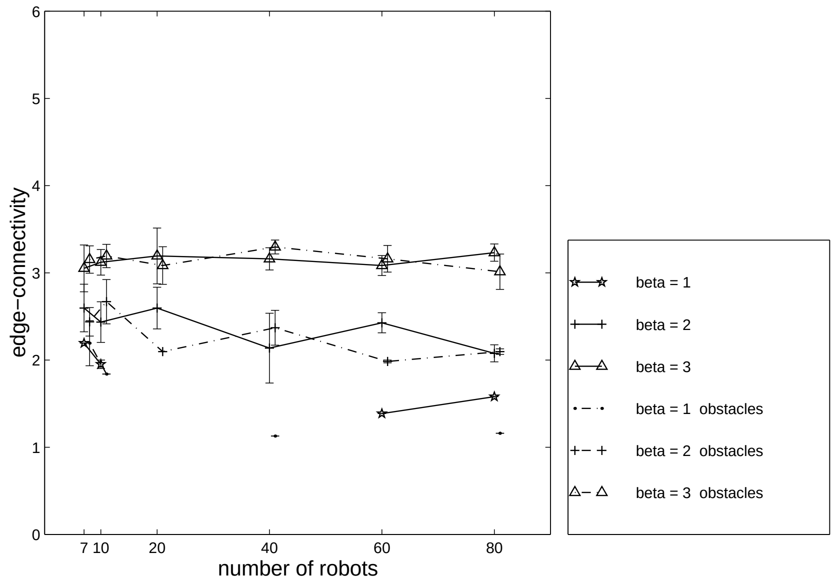


Figure 5.12: edge-connectivity for taxis behaviour

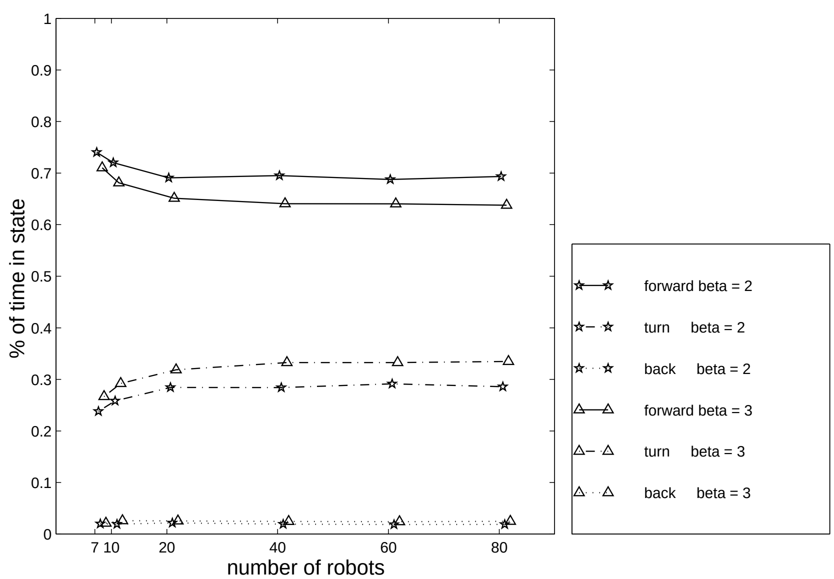


Figure 5.13: odometry for taxis behaviour

taking place, thus demonstrating the robustness of the solution. Also the difference in performance with differing swarm sizes appears to be reduced as noise increases.

Connectivity is maintained despite the increase in noise (figure 5.16), as was already observed in chapter 4. Only a slight general decrease is observable and that could be due to the small number of testing runs.

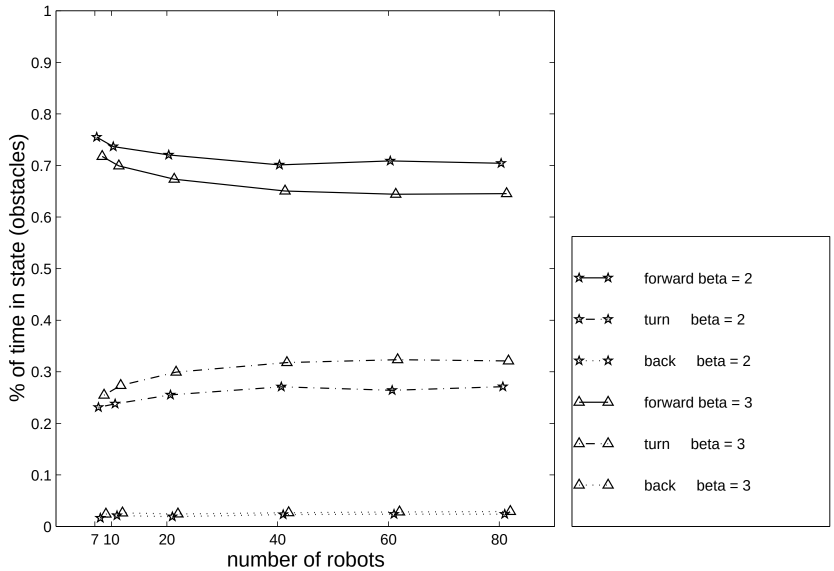


Figure 5.14: odometry for taxis behaviour with obstacles

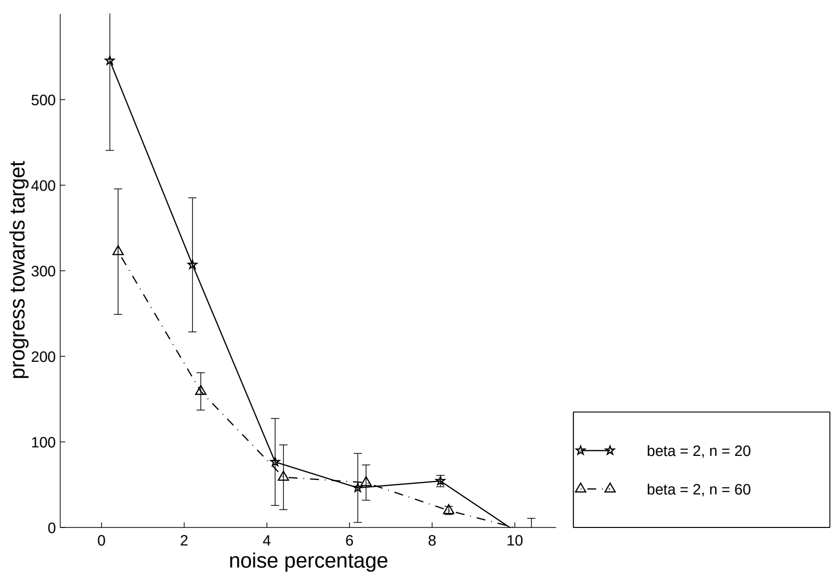


Figure 5.15: progression vs noise for taxis behaviour

Considering odometry (figure 5.17) the characteristic pattern of influence of noise seen in chapter 4 is again observable, providing us with an explanation for the degradation in performance: the taxis process relies on the fact that some of the movement of the robots is going in the right direction while the illuminated robots function as an anchor such that robots going in the wrong direction eventually choose a better one. If the proportion of straight motion is reduced in favour of turning

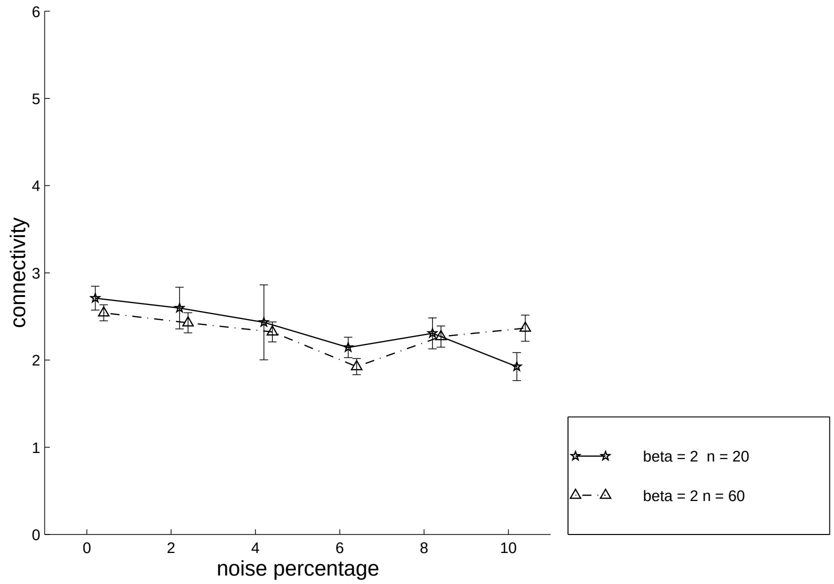


Figure 5.16: edge-connectivity vs noise for taxis behaviour

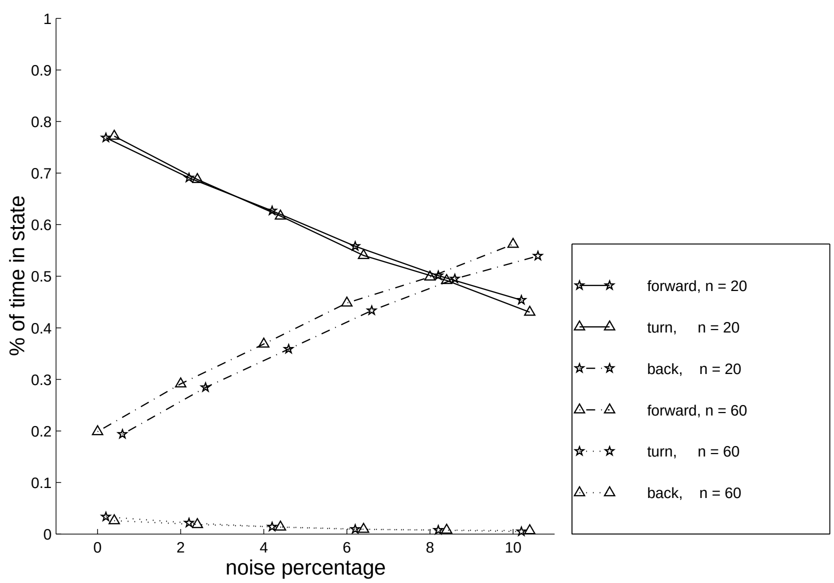


Figure 5.17: odometry vs noise for taxis behaviour

motion because of the need for reconnection, then global movement can only be reduced also.

But the increase of the proportion of the "turning" state is linear, whereas the decrease of performance is exponential, which suggests that the explanation above is not sufficient to explain the degradation. Actually, as already mentioned, the movement results from differentiation between the illuminated and non-illuminated robots. This differentiation lies in an increase of the β threshold, leading to a greater reactivity of the illuminated robots. The increase of noise increases the reactivity of all robots indiscriminately, which levels down the differentiation. Hence the decrease in performance.

When the level of noise on the communication device is fixed at 2%, the degradation of performance is greatly reduced (figure 5.18). The slight decrease in connectivity is no longer to be seen(figure 5.19). And the plot of the odometry shows a striking constancy (figure 5.20).

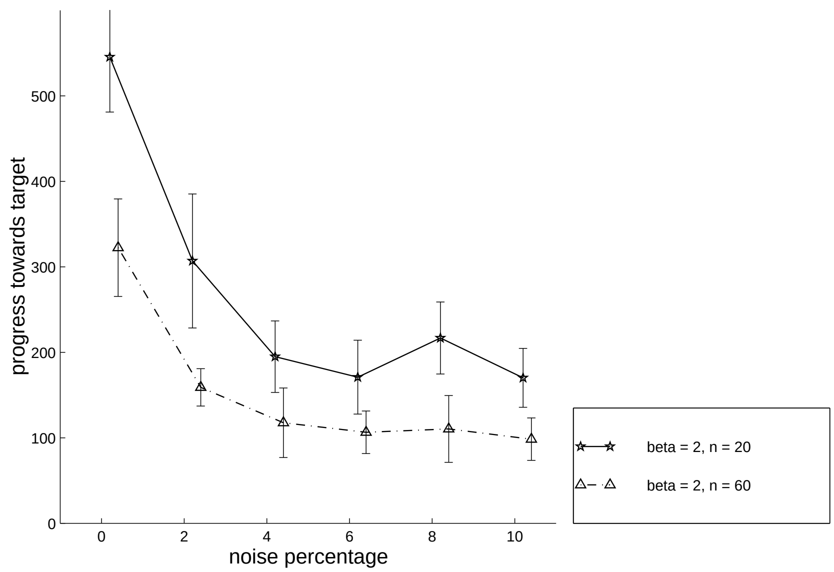


Figure 5.18: progression vs noise for taxis behaviour (communication noise fixed at 2%)

These results show that the degradation in performance with increasing noise is mainly due to noise on the communication device. Considering that the loss of 2% of messages represents, by current standards a very poor communications channel, there is good confidence that such a signal-to-noise ratio is achievable on real robots (if the problem of locality is solved, see section 4.3.5). This shows the robustness of the solution developed. Indeed the behaviour is strikingly insensitive to an increase in noise on the light sensor and this is of considerable interest for potential applications.

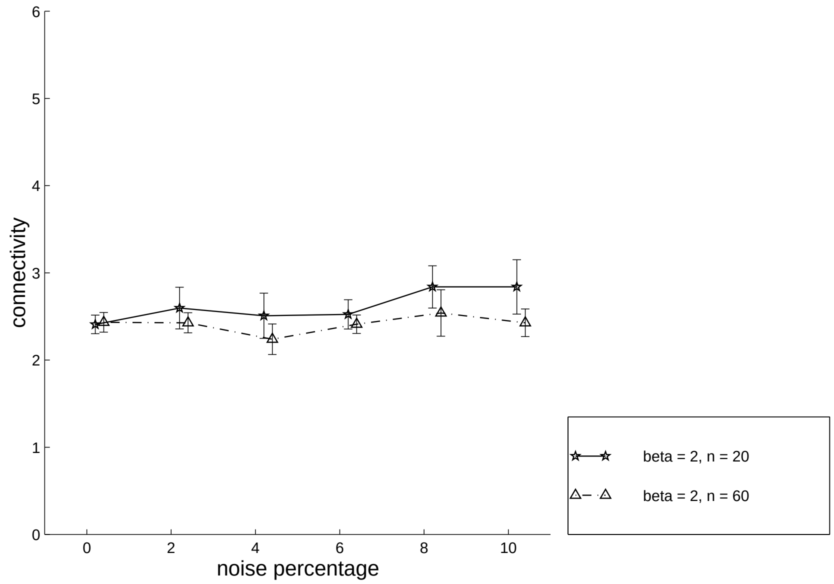


Figure 5.19: edge-connectivity vs noise for taxis behaviour (communication noise fixed at 2%)

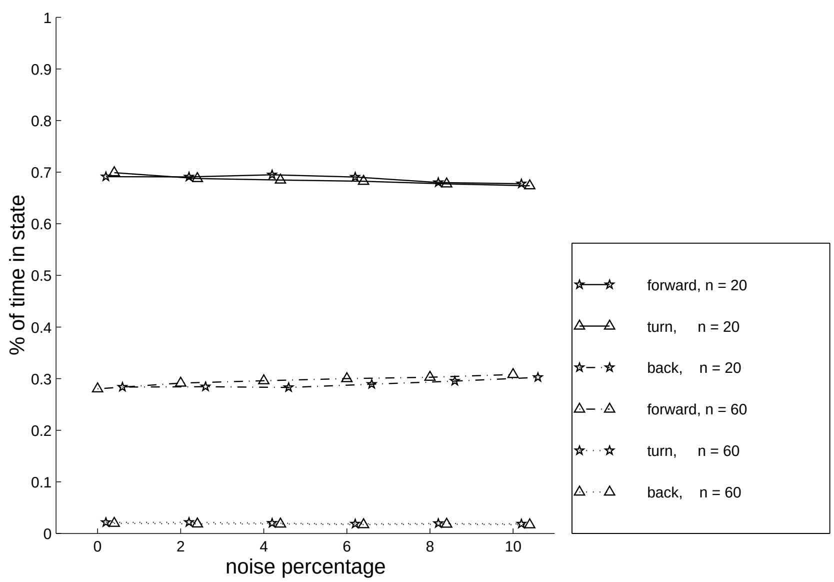


Figure 5.20: odometry vs noise for taxis behaviour(communication noise fixed at 2%)

5.5 Summary of the Chapter

movement is truly emergent

It is important here to stress that the sensing of "being illuminated" or not does not allow a single robot (a swarm of 1) to reach the beacon. The taxis behaviour only results from the subtle interaction of the illuminated robots and the others. This feature makes the behaviour truly emergent.

In fact, it could be argued that this behaviour actually represent a higher level of emergence than is commonly meant, in the sense that not only does it involve the interaction of the robots with the environment but also the interaction of the robots themselves. As a comparison, an ant-like clustering task is not as dependent on the other individuals: a lone robot is able to achieve the task, though in a longer time. In the case presented, the interaction between the robots is as crucial, if not more crucial, than the interaction with the environment. However, recalling the definition of emergence that was chosen in section 2.4.2, namely that the notion of emergence is *"a process through which entirely new behaviours appear, whose properties cannot be derived from a given model of how the system behaves, so that another model has to be built in order to deal with these new behaviours"* [Bonabeau and Theraulaz, 1994], implies the definition of a model (that can be seen as the observer model) in order to build the new describing model. Therefore the differing levels of emergence should be represented by a chain of describing models. Hence one model would describe the emergence of the coherence behaviour from the local rules, while another would present the taxis algorithm emerging from the interaction between the coherence behaviour and the differentiation induced by the beacon. Following the same definition, a model could describe the taxis behaviour directly from the local rules and the beacon influence on these local rules, losing the relevance of different emergence levels. Here is indeed an example that the notion of emergence is dependent on the observer.

As an emergent behaviour, the presented taxis behaviour is highly dependent on the various parameters in action: communication range, rate of occlusion, avoidance range, etc. These dependencies remain to be investigated in real robots experiments but the simulation results already predict that the conditions of noise and swarm size are determinants for success, especially for communication noise and in the presence of obstacles.

movement is possible through differential cellular adhesion

It has been demonstrated that cellular adhesion plays a crucial role in the *Dyctiostelium discoidum* slug migration [Savill and Hogeweg, 1997]. And as suggested in chapter 4, the β threshold value can be considered as an adhesion value between the robots. In introducing the environmental cue together with the extension of the original β-algorithm, this adhesion is differentiated over the swarm according to an external incentive, which provides the swarm with both direction and movement at the same time.

Although the process is not quite the same as in *Dyctiostelium*, where the production of the chemical cAMP is involved, this example of movement through differentiation is very similar in its essence. This similarity stands as a justification for this approach, for social amoeba show morphogenetic behaviours without apparently involving other abilities [Hogeweg, 2000b, Hogeweg, 2000a, Hogeweg, 2002]. The next chapter is devoted to the investigation of similar behaviours.

A feature also suggestive of the behaviour of an amoeba is what happens when the swarm reaches the beacon: the interplay of the avoidance behaviour and the beacon attraction makes the swarm enclose the beacon, similarly to the phagocyte behaviour of the amoeba (see figure 5.21). Although it remains to be properly investigated, this behaviour could be of considerable interest for real applications.

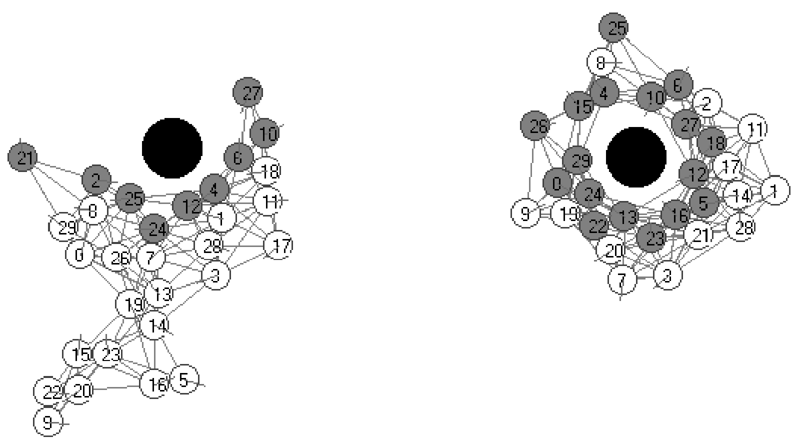


Figure 5.21: amoeba-like enclosure of the beacon

obstacle avoidance comes for free

The interplay of the local avoidance abilities of the robots with the taxis behaviour gives the whole swarm the ability to travel around occluding obstacles, while maintaining coherence. This behaviour is not coded in the algorithm and is again emergent.

When it is situated behind an occluding obstacle, the swarm actually functions as a distributed sensing network. It spreads moving randomly until one of its bounding robots gets beyond the shade of the obstacle and becomes illuminated by the beacon, starting the taxis process in the direction of this lighted area. As soon as this area is reached, the swarm starts moving in the right direction.

drawbacks and advantages

One of the drawbacks is the sensitivity to noise in the communication device that has the power to blur the differentiation induced by the light beacon. But as already mentioned the proportion of loss of messages considered are huge compared to current standards of communication performance, which give good confidence on the realisability of the algorithm.

Another drawback is the relative non scalability in the performance of the process to an increase in swarm size. Actually the performance is very low in any case, so a better performance measure would rather be whether movement is taking place or not. It has been shown that it is in most situations the case.

Perhaps the main drawback that might prevent the use of this algorithm in many real world implementations is its relatively slow speed. On the other hand its robustness to beacon noise makes it very interesting to applications which are not time-dependent.

Another advantage is the absence of any true positional information to carry out the global taxis. Indeed, apart from the crude information given by the neighbours messages and the added "redness" information, there is neither reference to nor exchange of position information in the algorithm. This is a huge advantage considering the fact that position information has always been difficult to produce and maintain accurately.

As the β-algorithm is dimension independent, so is the taxis extension. As a result, there is no reason why the taxis behaviour could not be straightforwardly implemented in three dimension, with of course an adaptation of the crucial parameters involved.

However, the most important advantage is the fact that the swarm moves as a whole. Indeed the movement is not a sum of all individuals moving towards the target that happen to be together because they move towards the same target. In our case the movement results from a complex interaction that allows for the maintenance of coherence while moving.

real robot implementation

The level of noise induced by the device used to simulate the locality of the connection, together with other sources of noise, was so high that changes in β threshold values did not have a sufficient influence on the swarm behaviour, as can be seen in the results of the real robot experiments on coherence (see section 4.4.3). Therefore, the essential differentiation could not take place, and the current platform made real robot experiments on taxis behaviour impossible.

Chapter 6

Shape

"The art of urban civilisations tends to be static,solid and symmetrical (...); to a greater and lesser extent nomadic art tends to be portable, asymmetric, discordant, restless, incorporeal and intuitive (...)."

Bruce Chatwin.

The swirling flock of birds, the crawling *Dyctiostelium* "slug" or the static insects' swarm in a field on a summer afternoon, all of these state the question of how to make a multitude move, not only in a distinctive direction, but also in a distinctive manner.

In these examples the shape that is formed remains constant while the place taken within it by the moving individuals changes constantly. What then is the reason for this constancy ? Are there internal or external causes ? Is it somehow mediated between the individuals ? Is there a plan ?

morphogenesis

In raising these questions, this thesis steps into the world of self-assembling shapes, growth and forms, or to put it another way the world of *morphogenesis.* These problems have of course already been beautifully solved by nature and numerous biologists have celebrated nature by studying the striking capacity for growth, repair and pattern generation of live beings [d'Arcy Thompson, 1917, Shapiro, 1988].

Starting with the pioneering work of Alan Turing [Turing, 1952], there has been constant exploration of the capacity of abstract models to generate patterns found in nature. Examples range from L-systems [Lindenmayer, 1968] or bacterial growth [Ben-Jacob and Cohen, 1997], to the modeling of the *Dyctiostelium* dynamics [Savill and Hogeweg, 1997]. These are a part of the Artificial Life framework (see chapter 2).

The area of research on computational models of morphogenesis has boomed in recent years thanks to the development of computer games with realistic three-dimensional interfaces. The speed requirements have placed pressure on the development of fast algorithms able to generate an abstract model of plants or textures, for instance. Indeed in these examples the use of pictures or drawings is rapidly and intractably consuming in terms of computer resources, as the purpose of the interface is to allow for the choice of any point of view.

Of course the requirements of an efficient game interface are not equivalent to those of a faithful model of *Dyctiostelium* dynamics, but ideas and solutions diffuse in both directions. The work of Reynolds on the flocks of "boids" was, for instance, initially developed for artificial reality [Reynolds, 1987].

dynamicity

Another pervasive feature in literature is the non-dynamicity of the morphogenetic process: once an element of the structure is assembled, either it is fixed or the algorithm will try to keep it fixed, as in the research area on robot formations (see section 2.8.2). But indeed this is not the case in Nature: the cells of the developing embryo have no fixed positions, and moreover biological research has shown the crucial role played by cell migration in the developmental process [Mombach and Glazier, 1996]. In its model of *Dyctiostelium* behaviour, Hogeweg tackles this problem with great elegance and his work stands as a powerful example that dynamicity together with differential adhesion are at the core of the morphogenetic process [Hogeweg, 2000b, Hogeweg, 2000a].

This chapter will investigate, in simulation only, the ability of the abstract β-algorithm to allow for the control of dynamic global shape. Let us here remind ourselves that dynamicity is a requirement of the β-algorithm and for this reason a solution to shape control has no choice but to be dynamic. We will first investigate the potential for pattern formation in the initial version of the β-algorithm (from chapter 4), with predefined heterogeneities within the swarm (in section 6.1). Next the light sensor extended algorithm used for the implementation of the taxis behaviour will be investigated (section 6.2) and finally a system developed for artificial evolution will be presented but only partially investigated (section 6.3).

6.1 Spatial Segregation

It has already been mentioned in chapters 4 and 5 that the β-algorithm could be considered as an adhesion mechanism with the β threshold as adhesion strength parameter. The work of Hogeweg suggests that some distinctive patterns could be dynamically formed by the variation of the adhesion strength over the population of the swarm.

In fact this differential adhesion is conjectured to be responsible for pattern formation in developing cells. One of the classical basic patterns that have been formed can be seen in figure 6.1 [Gilbert, 2002, Hogeweg, 2000b]

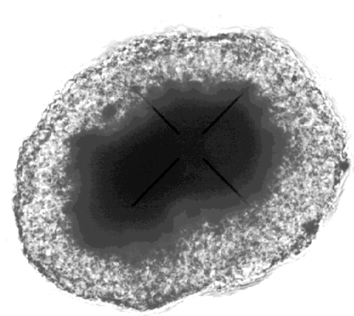


Figure 6.1: basic pattern found in developing living tissues

An interesting preliminary example, to be soon applied to robotics, is the work of Ottery, who built a model of biological cells to investigate differential cellular adhesion [Ottery and Hallam, 2004]. The importance of random forces to enable the cells to build new bonds is emphasised.

Drawing from the classical examples of figure 6.1, two experimental hypotheses can be proposed: is it possible to spatially discriminate between differing groups of robots, firstly segregating concentrically around the center of mass and secondly segregating radially (see schematic view 6.2)?

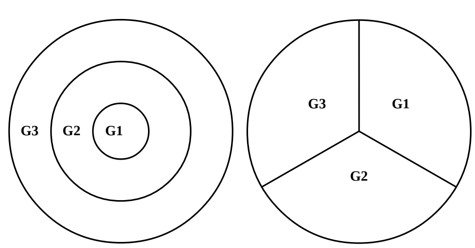


Figure 6.2: schematic view of the spatial segregation hypotheses

In other words is it possible to guarantee that one group would occupy the center of the swarm and another would distribute concentrically around it? Or to guarantee that the groups would distribute in differing segments around the center of mass, with a group in the center or not ?

From the point of view of distributed sensing an interesting feature that would be of value in a sensor network is the possibility of placing heterogeneous robots in a specified relative position. As the design presented precisely lacks this kind of information it is challenging to try to achieve such spatial segregation.

6.1.1 algorithms description

With the degree variant (α-algorithm) we have already attempted to increase the α threshold on the number of connections for a small number of robots. These robots were certainly more often near the center but the result suffered from the overall instability of the system.

concentric segregation

Applying the same idea as in the shared neighbour algorithm, we make use of the threshold β, in order to investigate how robots with different threshold values self-organise. This algorithm simply consists of setting different β values to the robots belonging to different groups. It will be referred to as the *concentric-β-algorithm*.

As a result, robots with higher β values are more sensitive to the quality and the number of connections in their neighbourhood. They react more to losses of connections and tend to stay at the same place. Therefore they group together while the robots with lower β values simply surround them. We will investigate the potential of this algorithm to discriminate between two and three groups of robots, with a proportion of one third/two thirds and equal thirds respectively. Figure 6.3 shows an example of 2-group segregation, while figure 6.4 shows an example of 3-group segregation.

radial segregation

To obtain radial segregation we propose to make a robot apply differing β values according to the group membership of the lost robot. This algorithm is named *radial-β-algorithm*. This results in a group being more reactive to losses from one particular group and less reactive to losses from another,

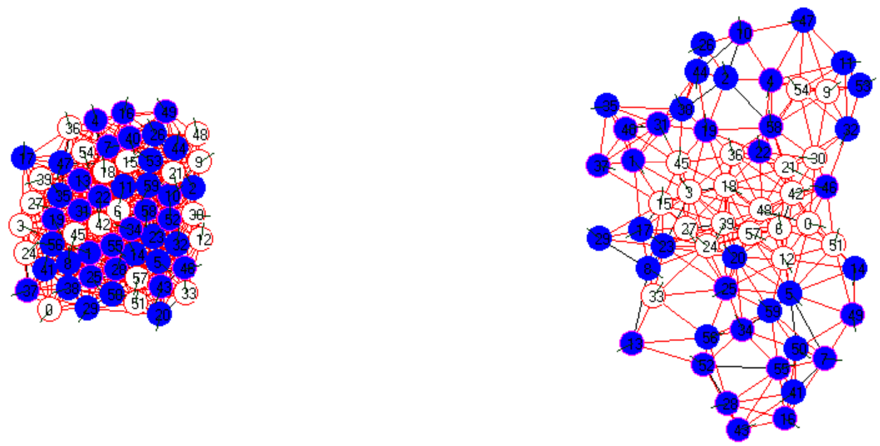


Figure 6.3: 2-group concentric segregation (start and resulting pattern)

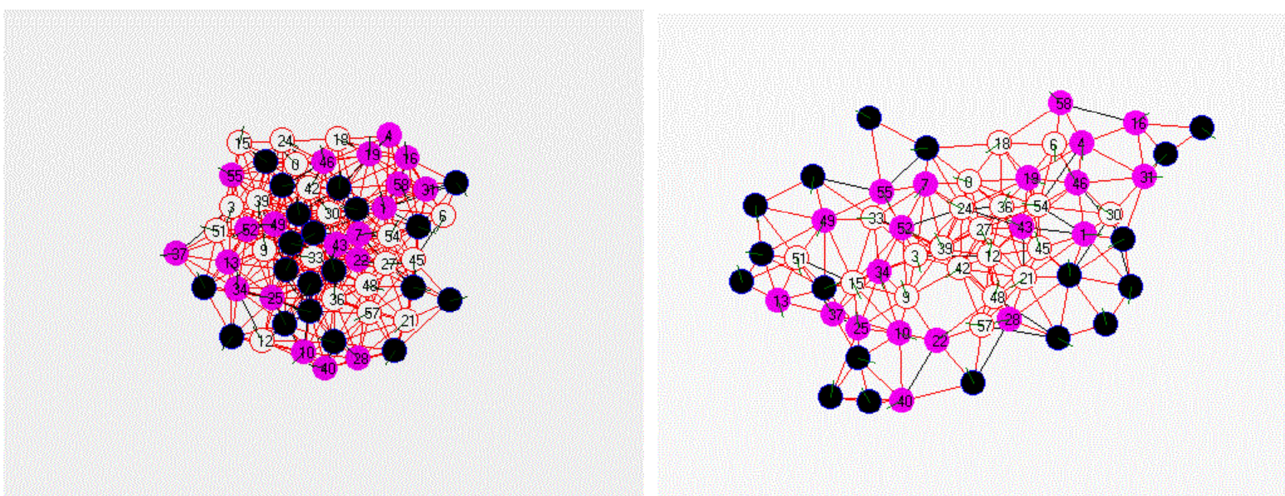


Figure 6.4: 3-group concentric segregation (start and resulting pattern)

while being specially reactive to the robots belonging to its own group. For this algorithm swarms composed of three groups will be investigated, with equal proportions for each groups.

The three groups example raises a question on the topology of the preferences. Taking a look at the schematic view in figure 6.5 in which the preferred groups are represented by an arrow towards them, the potential for fundamental differences can be seen. We will question whether the topological differences between situation **A** and situation **B** have their equivalence in the resulting swarm patterns, but topology **C** will not be considered. Figure 6.6 shows examples of resulting patterns with topologies **A** and **B**.

$$\begin{tikzcd} \bar{0} \arrow[rr] & & \bar{1} \arrow[dl] \\ & \bar{2} \arrow[ul] & \end{tikzcd}$$

A

$$\bar{0} \longleftrightarrow \bar{1} \longleftrightarrow \bar{2}$$

B

$$\bar{0} \longleftrightarrow \bar{1} \longrightarrow \bar{2}$$

C

Figure 6.5: different topologies of the group preferences

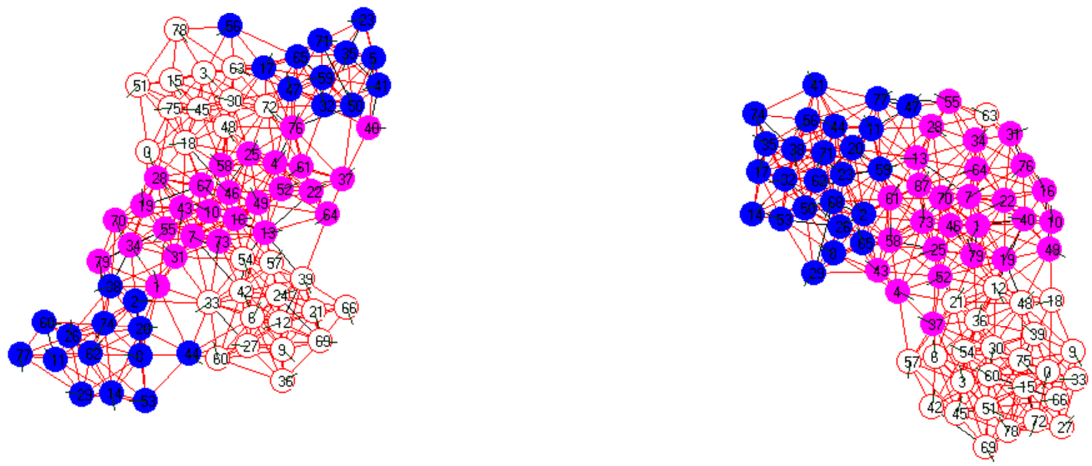


Figure 6.6: 3-group radial segregation resulting patterns with topologies A and B (5,000,000 steps)

6.1.2 measures

In order to quantify the performance of the different spatial segregation algorithms, the following measures will be used (see chapter 3):

- *group mean distance to the whole swarm center of mass* and its *standard deviation*, for each robot group the distance to the center of mass of the whole swarm is computed with its standard deviation to measure a potential difference between them.

- *mean minimum distance between individuals in different groups* to measure the distribution of the different groups, defined as follows:

$$d_{min} = \frac{1}{|G|} \sum_{r_i \in G} \min_{r_j \in G'} (d(r_i, r_j))$$

  This mean minimum distance is compared to a default value which consist of the mean over all pairs of group over 10 measures at the beginning of the run, when groups are randomly mixed together.

- *whole swarm mean distance to its center of mass* as defined in chapter 3 for comparison purposes.

The first measure is designed to test the concentric segregation and the second the radial one.

The groups are defined through the following rule: if $Robot_{ID} = \bar{i} \quad mod \quad 3$ then the robot belongs to group $\bar{i}$. This divides the swarm into three groups. In order to get two groups group $\bar{0}$ and $\bar{1}$ are merged. This results in the proportions mentioned above. In the case of 2-group concentric segregation, the merged group takes the lowest β value.

The length of the simulation run is 500,000 steps, in order to leave enough time for the robots to self-organise. Each value was recorded every 5000 steps and has been averaged over 10 values at the end of the run. For a number of values of swarm size, β and noise, 5 runs have been performed and the result depicted is the mean value over these runs with its standard deviation. When the purpose of the investigation is not to vary them, the chosen values for the remaining parameters are shown in table 6.7 ((7,3,1) means the β parameter of the group $\bar{0}$ is set to 7, the β parameter of the group $\bar{1}$ to 3, and the β parameter of the group $\bar{2}$ to 1).

size	20 or 60
cadence	100
β	(7,1) or (7,3,1)
steps	500'000
runs	5

Figure 6.7: general parameter values for segregation

For the 2-group experiments the β parameter of the group $\bar{0}$ is set to 7 and varies from 1 to 5 for the other group . For the 3-group experiments the β parameter of the group $\bar{0}$ is set to 7, the parameter of the group $\bar{2}$ to 1 and the parameter of the remaining group varies from 2 to 5.

6.1.3 simulation results for concentric segregation

The results for the concentric segregation experiments for a 2-group partition are depicted in figures 6.8 to 6.10. From 6.11 to 6.13 are shown the results for a 3-group partition.

Figure 6.8 shows the behaviour of the group distances to the center of mass with an increasing β value for the group with minimal value. A clear difference in the distances is visible, which levels down as the difference between the β values diminishes.

Figure 6.9 presents a more general view of the parameter influences. The difference between the distances for the group with increasing β value (group $\bar{1}$) and the group with fixed β value (group $\bar{1}$) is observable over the whole parameter space.

The influence of noise on the segregation performance is limited as can be seen in figure 6.10. The high variability of the standard deviation is due to the low number of different runs as only runs that finish with a connected swarm are considered in computing the performance measure.

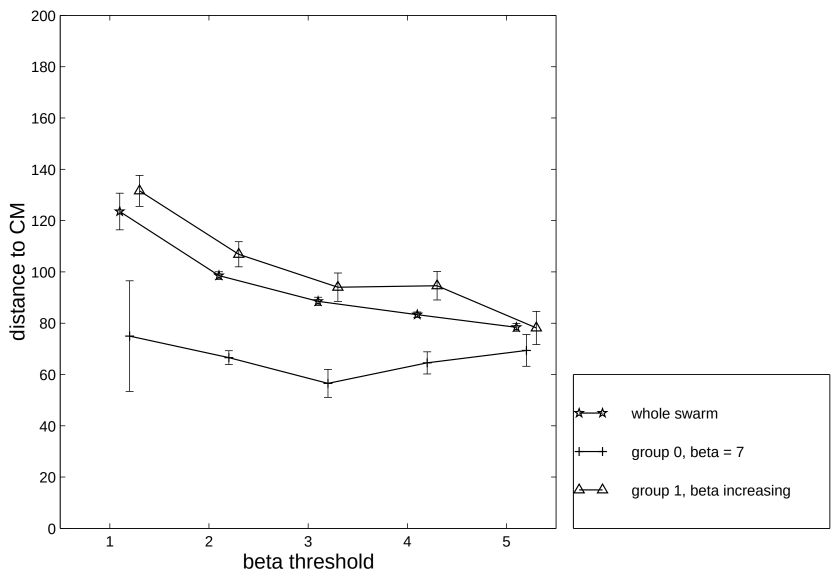


Figure 6.8: 2-group concentric segregation (n=60)

The behaviour of the group distances for a 3-group partition is depicted in figure 6.11 and the distribution meets with expectation. Indeed the group with an intermediate β value presents a mean distance to the center of mass inbetween the groups with maximal and minimal β value. The difference between the intermediate and the maximal value group levels down as expected. Figure 6.12 shows that the distribution in sequence of the preceding figure is independent of swarm size. 3-group segregation seems to be more sensitive to noise with increasing variability among the runs. But on the average the general character is conserved (figure 6.13).

The process of segregation is quite slow. Indeed each robot has to discover neighbours with stronger

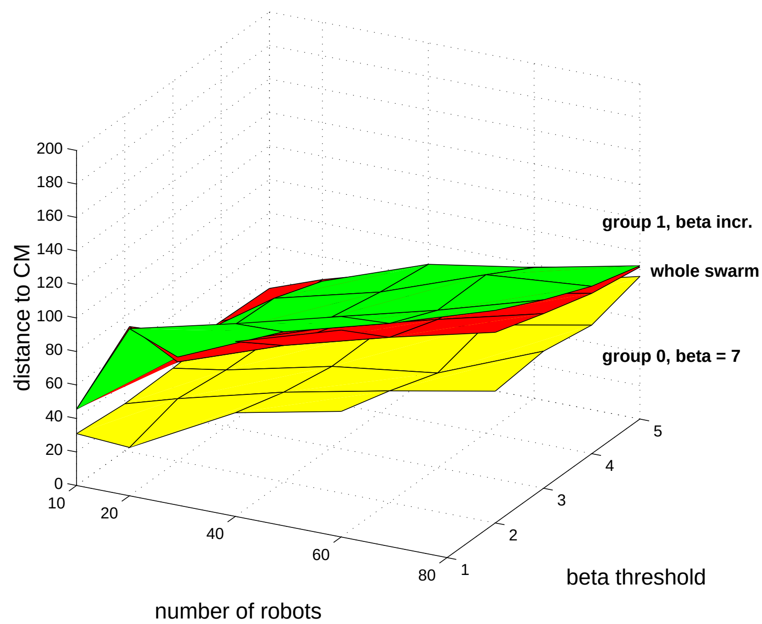


Figure 6.9: 2-group concentric segregation

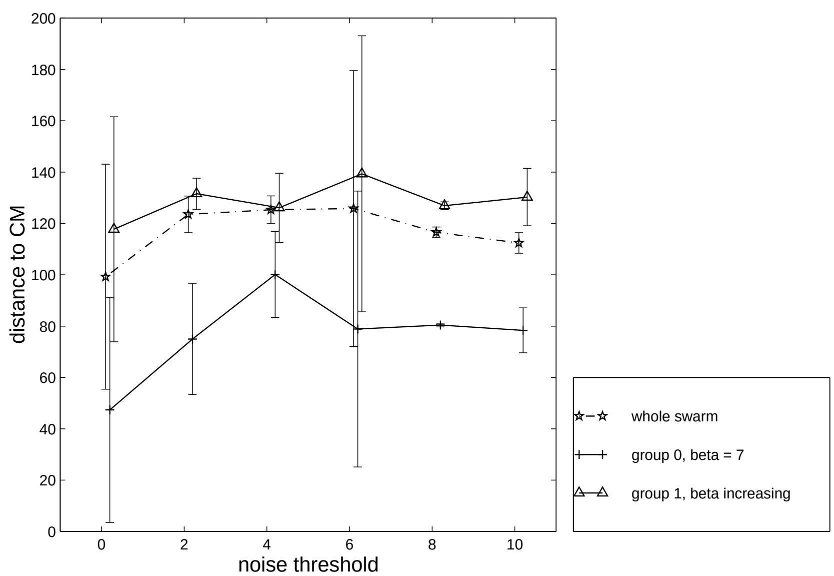


Figure 6.10: 2-group concentric segregation (n=60)

bonds by random movement while the requirements of coherence restrict freedom of movement within the swarm. This is analogous to the dynamics of area coverage of figure 4.43 (page 119).

The discussion above shows that heterogeneities in the β values across the swarm are sufficient to concentrically segregate the groups with differing values, with greater β values nearer to the center

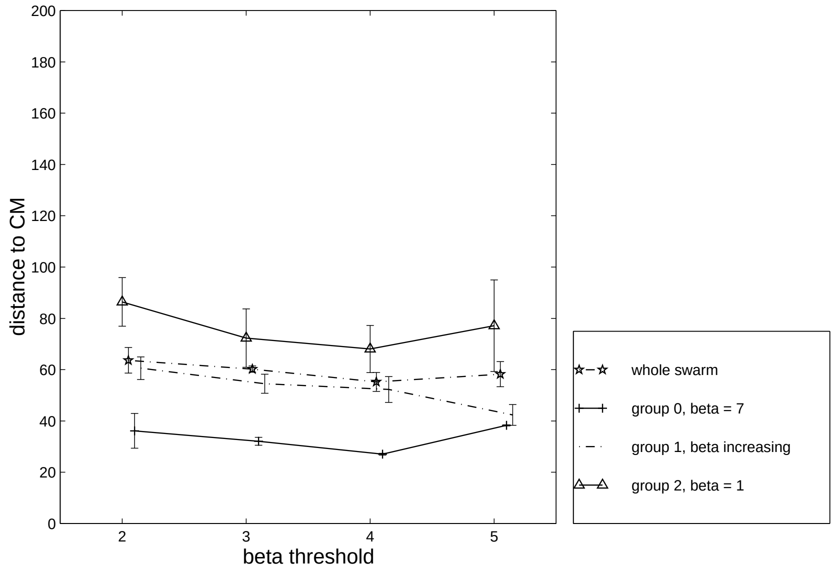


Figure 6.11: 3-group concentric segregation (n=20)

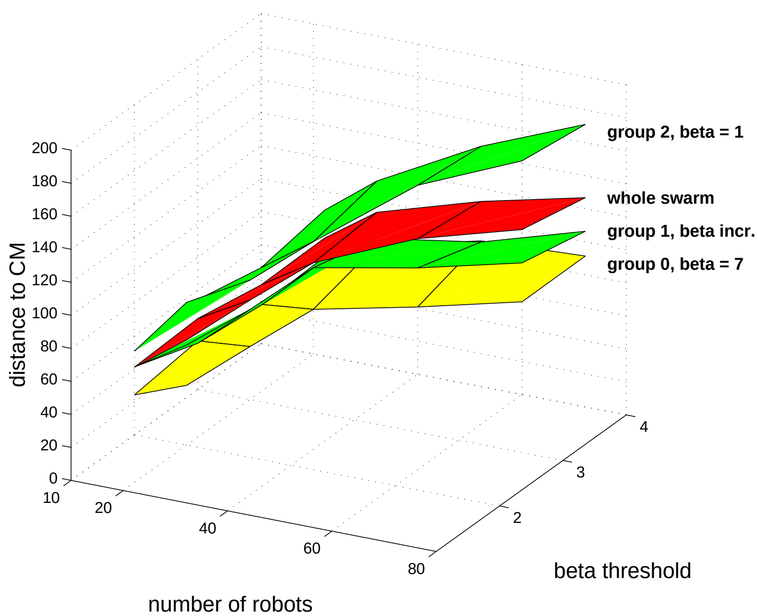


Figure 6.12: 3-group concentric segregation

of mass. Hence we have an example of global shape control through local rules. The only drawback is in the speed of the process which needs long runs to reach the desired equilibrium, the reason being that the chosen robots need to increase their connections one after another. Nevertheless it is a powerful behaviour of the β-algorithm that presents good scalability and robustness to noise.

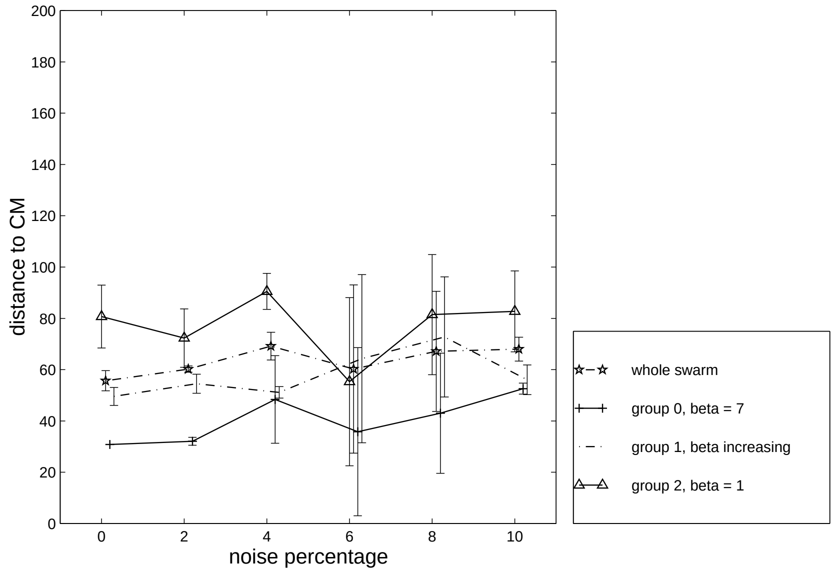


Figure 6.13: 3-group concentric segregation (n=20)

For the time being, the simulated robots have differing thresholds, fixed at the beginning of the run, but an alternative approach would be to find a distributed way of fixing threshold, possibly by choosing robots that are already suitable for that group; and therefore decreasing the time of completion. One possibility could be to use the insider/outsider estimation of the spatial selection algorithm described in section 4.5.

6.1.4 simulation results for radial segregation

The results for the radial segregation experiments of a 3-group partition, with group preferences topology A are depicted in figures 6.15 to 6.17. Figures 6.18 to 6.20 show the results for group preferences topology B (see section 6.1.1).

Figure 6.15 shows the behaviour of the mean minimum distances between the different groups as compared with the non-segregated case. Each group shows similar values which are qualitatively different from the non-segregated case, diminishing towards the default with increasing intermediate β value. When considering a more general parameter space, the behaviour is similar for all swarm sizes (figure 6.16, for readability only the non-segregated case and one of the three group value are shown). The time needed for larger swarms to reach the desired equilibrium explains the better performance of smaller swarms and suggests longer simulation runs are needed to investigate if the performance at the equilibrium is dependent on swarm size. But single long runs (figure 6.6) showed very good performance with large swarms. Figure 6.14 shows a sequence of a 3-group radial segregation process.

Noise, as expected, diminishes the segregation measures towards the non-segregated case, with gentle degradation as the level increases (figure 6.17).

Figure 6.18 shows simulation results for topology B. The mean minimum distances for the pair $\bar{0}-\bar{2}$ are clearly larger than for the pair $\bar{0}-\bar{1}$ or $\bar{1}-\bar{2}$. $\bar{0}-\bar{1}$ and $\bar{1}-\bar{2}$ are neighbouring pairs in the topology, while the group $\bar{0}$ is not a neighbour of group $\bar{2}$. Here again the difference from the non-segregated case levels down along the increase in the intermediate β value, but not for the pair $\bar{0}-\bar{2}$ as the third β value (non-preferred group) stays equal to 1. The variations are high due to limited number of simulation runs.

In the more general picture of the influence of swarm size (figure 6.19 with $\bar{0}-\bar{2}$ distances represented in the top surface, $\bar{1}-\bar{2}$ in the middle and the non-segregated below), a good scalability of the radial segregation algorithm is clearly observable.

The behaviour again shows good degradation to noise with differences gently leveling down. The mean minimum distance for the non-neighbouring pair $\bar{0}-\bar{2}$ always stays above the others (figure 6.20).

In fact the topology of group preferences is transported into the topological structure of the swarm. As topology A presents central symmetry, the measure shows similar values on all pairs. On the other hand topology B assigns a central role to group $\bar{1}$ which can directly be observed in the structure of the swarm. Again with only a slight change in the algorithm and more importantly without increasing the information exchanged, potential control of the global structure of the swarm is demonstrated. The linear nature of the group preferences of topology B indeed leads to the formation of "linear" swarms as can be seen in figure 6.6. But this process needs several million steps to reach equilibrium and is therefore beyond thorough investigation with the available computing power of the laboratory. The slow speed is due to the tendency of each robot to maintain connections for the sake of coherence, which diminishes its mobility within the swarm. In the next section we investigate the potential to use an external cue (the taxis beacon) to differentiate the robots in order to build a linear shape.

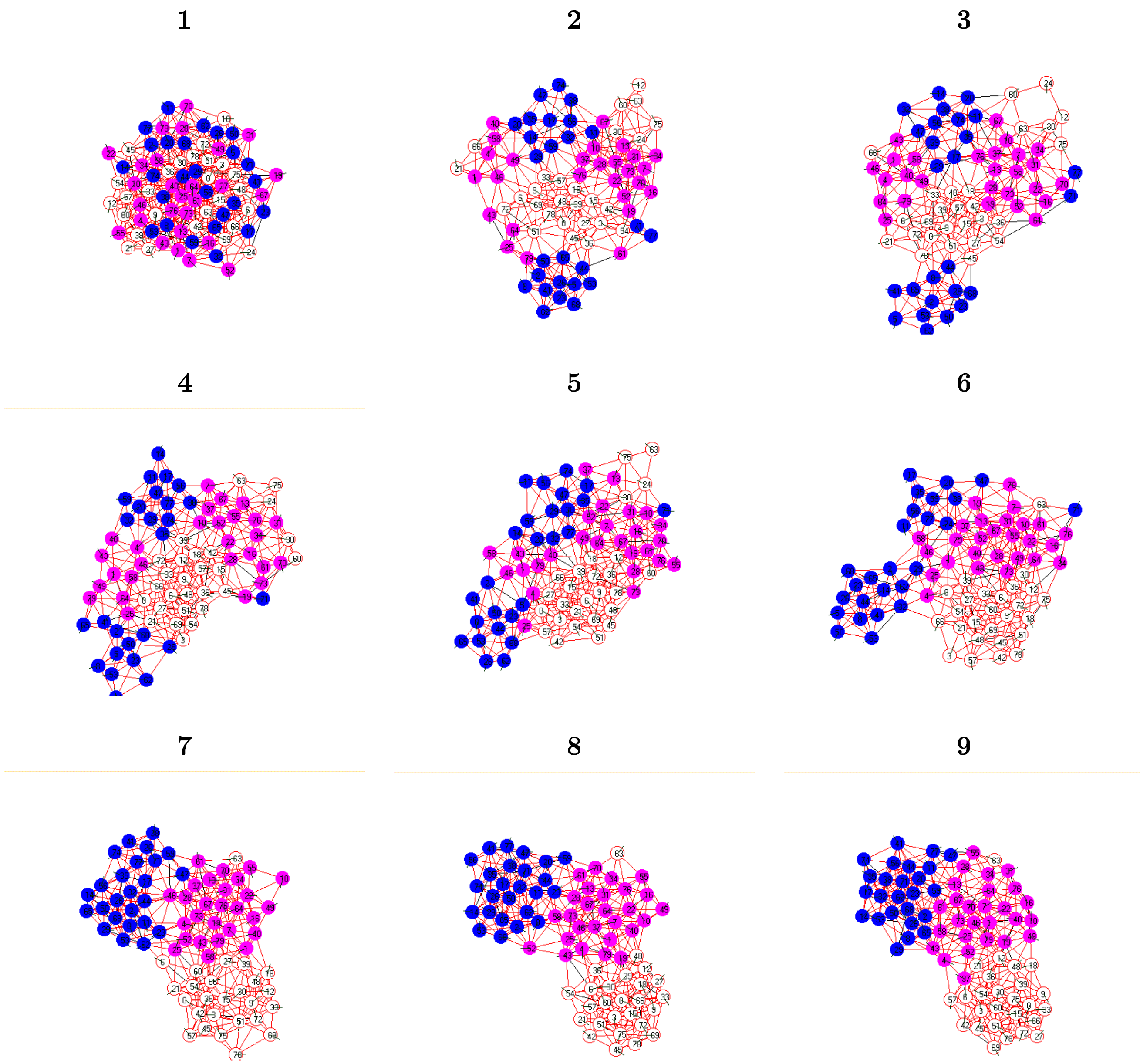


Figure 6.14: sequence of 3-group radial segregation process (topology B)

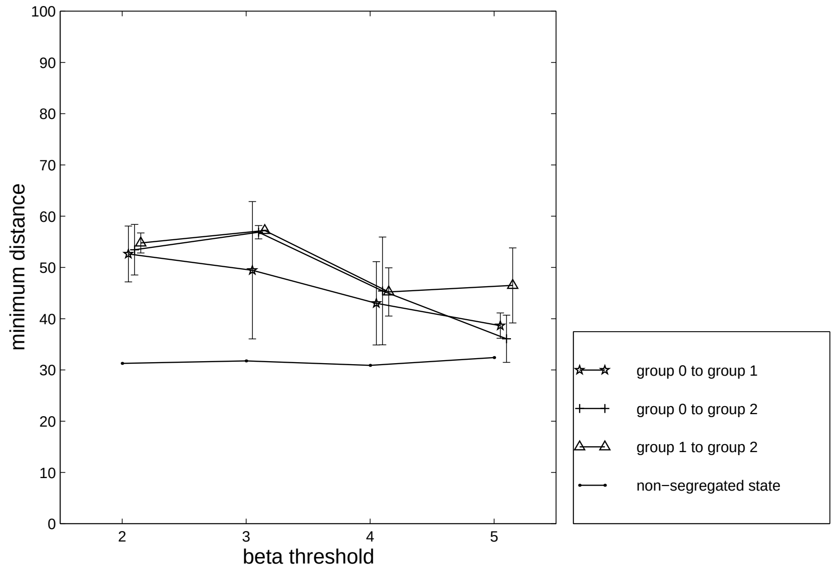


Figure 6.15: 3-group radial segregation topology A (n=60)

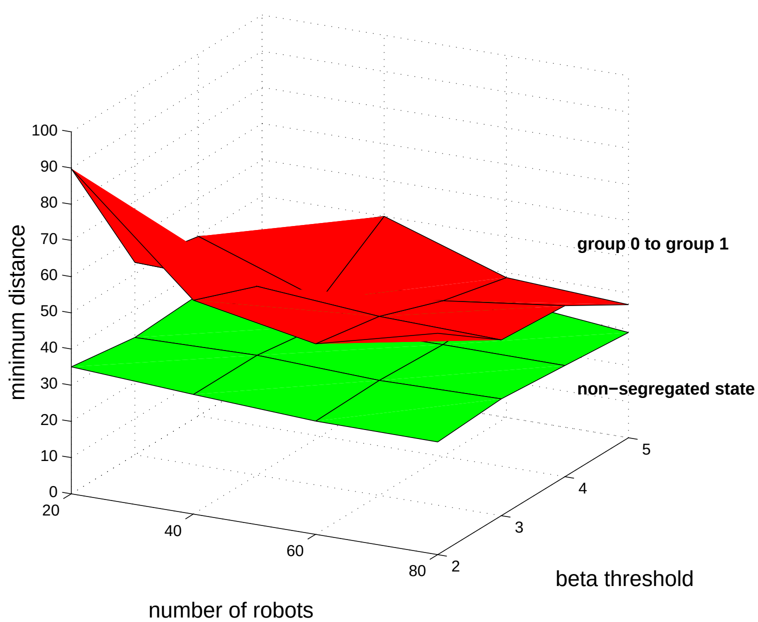


Figure 6.16: 3-group radial segregation topology A

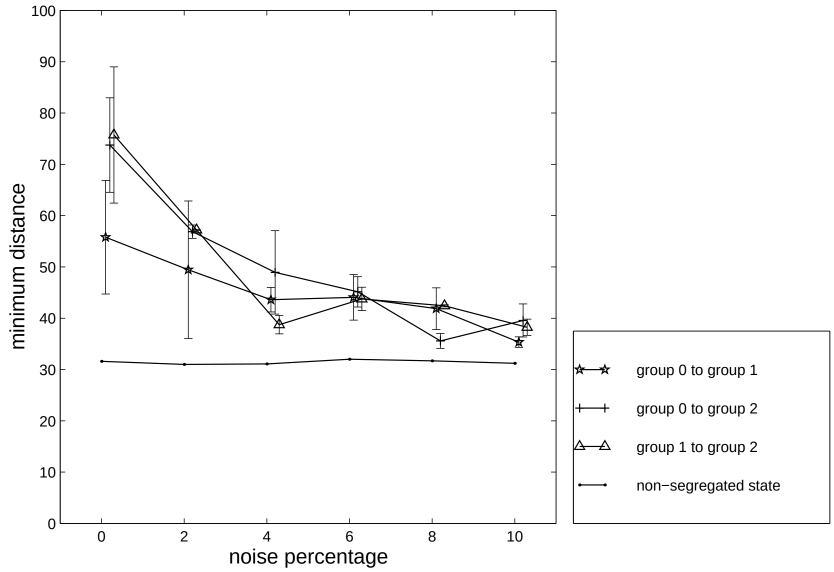


Figure 6.17: 3-group radial segregation topology A (n=60)

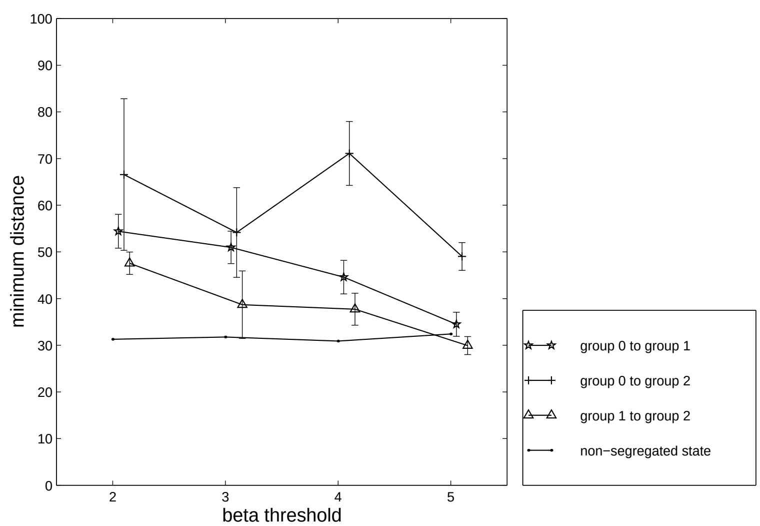


Figure 6.18: 3-group radial segregation topology B (n=60)

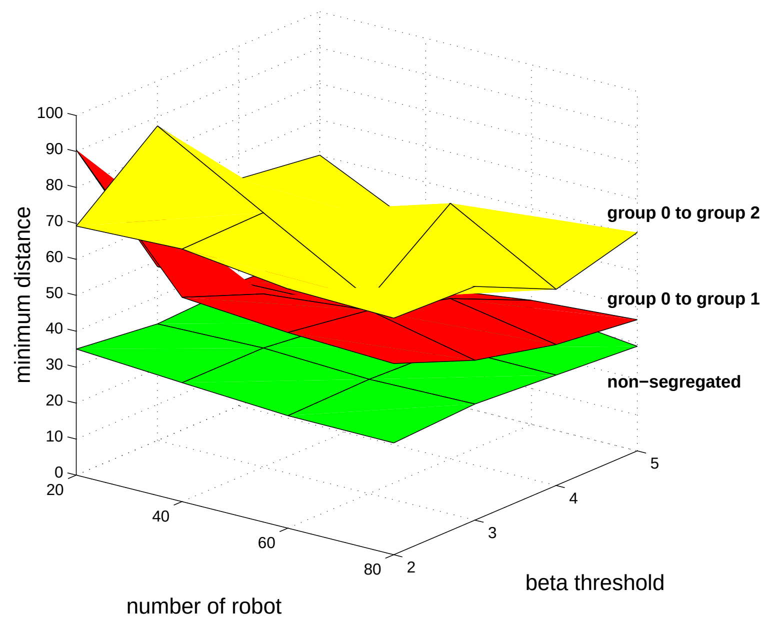


Figure 6.19: 3-group radial segregation topology B

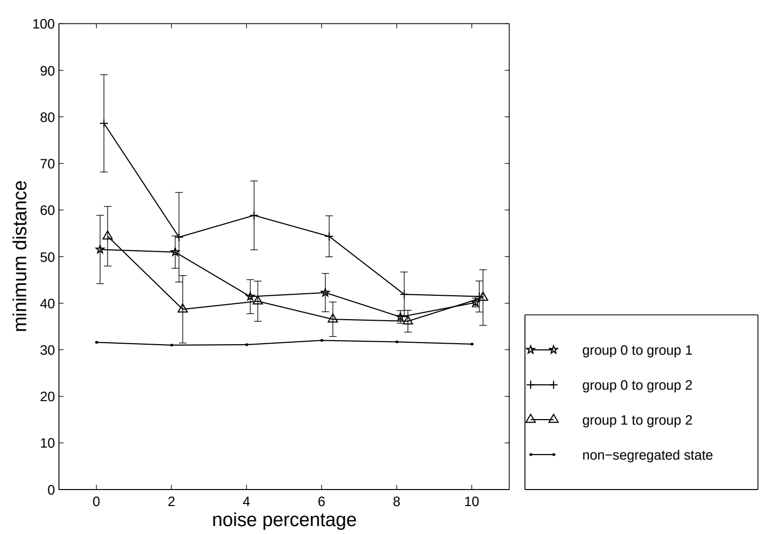


Figure 6.20: 3-group radial segregation topology B (n=60)

6.2 Axis Formation

One of the most striking achievements of morphogenesis in the development of the embryo is the transformation of the initial circular symmetry of the zygote into an axial symmetry that leads to head-tail and dorso-ventral differentiation. Considered in perfect geometrical spaces, this transformation states the conceptual problem of choosing the direction to grow the axis. It is referred to as the *break of symmetry.* If there is no preferred direction, there is no reason why a particular cell starts to divide at a higher rate in order to create the axial pattern. The temptation is high to see, in this phenomenon, the proof for a predetermination of the cell. But careful experiments have shown that the choice of a preferred direction can also be dynamically reorganised to fit particular environmental conditions. For instance the axis differentiation of a plant is greatly influenced by gravity. The germination of a seed turned upside down is of course not making the plant grow inside the terrain. Further experimental results on the influence of external cues on symmetry breaking in the embryo of the fruit fly, in the form of maternal diffusing morphogens, can be found in [Nusslein-Volhard, 1996].

Things could be simple and the problem settled: symmetry breaking does need environmental cues. But unfortunately experiments have gone further and investigation of the growth of plants in microgravity have shown that the presence of gravity is not necessary for their development. Also attempts to identify the morphogens responsible for the break of symmetry in the mammal embryo have until now been inconclusive [Gilbert, 2002]. These experiments suggest that the mechanisms involved in the break of symmetry make use of environmental cues decisive for the survival of the individual, but that these mechanisms can also cope with the absence of these cues.

This ability is of primary importance for electromechanical devices such as robots. Indeed these devices normally show a very poor performance when the environment they have been designed for is changed. In this section we investigate the possibility of using the light occluding mechanism presented in chapter 5 to provide the environmental cue for axis formation. Cueless axis formation will be investigated in section 6.3.

6.2.1 description

To achieve axis formation, the idea is to use the same algorithm that was used for the taxis task (see pseudo code 5.4) introducing more differentiation between the "red" robots and the others. In this algorithm the "red" robots have a priority in reaction in the sense that any robot losing a connection to a "red" robot has to react whatever the number of shared connections. The influence of this differentiation is described in section 5.2.

It is proposed that being "red" or not also has an influence on the speed of the robots: both cases of the "red" robots being faster or slower than the others will be investigated. This variant of the β-algorithm will be referred to as the *axis-β-algorithm*.

This very slight differentiation has actually a huge impact on the overall shape of the swarm. Indeed in the case of the "red" robots being faster, the group of these robots move fast while the others are slowed down. As a result the "red" group elongates the swarm in the direction of the beacon. In the other case where the group of the non-"red" robots moves faster, the robots that become illuminated suddenly slow down and are overrun by fast robots that soon become illuminated. The result is a swarm growing in both directions perpendicular to the direction of the beacon. Figures 6.21 and 6.22 show examples of horizontal and vertical axis. Figure 6.23 on page 182 shows a sequence of horizontal axis formation.

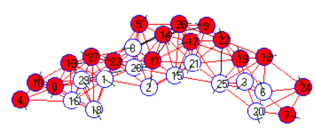


Figure 6.21: horizontal axis formation with speed ratio = 10

6.2.2 measures

The measures used to test the potential of the *beta*-algorithm to achieve coherence are as follows (see chapter 3):

- *number of successful runs*, a successful run being a run that ends with a connected swarm.

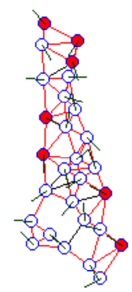


Figure 6.22: vertical axis formation with speed ratio $= -10$

- *edge-connectivity*, which represent the minimal number of edges that have to be removed to disconnect the network.
- *odometry* which records the proportion of time spent in the different states (forward, turn, backwards and stop) over the whole run.
- *vertical/horizontal axial ratio* which gives the performance measure of the axis formation algorithm, as the beacon is situated north of the starting area. It is computed as the added squared distances to the horizontal line through the center of mass, divided by the added squared distances to the vertical line. Hence a value above 1 indicates that the swarm is more vertical than horizontal.

The speed ratio is defined as follow: if it is positive, the "red" robots move with a speed equal to $\frac{1}{ratio}$ while the others move with a speed equal to one. If the ratio is negative the "red" robots move with a speed of one and the others with a speed equal to $\frac{1}{|ratio|}$. The value 0 corresponds to no speed differentiation between the "red" robots and the others.

Each run lasted 500,000 time steps with a cadence value of 100. Hence, each robot sent 5000 messages. Each value was recorded once every 5000 steps and has been averaged over the second half of the run. For a number of values of swarm size, β and noise, 5 runs have been performed and the result depicted is the mean value over these runs with its standard deviation. When the purpose

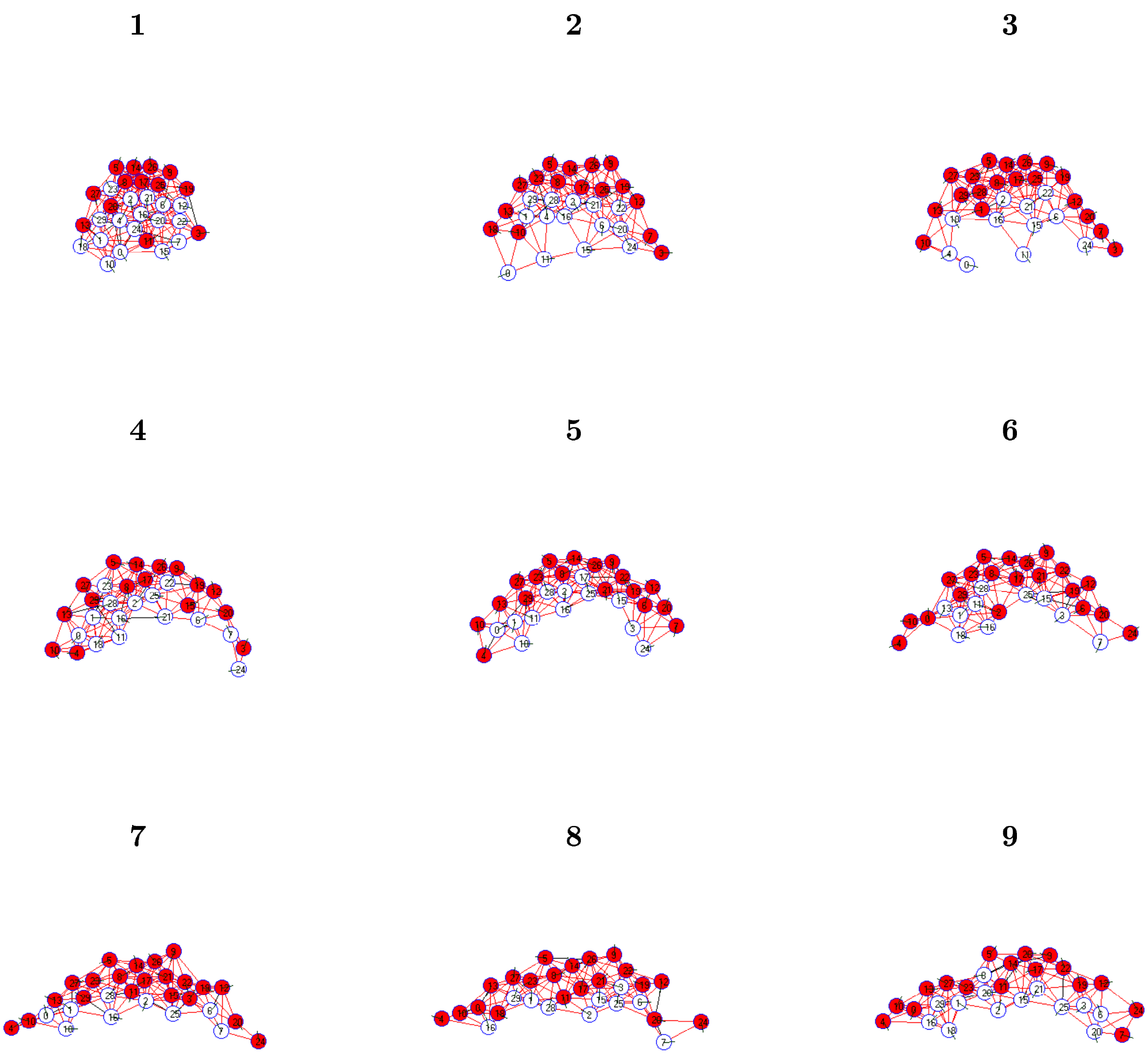


Figure 6.23: horizontal axis formation sequence with speed ratio = 10

size	20 or 60
cadence	100
random noise	2%
β	2
ratio	-10 or 10
steps	500,000
runs	5

Figure 6.24: general parameter values for axis formation algorithm

of the investigation is not to vary them, the chosen values for the remaining parameters are shown in table 6.24.

6.2.3 simulation results

swarm size and β threshold variation

Firstly the influences of increasing swarm size and increasing β parameter are investigated and the results depicted in figures 6.25 to 6.31.

The influence on the vertical/horizontal ratio can be seen in figure 6.25. The top surface corresponds to a speed ratio value of -10; the lower surface to a value of 10. Clearly a separation is observable between the behaviours with different speed ratios. Also one notes a drop in performance with increasing swarm size and increasing β threshold. The first drop is mainly due to the restricted length of the run, as the self-organisation of the swarm takes more and more time with increasing swarm sizes, especially where the speed of a group of robots is slowed. The other drop corresponds to the fact that the algorithm needs a high differentiation between the "red" robots and the others. Increasing the β threshold reduces this difference.

Taking a look at connectivity (figures 6.26 and 6.27), the characteristic increase corresponding to increasing β values is clearly present. Also a drop in connectivity for the speed ratio value 10 when swarm size increases is observable, which corresponds to a drop in the proportion of finished runs (see figure 6.28). This suggests that the swarm becomes more brittle with increasing size.

Figures 6.29 and 6.30 depict the odometry for speed ratio values equal to 10 and -10 which

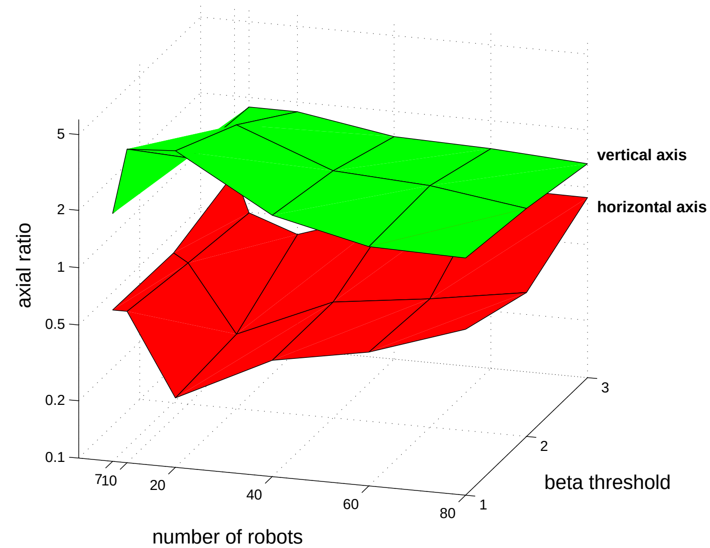


Figure 6.25: horizontal/vertical ratio for speed ratio = 10 and −10

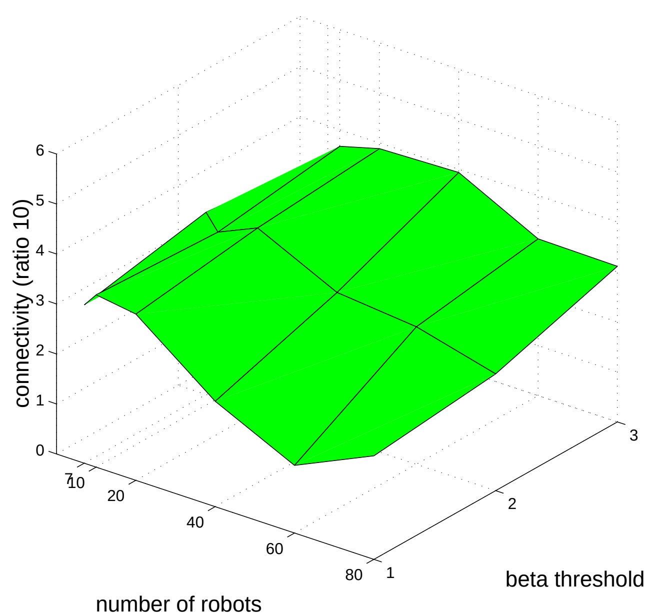


Figure 6.26: edge-connectivity for speed ratio = 10

presents a net difference of the proportions in different states between the two cases. The value 10 has a higher proportion of "turning" state which is related to the brittleness noted in connectivity.

Figure 6.31 shows the behaviour of the normalised area with both speed ratio values. The upper surface corresponds to the speed ratio value of 10 and the lower surface to the value −10. Despite

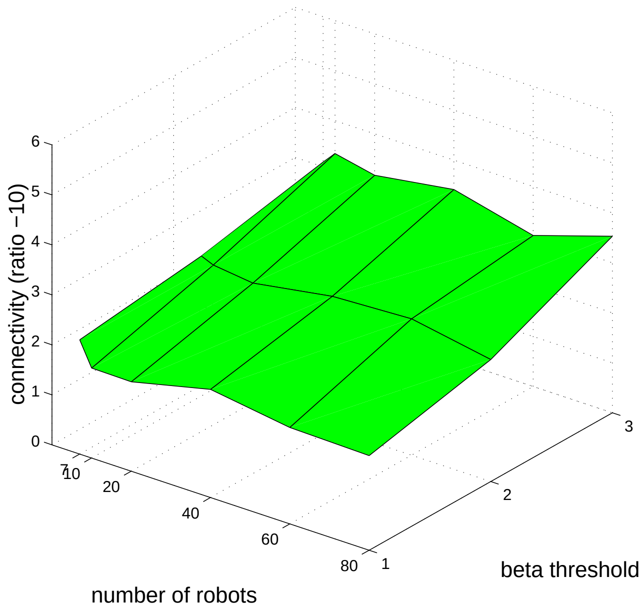


Figure 6.27: edge-connectivity for speed ratio = −10

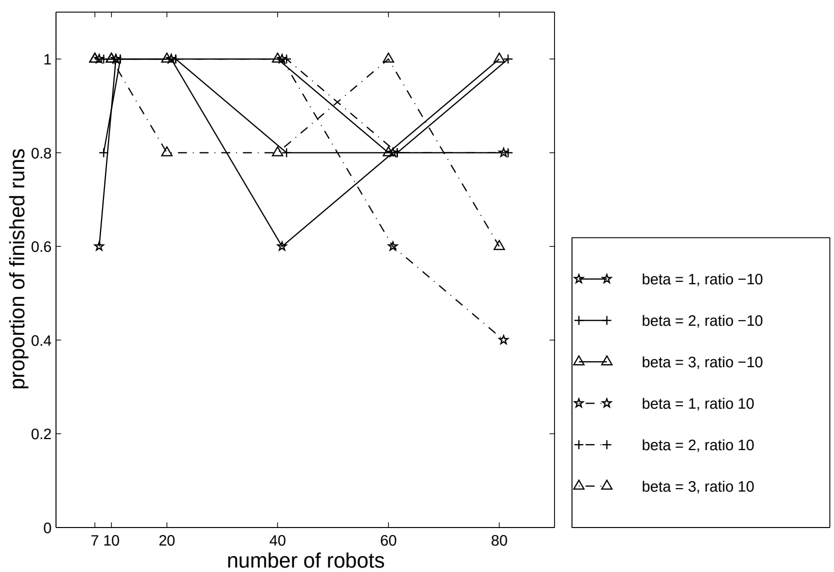


Figure 6.28: proportion of successful runs

a huge difference in the axial ratio, the normalised area is very similar for both values. What is especially remarkable is the absence of decrease in area for increasing beta values. Taking a look at figure 4.46, it is noticeable that the minimum value for each swarm size is not reached. Instead the swarm keeps the area in a middle value, even with varying ratio.

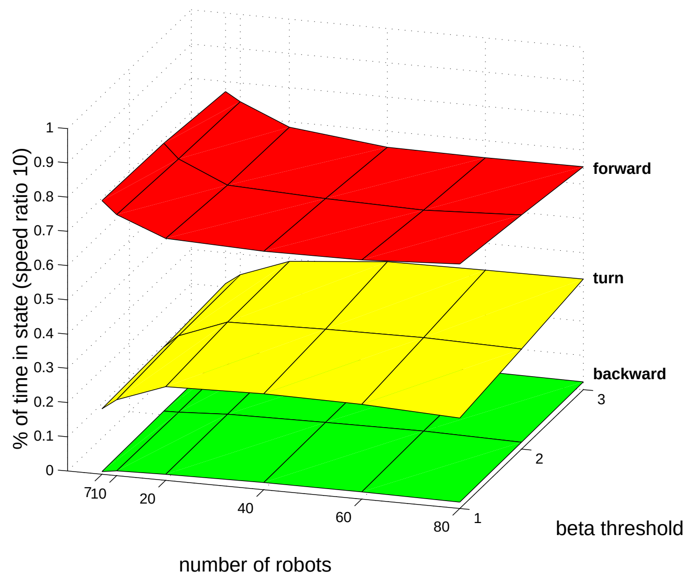


Figure 6.29: odometry for speed ratio = 10

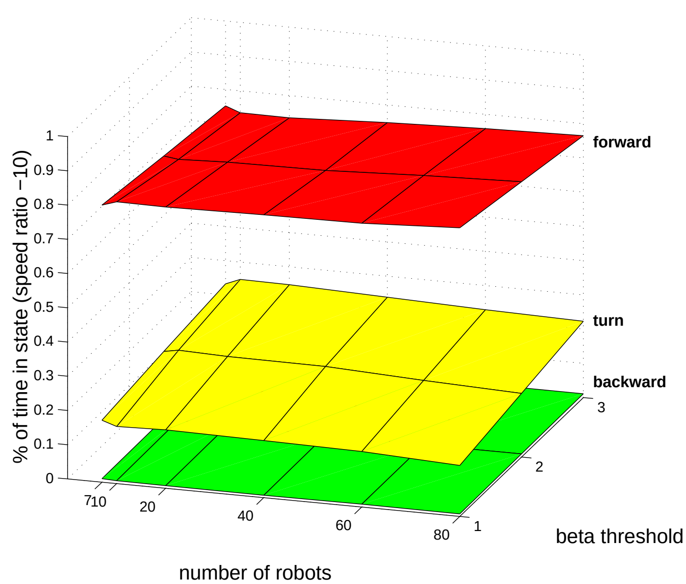


Figure 6.30: odometry for speed ratio = −10

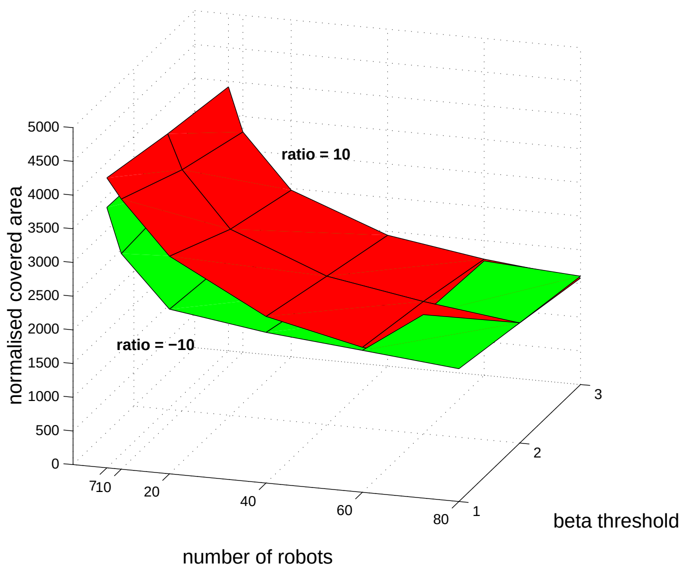


Figure 6.31: normalised area for speed ratio = 10 and -10

speed ratio

Next we study the change of behaviour of the swarm with varying speed ratio values. Figures 6.32 to 6.35 present the results. The behaviour of the axial ratio with increasing speed ratio values shows a remarkable instance of *phase transition* (figure 6.32). Indeed, for both swarm sizes $N = 20$ or 60, though to different extent, the axial ratios for negative and positive speed ratio values are qualitatively extremely different. The behaviour of the swarm stays at a relatively high ratio value for negative values and sharply decreases to reach the state of the swarm for positive values. Whereas the neutral speed ratio value stands as a middle point between these two different states.

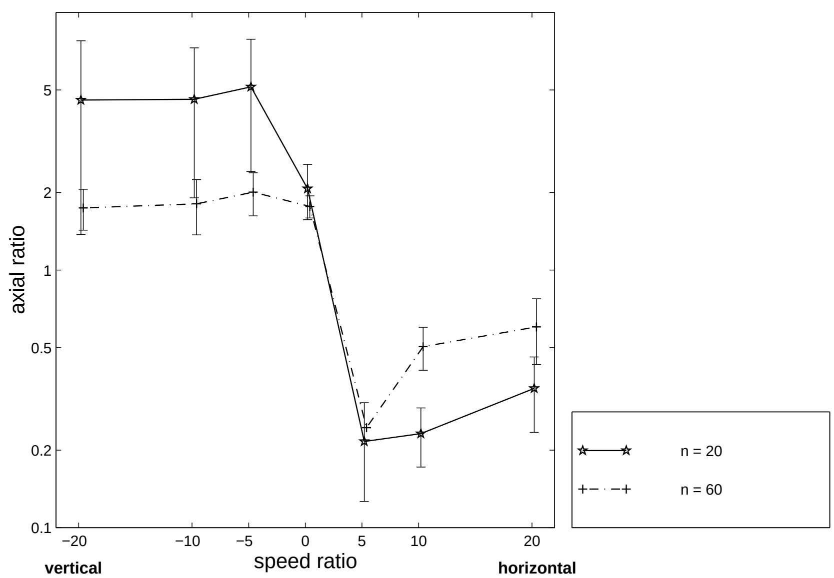


Figure 6.32: axial ratio for increasing speed ratio values

Phase transition is a pervasive phenomena in complex systems, and was first used by Erdös and Renyi to mathematically define the appearance, in randomly growing graphs, of a giant connected component [Erdös and Renyi, 1960]. This phenomena is at the core of the concept of *Small Worlds* [Watts, 1999] and is also the basis of Kauffman's theory for the emergence of Life [Kauffman, 1989]. An example of an investigation in phase transition phenomena related to robotic communication is presented in [Brodie, 2000]. It explores, with the help of evolutionary computing, the appearance of communication in a group of robots engaged in a foraging task.

The plot for edge-connectivity shows that connectivity stays relatively constant for negative speed ratio values for both swarm sizes (figure 6.33). On the other hand, positive values corresponding to horizontal swarms show differences in the connectivity trend with increasing speed ratio values: the smaller swarm ($N = 20$) has increasing connectivity while in the larger swarm ($N = 60$) connectivity remains more or less constant. This difference is also seen in the proportion of good runs (figure 6.34). A reduction in connectivity for larger swarms in the case of the speed ratio value of 10 has already been noted, and it is clear that this is a general feature of positive speed ratio values. Taking a look at the odometry plot in figure 6.35 we see a comparable state transition between negative and positive speed ratio values.

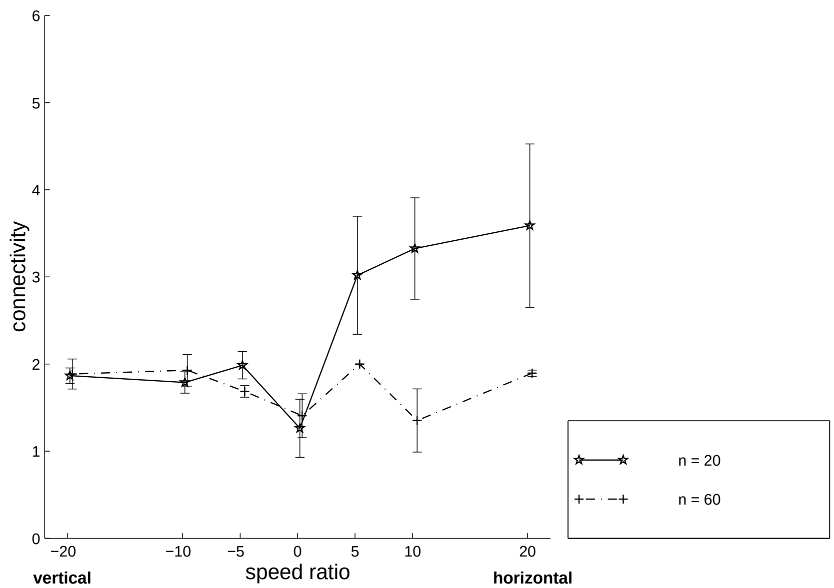


Figure 6.33: edge-connectivity for increasing speed ratio values

noise

Finally we investigate the influence of increasing levels of noise on the performance of the axis formation algorithm. The results are seen in figures 6.36 to 6.39. In this case, a parallel increase in the noise level is used: the influence of communication is not decoupled from the other source of noise.

As expected we note a decrease in performance with a leveling of the qualitative difference between positive and negative speed ratio values (figure 6.36).

Edge-connectivity (figure 6.37), is largely conserved with increasing noise. The difference between small and large swarms for speed ratio value of 10 is observable as well. We should remind ourselves

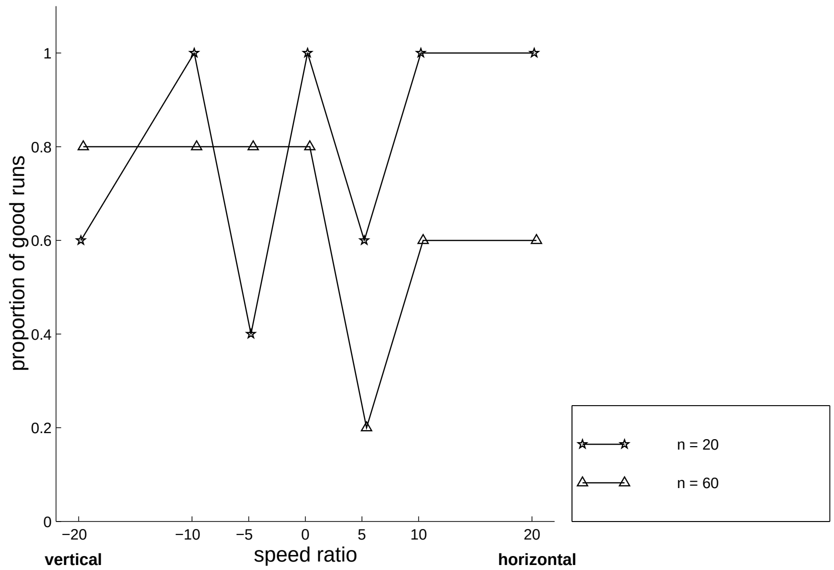


Figure 6.34: proportion of good runs for increasing speed ratio values

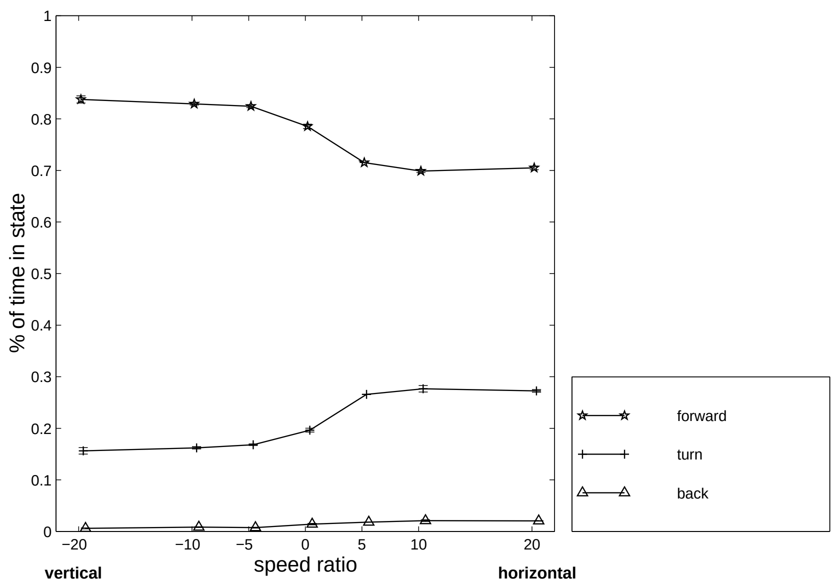


Figure 6.35: odometry for increasing speed ratio values

here that all values are recorded only for successful runs (except of course for the mean length of the runs). Indeed when the influence of noise on the proportion of successful runs is observed, a sensible decrease is seen (figure 6.38).

The plot of the odometry (figure 6.39) shows a characteristic decrease of the "forward" state proportion to the point where the swarms spends half of its time turning and half going forward.

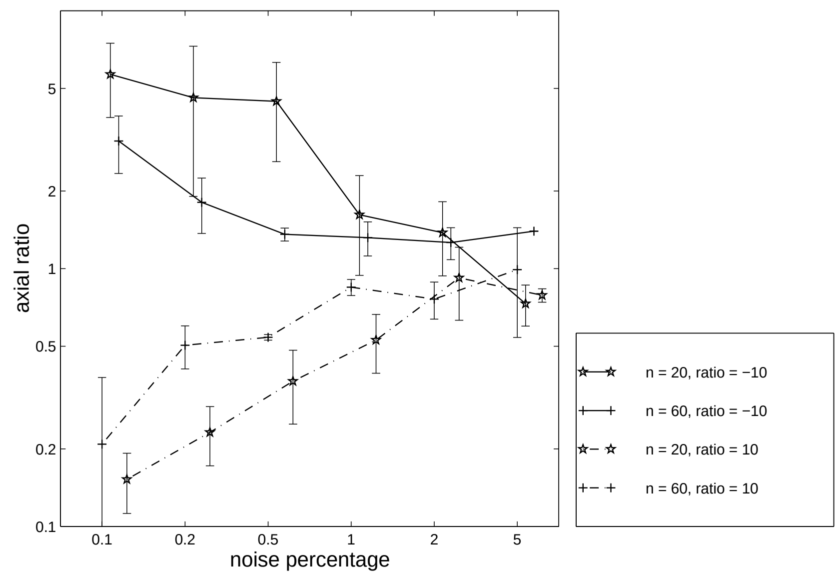


Figure 6.36: axial ratio for increasing noise level

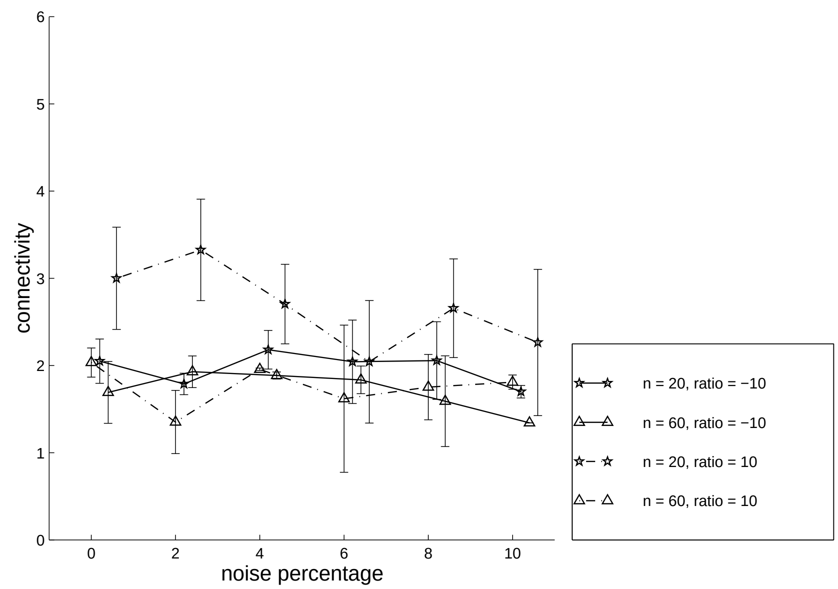


Figure 6.37: edge-connectivity for increasing noise level

The same influence was noted when studying the initial β-algorithm (see figure 4.22, page 100).

Considering that the algorithm used for axis formation is only slightly modified from the algorithm for taxis behaviour in chapter 5, and given that the decoupling of noise sources has not been investigated, the results in this section give good confidence that communication noise is largely responsible for the drop in performance. As already noted, the levels of noise in communication simulated here

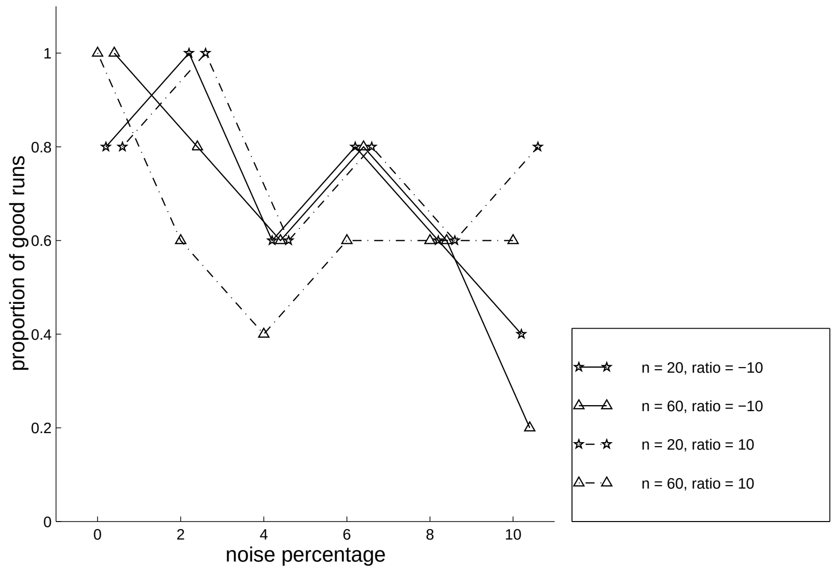


Figure 6.38: proportion of good runs for increasing noise level

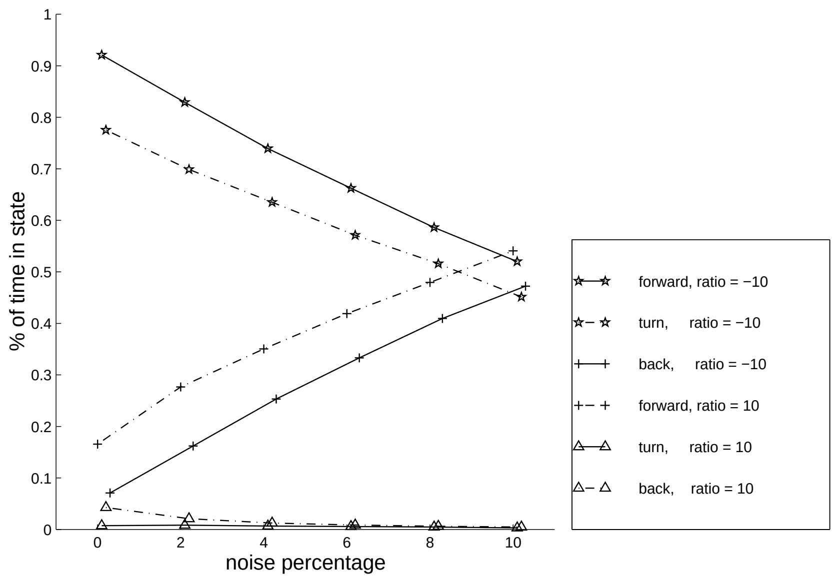


Figure 6.39: odometry for increasing noise level

(up to 10% of message loss) are far above what would be found in practice in the communication industry.

6.3 Evolution

The previous section has shown that, with a beacon and the light extended β-algorithm, we can induce the swarm to adopt a linear global shape. Also mentioned in section 6.1 was that topology B of group preferences leads to linear shape equilibrium, although only after a huge number of steps. This section presents an extension of the β algorithm, referred to as the *evolution-β-algorithm*, to investigate, through artificial evolution, the possibility (in the absence of a beacon) to speed up the self-organising process. The aim is a self-induced and self-sustained linear global shape for the swarm.

The previous sections of this chapter have made it clear that differentiation in the robots' speed and β threshold is potentially sufficient to allow the linear property to emerge. However, in the case now considered there will be no external cue, thus only an internal mechanism can induce such a differentiation.

In the animal embryo it has been shown that this differentiation is the result of the diffusion of internal and external *morphogens*, with different local concentrations, which trigger the expression of locally differing morphogenetic processes, leading to a break of symmetry [Nusslein-Volhard, 1996]. This crucial role was first imagined by Alan Turing in a seminal paper in which is described a mechanism explaining the formation of patterns through the diffusion of such morphogens [Turing, 1952].

Hence the β-algorithm is extended, following this morphogen diffusion metaphor, by making each robot produce morphogens that diffuse across the network, triggering change in speed and β threshold.

6.3.1 mechanism description

Consider now that each of the robots carries two different elements : a string of integers with value between 0 and 100, the *genotype*, and a set of local concentrations of virtual diffusing chemicals, the *transcription factors* (TFs). The first element carries the "genetic" information, common to all robots, while the second is a set meant to mediate diffusion of genetic information between robots. It is with differing values within this set of transcription factors that the robots differentiate. The genotype directs the production of the TFs while their local concentrations have a direct influence on the genotype production. Considering only one robot, this framework represents a dynamical system that soon reaches equilibrium. However, by their ability to diffuse across the network, the TFs not only have an influence on the concentration of the robot considered but also a local influence on the concentrations of its neighbours.

In tying together concentration levels with *phenotypic* features such as speed and β threshold, the production of the genotype has an influence on the behaviour of each robot. However, the behaviour of each robot creates changes in the topology of the network, thereby changing the dynamics of diffusion. This, in turn, has an influence on TF concentrations and hence on genotypic TF production. The overall mechanism is depicted in figure 6.40.

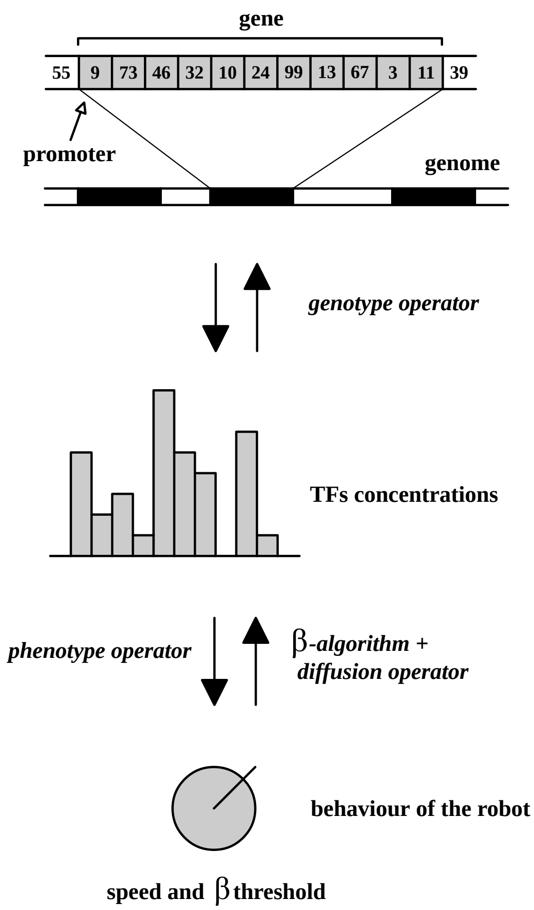


Figure 6.40: genotype-phenotype mechanism

More formally the mechanism can be described as follows:

- the vector of concentrations of transcription factors in robot i is transformed by the genotype operator G and decayed by a constant decaying factor ξ.

$$v_{tf}^i := G(v_{tf}^i) - \xi$$

- speed and β threshold of robot i are modified by the phenotype operator PH, a function of current speed, β threshold and the concentrations of TFs.

$$(s^i, \beta^i) := PH(v^i_{tf}, s^i, \beta^i)$$

Define the total distribution of TFs concentrations, speeds and β thresholds as follows:

$$M_{tf} = (v^i_{tf})_{i=1..R} \qquad v_s = (s^i)_{i=1..R} \qquad v_\beta = (\beta^i)_{i=1..R}$$

- the total distribution of concentrations is updated by the diffusion operator $DIFF$ which is a function of the current concentrations and the topology of the network N.

$$M_{tf} := DIFF(M_{tf}, N)$$

- the topology of the network N is a function of the current state of the network and the total distribution of speed and β threshold.

$$N := F(N, v_s, v_\beta)$$

Unlike operators G, PH and $DIFF$, the function F is not deterministic but emerges from the interactions of all robots, and previous sections have shown the crucial role played by speed and β threshold.

The idea is that such a mechanism, provided with a suitable genotype, creates a dynamic of reinforcement leading to dynamical equilibrium in a selected global shape. In order to find such a suitable genotype, an artificial evolution is run, mating high fitness individuals by crossover and mutation.

With this mechanism, the robots remain homogeneous in the sense that they all carry the same genotype, but they become heterogeneous as the purpose is to trigger differentiation in speed and β threshold.

genotype operator

The genome consists of genes spread on a string of integers which can vary in length. At the end of each listening round the genome is scanned. The beginning of a gene, the *promoter*, is marked by a number below a certain threshold. When a gene is found the next 10 numbers, the *bases*, have the following meaning:

- Bases #1 and #2 are the IDs of the transcription factors that *regulate* the genes.
- Bases #3 and #4 code for the 2-input boolean function directing the action of the two regulating TFs.
- Bases #5 and #6 code for the concentration interval needed for the first regulating TF to be considered as active for this particular gene.
- Bases #7 and #8 code a concentration interval for the second regulating TF.
- Bases #9 and #10 represent the ID of the expressed transcription factor and the concentration expressed respectively.

So, in order for the gene to express its TF, the regulating TFs have to be within their active interval or outside it, depending on whether they have an inhibitory or excitatory role on this gene. This role is determined by the 2-input boolean function coded by bases #3 and #4. This boolean function is chosen to be *canalysing* to reduce search space. A canalysing boolean function is a function where the output is totally determined by one of the inputs taking a specific value. It has been shown that nature makes extensive use of such canalysing functions [Kauffman, 1989, Kauffman and Goodwin, 1994] (see section 2.6.1).

The genome is then scanned further for possible other genes. The concentrations are updated at the end of the scanning phase and normalised such that their sum always remains within the simplex.

$$i.e. \qquad \sum_{i=1..N_{tf}} v_{tf,i} = 1.$$

The great advantage of this approach for genome coding is the absence of a set number of genes; there is only an upper limit because of memory limitations, which is set to 100 genes.

phenotype operator

To translate the genomic information enclosed in the concentrations of transcription factors, the first six transcription factors are arbitrarily chosen as *structural* TFs that can change the behaviour of the robot. As speed and β threshold have been shown to be sufficient to induce specific global shape equilibrium, the first two TFs arbitrate increase and decrease in speed, while the following two change the β threshold. The remaining two make the robot change to the "red" state (Chapter 5). This last ability allows for a sudden change of β threshold from the current value to infinity.

More formally, robot i speed is updated as follows:

$$s_{n+1}^i = \begin{cases} s_n^i - \sigma & \text{if } tf_0^i - tf_1^i \leq -\lambda \\ \text{no change} & \text{if } -\lambda < tf_0^i - tf_1^i < \lambda \\ s_n^i + \sigma & \text{if } tf_0^i - tf_1^i \geq \lambda \end{cases}$$

The β threshold and the transition to "red" state are updated analogously. Then the robot behaves following the β-algorithm with the "red" state extension as described in chapter 5.

diffusion operator

The information exchange of the β-algorithm is increased to include diffusion of transcription factors. This diffusion is defined such that some transcription factors have only an internal function whereas others diffuse to neighbouring robots: only TFs whose concentrations exceed the diffusion threshold δ contribute to diffusion.

The contribution to diffusion γ is evenly distributed between the N_i neighbours. This is to introduce difference in the TF concentrations, as the network topology is the only source of heterogeneity that can be exploited to break the symmetry.

Formally, for each transcription factor, a robot i TF concentration is updated as follows:

$$tf_{n+1}^i = \begin{cases} \text{no change} & \text{if } tf_n^i \leq \delta \\ tf_n^i(1-\gamma) & \text{if } tf_n^i > \delta \end{cases}$$

and a neighbour j receives

$$tf_{n+1}^j = tf_n^j + \frac{\gamma}{N_i} tf_n^i$$

All neighbours contributions are summed at the end of the listening round and the new concentrations updated and normalised. That is:

$$tf_{n+1}^i = \begin{cases} tf_n^i + \gamma \sum_{j \quad with \quad R_j > \delta} \frac{1}{N_j} tf_n^j & \text{if } tf_n^i \leq \delta \\ tf_n^i(1-\gamma) + \gamma \sum_{j \quad with \quad R_j > \delta} \frac{1}{N_j} tf_n^j & \text{if } tf_n^i > \delta \end{cases}$$

It has to be stressed that this diffusion operator is not a model of real diffusion phenomena but rather an adaptation to fit into the network framework without any additional exchange of information. It is worth reminding the reader that the β-algorithm does not rely on a concerted exchange of information. Instead information is sent regardless of the hypothetical presence of neighbours, and robots cannot request information from a neighbour. As a result the diffusion mechanism has to function without prior knowledge of the neighbouring concentrations, which restricts the possibility of accurate modeling. In addition to diffusion, the concentrations of the TF decay over time by a constant decaying factor ξ.

The mechanism presented here is similar to that presented by Bongard, but his case assumes and considers an "organism" that is very unlikely to be physically implemented [Bongard, 2002]. On the other hand Taylor presents the result of an evolutionary model that is from the start aimed towards real robot implementation [Taylor, 2004]. The preliminary results are encouraging even if the achieved task is very similar to secondary swarming (see section 2.8.2)

In Bongard's case there is a fixed topology of the TFs sites of expression (cartesian) which has a great influence on the morphologies that can be evolved. In the model presented here the network topology is random and the role of the genotype is to influence this randomness to transform itself into some meaningful global shape.

6.3.2 artificial evolution algorithm

Given the genotype-phenotype mechanism described above, the task is now to find the genome able to arbitrate differentiation within the swarm in order to tend towards specific global shape equilibrium. This is done with the help of a classical genetic algorithm with crossover and mutation [Bäck et al., 1991]. The individuals of a population of genomes are tested one by one and assigned a fitness. Then the best ones are chosen to generate a new population which is tested again. The process repeats for several generations and, provided the problem can be solved by the genotype-phenotype mechanism, eventually leads to the emergence of a locally optimal genome. For a discussion of problems of evolvability and global optimum search strategies, see [Toffoli, 2000].

fitness function

In order to show that a break of symmetry can be self-induced by the interplay of random fluctuations and self-organising behaviour, the specific global shape the swarm has to reach has been chosen to be a linear shape.

Hence the base of the fitness function is the measure of the ratio of the added square distance as described in chapter 3. In the case of robots disconnected from the swarm, these do not contribute to the ratio and the ratio value is reduced. This is to reward for coherence without prematurely condemning non-cohesive genomes.

In order to reward stability of the global shape, the performance measure is recorded several times at the end of the testing run and the fitness is the result of the multiplication of the mean value over the samples by the entropy of the normalised sequence. In other words:

$$\bar{s} = \frac{1}{N_s}\sum_{i=1}^{N_s} s_i \qquad \sigma_i = \frac{s_i}{\sum_{i=1}^{N_s} s_i} \qquad E = -\sum_{i=1}^{N_s} \sigma_i \log \sigma_i$$

$$fit = \bar{s}E$$

where s_i is the sequence of performance measure and N_s its length.

algorithm features

The genetic algorithm proceeds as follows:

- the population is composed of P genomes.
- per generation each genome in the population is tested for N_t runs of 200,000 steps and its fitness is averaged over all runs, including runs of previous generations. The level of noise is set to 2%.
- then some of the B best individuals are selected by pairs to produce $O = P - B$ offspring. They are selected by tournament wheel with probability proportional to their fitness.
- the pairs are mated using a *crossover* operation, which consists in splitting both genomes and forming the new genome by joining two pieces from different parents together. A constant probability of *mutation* is also included, with the replacement of the base value by a new random integer between 0 and 100.
- the process is repeated for G generations.

This genetic algorithm can be represented as a $(B + O)$evolutionary strategy in the notation introduced by [Bäck et al., 1991]. For computational tractability and good evolutionary dynamics, the parameters have been chosen to have the values shown in table 6.41.

P	50
B	30
O	20
N_t	2
G	10000

Figure 6.41: parameter values for genetic algorithm

6.3.3 results

In order to study the evolutionary dynamics, the following values are recorded for each generation :

- the fitness of the best individual, the mean fitness and its associated standard deviation.
- the number of genes of the best individual and the mean number of genes across the population.
- the *persistence* for the best genome and the average persistence across the population. Persistence is the difference between the current generation number and the generation number in which the genome considered first appeared.

Persistence measures the turnover of genomes in the population. A high value means that the population on average is rather old, which means that newly created genomes have difficulties in performing better.

Figure 6.42 presents preliminary results of an evolutionary run. Despite a large number of generations (5300), the fitness does not increase. The number of genes and the persistence suggest the presence of two different epochs: after an initial plateau around the value of 40 genes, the number of genes slowly decreases towards another plateau near the value 20. Simultaneously, the mean persistence starts with quite a high value (around 100), with the persistence of the best individual very low, and decreases towards the value 25. These features suggest that despite a non-increasing fitness value, exploration of the search space is nevertheless taking place. The system is thus still in the initial exploratory period. Indeed, no decisive interaction of genes has yet been found.

Figure 6.43 shows the differences in concentrations, speed and β threshold between two robots with best genome of generation 523. The differentiation in concentrations is clearly observable (upper plot); and this difference is translated into β threshold differentiation (lower plot). In this example

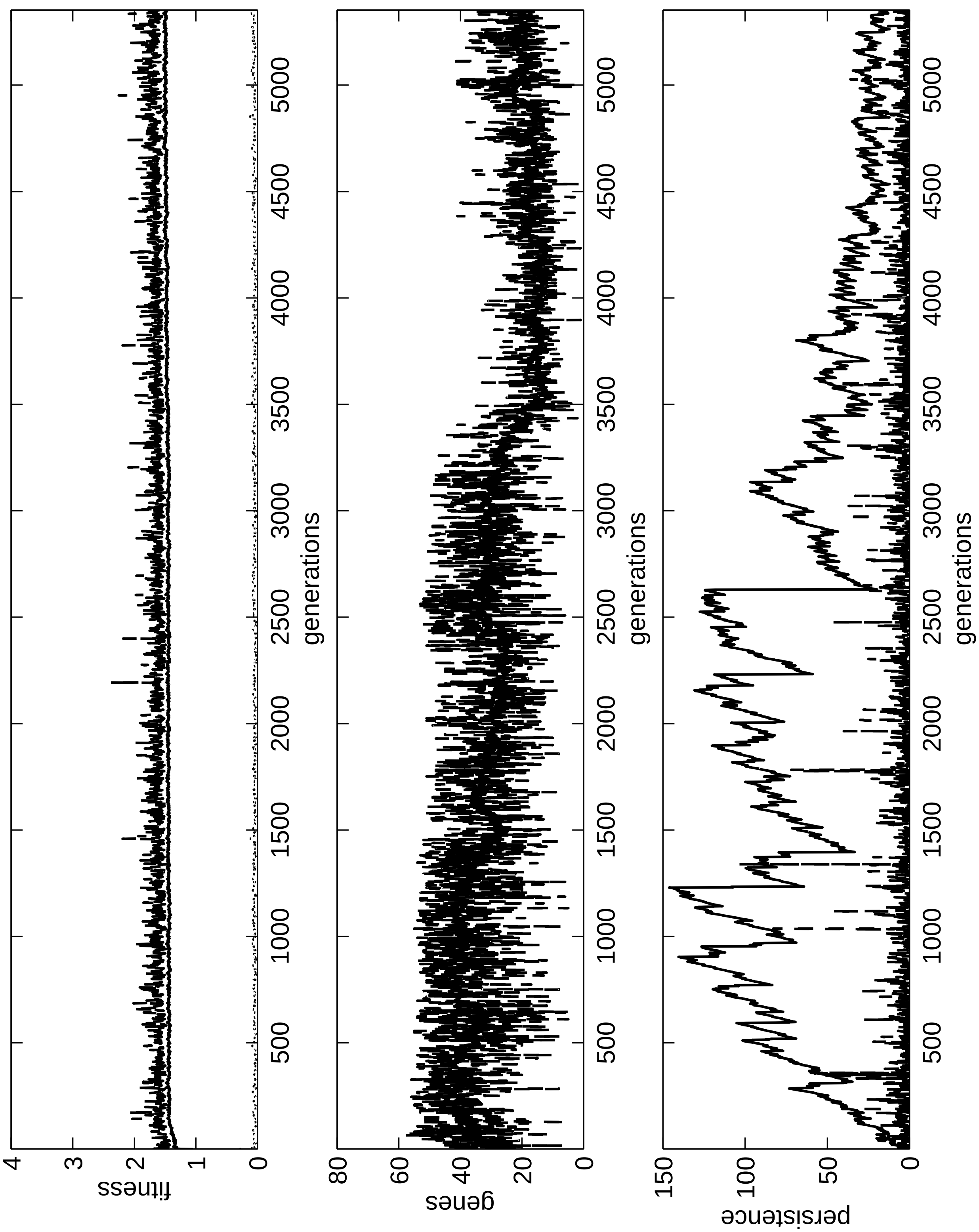


Figure 6.42: evolution results

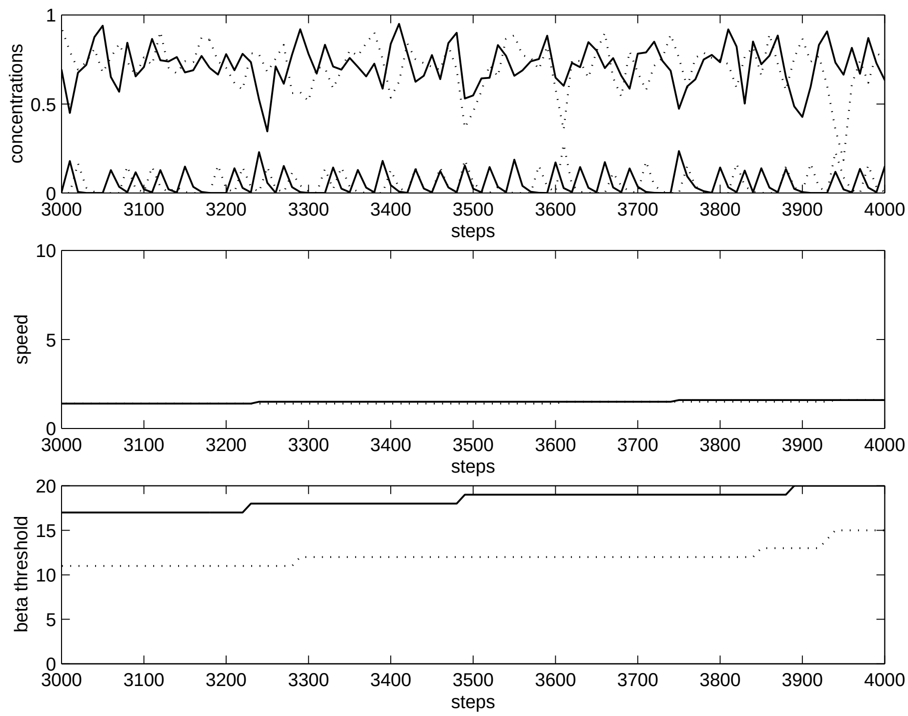


Figure 6.43: differentiation between 2 robots (generation 523)

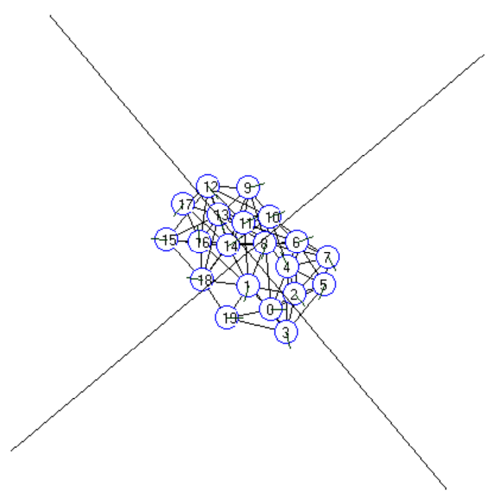


Figure 6.44: typical swarm configuration for best genome of generation 523

the genome contains only a gene coding for the TF allowing increase of the β threshold. A gene coding for the TF necessary for a decrease has not yet appeared. Figure 6.44 shows a swarm running the genome that codes for the differentiation seen in figure 6.43.

These results show that the discovery of acceptable solutions is very slow. Indeed the search space is very large (possibly 100^{1100}) and the available computing power was limited to a SUN UltraSparc 10 machine. With this machine, the presented results needed 10 computing days to complete. The question remains whether the whole system is suffering from ill-definition, that is the genotype-phenotype mechanism is not able to cope with the randomness and dynamicity of the swarm, or whether the exploration needs a larger population, more testing runs or fine-tuning of the parameters. However, with the available computing power this investigation could not be done. Nevertheless the differentiation that was sought is observable (figure 6.43), which suggests that the mechanism may very well be able to display the specific global shape equilibrium.

6.4 Summary of the Chapter

This chapter has presented investigations into the potential for locally controlling the global shape of the swarm by introducing heterogeneities between robots. Firstly these heterogeneities have been predefined and fixed. Secondly they have been induced by a beacon and could therefore dynamically change through the dynamics of the swarm. Finally they have been self-generated by a genotype-phenotype mechanism.

As a consequence of the algorithm, global connectivity of the communication network formed by the swarm of robots is maintained. This could lead to applications in large mobile sensor arrays which require them to adapt their shape to provide appropriate sensing.

concentric and radial segregation

In section 6.1 the possibility to segregate predefined groups of robots has been investigated by firstly giving each group a different β threshold. As a result the groups self-organise concentrically with the group with highest β value in the center, surrounded by the other groups in decreasing β value order.

Secondly the β threshold value has been chosen to be dependent on the group, with the accompanying preference topology. The results show the possibility to radially segregate groups of robots, faithfully reproducing the topology of preference on the global shape of the swarm.

Although showing good scalability and gentle degradation in presence of noise, both processes need long runs to reach an acceptable state and even longer runs to reach the concentric equilibrium.

axis formation

In section 6.2, the beacon sensitive β-algorithm extension of chapter 5 is used to define heterogeneities in speed and β value within the swarm. Speed is shown to be a crucial parameter that arbitrates an impressive phase transition between horizontal and perpendicular axis formation.

The use of a beacon greatly speeds up the process and its steady presence helps in maintaining a very stable equilibrium. Scalability and robustness to noise are conserved.

evolution

The evolution section (section 6.3) presented a mechanism able to control the differentiation in speed and β value, which was specifically designed to be open to artificial evolution. With this mechanism we have investigated the goal of self-generating and self-maintaining a linear global shape. Despite inconclusive results on the ability of the mechanism to reach the desired equilibrium, the mechanism has been shown to be able to induce differentiation within the swarm.

This chapter closes the investigation of the potential of the β-algorithm beyond the initial requirement of coherence that was initiated in section 4.4 of chapter 4. The following chapter will sum up the work accomplished, draw the conclusions that can be extracted from it, discuss the drawbacks of the solutions developed and point to further research directions.

Chapter 7

Conclusion

"...de toute façon t'es foutu."

Salomé

Because there eventually comes a time when research has to be stopped, this chapter will try to sum up the work accomplished during the past three and a half years. It will firstly present the solutions developed, then discuss their drawbacks and advantages. Next it will draw the necessary conclusions and finally point to the potential for further research directions.

7.1 Work Accomplished

The major hypothesis of this thesis is the contention that decentralised control can lead to global coherence of the robot swarm based only upon range-limited communication. Chapter 4 presented the investigation towards this hypothesis and showed, with the help of simulation, that second-order information (information on neighbours' neighbours) is needed to guarantee coherence. It was shown that the algorithm developed (the β-algorithm, see section 4.2.3) is scalable and robust to high levels of noise and that it can be implemented on real robots, despite serious discrepancies between the robotic platform used and the assumptions of the simulation. This algorithm involves only local broadcast of neighbours' information, and can therefore be considered as fully distributed and thus fully scalable.

The minor hypotheses are concerned with the potential for the β-algorithm to allow the swarm to display complex global behaviour such as area coverage control, coherent swarming or global shape control.

Chapter 4 described the potential for area coverage control, in a rather amorphous manner, by tuning the β threshold. This area control is closely linked with the ability of the β threshold to control the edge- and vertex-connectivity of the network. These are global metrics that relate to the resilience of the network to component failure. The ability of the algorithm to distributedly influence global features of the underlying network is of great interest.

A solution for the spatial selection of the robots using the information already available to them for the β-algorithm was also presented. It is able to discriminate between robots that are at the boundary of the swarm and robots that are within the swarm. As the β-algorithm does not rely on positional information, this localisation ability, albeit limited, is quite remarkable. The same idea to use information already available was applied to investigate the possibility to compute local estimates of the area coverage. Although being of limited precision, the estimates still provide reliable bounds on area coverage.

In chapter 5 the extension of the algorithm to include beacon perception led to the development of a truly emergent taxis behaviour, with the additional ability to avoid beacon occluding obstacles coming as a valuable side-effect. Strongly relying on subtle robot interactions, the solution presented cannot make a single robot move towards the beacon. Only the dynamic equilibrium between robots that sense the beacon and robots that don't, makes the swarm as a whole move towards the beacon. This algorithm has the advantage to display taxis while still being able to maintain coherence. Although the purpose of this research is not in investigating biologically plausible solutions, the taxis behaviour is highly reminiscent of bacteria and slime molds such as *Dyctiostelium*.

Finally chapter 6 presented the potential for fixed and dynamical heterogeneities between robots within the swarm to induce the control of the overall shape. Firstly, differential β thresholds across the swarm have shown to be responsible for the concentric dynamical self-organisation of different predefined groups of robots, whereas group dependent β-thresholds presented the ability to segregate different groups within the swarm, according to the topology of the group preferences.

Secondly, the use of a beacon to trigger further dynamical differentiation in speed between the robots was able to achieve dynamical axis formation. Speed is shown to arbitrate a phase transition between co linear and perpendicular axis formation. With this result the potential of the β-algorithm and its variants to exhibit complex behaviours through the tuning of a small set of parameters is definitely confirmed. Again, these behaviours relate to numerous biological examples of morphogenesis, ranging from *Dyctiostelium* to the development of the animal embryo.

These biological analogies are incentives for the development of an evolutionary framework to dynamically tune the crucial parameters according to virtual morphogens diffusing across the network. Despite being deceitful, most likely because of a huge search space and limited computing resources, the algorithm was able to evolve dynamical robot differentiation within the swarm, but failed to achieve the proposed self-induced axis formation task.

All these complex behaviours have been shown to be to some extent scalable and robust to high levels of noise.

7.1.1 algorithms developed

This thesis has presented several algorithms which displayed differing behaviours (see figure 1.1, page 16):

- the preliminary α-algorithm managed to keep the robots together for a while but could not guarantee coherence in the long term because of its fundamental theoretical flaws.

- the main β-algorithm achieved long term coherence, tuning of the network connectivity, area coverage control and the exchange of information needed by this algorithm was demonstrated to be sufficient for crude relative positioning within the swarm (localisation-β-algorithm). Real robot experiments and simulations presented sufficient similarities to support the correctness of the implementation.

- Heterogeneities in β threshold have been shown to induce global group segregation (concentric-β-algorithm and radial-β-algorithm).

- the light beacon extension of the β-algorithm (taxis-β-algorithm) presented, through dynamic differentiation, a truly emergent swarming behaviour that allowed for obstacle avoidance.

- Further differentiation in robot speed within the taxis-β-algorithm led to perpendicular and co linear axis formation (axis-β-algorithm). The investigation of the speed parameter revealed the occurrence of a phase transition phenomena.

- the genotype-phenotype mechanism together with the evolutionary algorithm showed the ability to induce differentiation within the swarm but failed to achieve beaconless axis formation, most likely because of a tremendously huge search space and insufficient computing resources (evolution-β-algorithm).

7.1.2 drawbacks

The essential step from the α to the β-algorithm increased information exchange and sensitivity to the message content. Because the algorithm relies on comparisons of neighbourhood, the compared neighbours have to agree on the IDs they exchange. This could restrict the scalability of the system as each of the robots needs to possess a unique ID. This could be a problem in the case of minimalist robots, but for robots using standard LAN cards, we note that each such card has a unique hardware ID: the MAC address. A possible way to circumvent this restriction was proposed in section 4.6.

One of the drawbacks of the β-algorithm and its extensions is the relative slowness of the self-organising processes. For instance in area coverage control, swarm contraction takes some time to reach equilibrium. This is even more true with the taxis process that needs very long runs to complete. It is clear that applications that require fast completion cannot rely on such algorithms, but applications where reliability is more important can have confidence that the swarm will eventually reach equilibrium (or the beacon). This slowness has greatly impaired the evolution algorithm as long runs were needed to efficiently test the different genomes.

Another drawback was the practical difficulty encountered in limiting the range of the radio device. There is good confidence that the WLAN radios used present good communication range geometry, but this remains to be tested in a (large) real robot experiment. This difficulty may actually be intrinsic to the choice of the radio medium and might restrict the use of such algorithms to large spaces. However, new lower power radio standards such as IEEE 802.15.4 (Zigbee) have tunable range that can be limited to 1 meter [Artaud et al., 2004]. Shifting to another means of communication (sound or chemical for instance) could also circumvent this range difficulty.

7.1.3 advantages

The main advantage of the different algorithms presented is the fact that, in addition to being fully distributed, they all rely on strictly local information exchange. As a result, scalability is almost guaranteed. Only the use of a beacon introduces some performance decrease against increasing swarm size. Except for the ID sensitivity discussed in the previous section, the β-algorithm presents perfect scalability, thanks to the limited communication range.

Although being potentially a drawback, the dynamicity of the swarm is the reason for the swarm's potential in complex behaviours. Not only crucial to the basic algorithm, constant movement (and randomness) is needed to give the robots sufficient freedom to self-organise. As a result the behaviours are dynamical equilibria towards which the self-organising swarm proceeds. Built on such consideration, an evolution framework presented, if it eventually copes with the randomness of the swarm, it may produce genomes coding for the behaviour leading to any shape equilibrium, provided a fitness function is available.

The use of a radio communication device as a virtual bound between the robots leaves freedom in the design of the robotic individuals implementing the different algorithms. Using legged or flying robots only changes the possible strategy to implement the necessary U-turn.

These advantages suggest that the potential of the presented family of algorithms is not restricted to the robotic platform used. In fact, we believe that they represent an algorithm design framework to be applied to the field of swarm robotics in general.

7.2 Hypotheses Demonstrated

From the previous chapters and the discussion above it can be declared that the hypotheses have been proved, although only partially in some cases. Indeed the methodology presented in chapter 3 requires the confirmation of all simulation results by a real robot experiment, at least for a reasonable swarm size. This has been possible only for the most simple behaviours possible with the β-algorithm, as the more complex behaviours require a very long term coherent swarm; this was not possible, largely because of the practical choice of device used to simulate locality. But the success of the implementation, in spite of huge differences between the real robots and the simulation, give good confidence that once a solution to the practical problems can be found, verification of the full set of behaviours will only be a question of time.

This stated, the complexity and richness of the behaviours displayed by robots fitted with such a simple ability as the range-limited broadcast of a list of neighbours, is undeniably impressive and shall certainly lead to further research. It is clear that this research has opened many directions that could be extended much further, some of which represent whole new research areas. This could be seen as a weakness, but the purpose has been from the start to explore the potential of range-limited communication for emergent behaviour and we believe that the present thesis has discovered and explored many interesting features of this approach.

Throughout this thesis the constructionist approach has permitted us to imagine and conduct experiments that, we believe, have increased understanding of the capabilities of such complex robotic systems. It is doubtful that the algorithms that are at the source of the beautiful behaviours presented in the previous chapters could have been invented following a pure analytical approach. Actually we believe that this thesis provides an example of the power of the constructionist approach and that only this approach could yield such results. Now, a better understanding of the reasons for such capabilities is needed and this is where analytical study is required.

7.3 Further Research Directions

Because the initial choice of this study was to demonstrate the potential of the range-limited communication approach, the mechanisms revealed have not been exhaustively investigated. Many parameters were arbitrarily fixed, such as the ratio of communication/avoidance range, for instance. These dimensions of the parameter space remain to be further studied.

Moreover, the discovery of the potential of the β-algorithm has revealed areas of research whose investigation was beyond the scope of the thesis. In this section we articulate some of the potential for further research that we believe might deliver interesting results.

7.3.1 real robots implementation

The most obvious area for further work, because of the practical difficulties encountered by the present study, is in the real robot implementation of the behaviours that still need methodological confirmation. Implementation should either proceed through a better localisation simulation framework, such as video frame grabbing for instance, or concentrate on the realisation of a dedicated radio device with all the hardware and software difficulties inherent to this choice, such as message collision for instance. Even more interesting would be the investigation of other communication media, more compatible to the framework of minimalist robotics, perhaps sound or chemical communication.

7.3.2 communication protocols

As the β-algorithm has shown its ability to tune the connectivity of the communication network, it is now of great importance to study the communication properties of the dynamic network created. Of primary interest is the investigation of different routing protocols that will make, despite the constant reorganisation of the network, multi-hop exchange of information possible. Indeed, the possibility for two robots to communicate thanks to the relay of messages by other robots, is the motivation underlying this thesis. A great success would be to achieve this routing with only the help of the information already provided by the β-algorithm, namely the neighbours' list of neighbours. Furthermore, the routing protocol could be linked to the behaviour of the robots, in order to self-organise a reliable dynamical communication network.

This advance could lead to applications in large mobile sensor arrays which, as demonstrated, could exhibit adaptation of their shape to provide appropriate sensing. Such protocols are highly sought after by telecom companies to develop a potentially infrastructure free mobile telephone network.

7.3.3 behaviours

The behaviours presented need extensive investigation to determine the precise role of the different parameters; to cite just a few, the influence of randomness, communication range/avoidance range ratio, obstacles' sizes or different topologies of the radial-β-algorithm. As differentiation within the swarm has shown, its potential for complex behaviour and pattern formation, the potential of the remaining parameters for further differentiation should be investigated as well, for instance differentiation in the scope of the random turn after reconnection.

As the localisation-β-algorithm and the area estimators used only connection degree information, the ID information provides room for improvement that should be investigated; as also should the potential use of these results to influence the behaviour of the robots. More generally, the use of internal cues to induce dynamical differentiation within the swarm should yield further results. The evolution framework of section 6.3 attempted an automatic search of this potential, but more focused research on this particular topic might be more successful.

The phagocyte behaviour presented in the summary of chapter 5 could have interesting real applications. Therefore, more insight must be gained by thorough investigation of this situation. We suggest the study of an algorithm able to decide whether the beacon has been enclosed or not (possibly weßnitzer's collective decision algorithm [Weßnitzer and Melhuish, 2003]).

7.3.4 analysis

Powerful tools of analysis are crucially needed in the field of collective robotics [Winfield, 2004]. There have been several propositions already (see section 2.13) and the most promising seems to be the probabilistic modeling approach initiated by Martinoli [Martinoli and Mondada, 1998, Lerman and Galstyan, 2001]. The application of such an approach to the problems presented in this thesis is of remarkable interest. Although it is possible that, by their inherent dynamicity and unconstrained environment, the behaviours presented in this thesis show themselves resistant to the methodology. Nevertheless, these behaviours strongly suggest that some fundamental property is involved which would yield to be formal analytical investigation.

7.3.5 learning and evolution

An area that has been completely, although consciously, neglected is the application of learning algorithms across the swarm. We believe that the use of learning has the potential to improve the adaptability of the coherent swarm to changing environments.

The relative failure of the presented genotype-phenotype mechanism to converge through artificial evolution should not be taken as a proof of impossibility. Within this framework there is indeed room for improvement, through the investigation of different variants of the mechanism for instance. Better computing resources together with a more noise-resilient genotype mechanism, a less dynamic diffusion operator and a less difficult task, should lead to more positive results. A serious constraint of the search space should also be considered.

Potentially the evolution approach applied to our dynamical swarm is open to the development of any shape for which we can produce a fitness function. Thus this direction merits further effort to eventually produce its first results.

Chapter 8

Appendices

8.1 Appendix A: upper bound for area coverage

Assuming that the robots have a circular range of communication we are going to produce an upper bound of the area covered by the swarm using only information on the distribution of degrees of connection. As stated in chapter 3, the advantage of this estimator is its local definition. Indeed the β-algorithm requires an individual robot to receive local information on connections. Hence the method described below gives a robot an upper bound of the area covered by itself and its neighbours, and could be used to influence its behaviour.

Using simple geometric properties we compute the contribution of a single robot to the area in all possible cases of degree of connections. Then the upper bound will be given by the following formula:

$$A = \sum_{i=1}^{Nrobots} \alpha(k_i)$$

where k_i is the degree of connections and $\alpha(n)$ is a function that returns the maximal contribution according to degree n.

To compute the maximal area for a degree value of k we put $k+1$ robots symmetrically on the largest possible circle such that they are all interconnected, *i.e.* such that they form a *complete graph*. We compute the area and divide by $k+1$ to get the individual contribution to the area. The robots are placed on a circle in order to get the same value for each robot by symmetry.

When we draw the area covered by the robots we get a *rosace*(see figure 8.1). It is possible to compute the area by subtracting all overlaps to the addition of $k+1$ single robot areas; in other words we add the area as if the robots were all far from each other and then subtract all overlaps. We label the regions starting from outside the area towards the center, and each time we cross a circle boundary we increase by one.

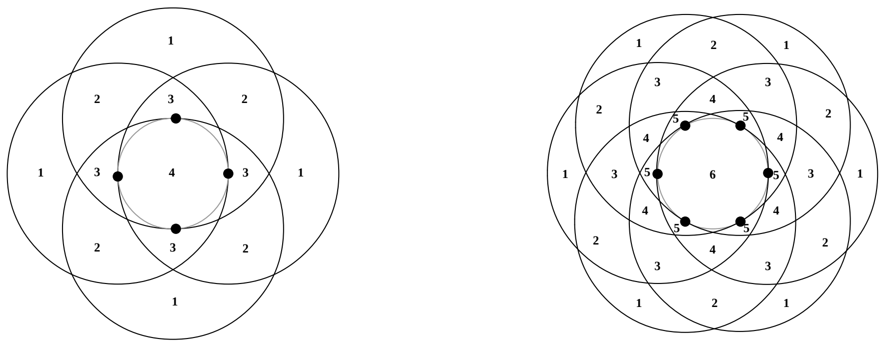


Figure 8.1: rosaces with $k=3$ and $k=5$

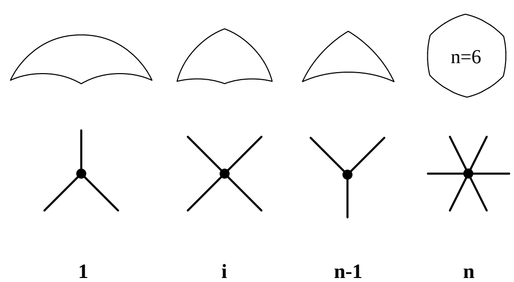


Figure 8.2: different region types and their corresponding graph

In such a drawing we get four different types of regions (see figure 8.2). By subtraction we do not need to compute the first case and get the following formula:

for the whole rosace

$$A_k = na - nB_2^n - 2nB_3^n - \cdots - n(n-2)B_{n-1}^n - (n-1)B_n^n$$

for the individual contribution

$$a_k = \frac{A_k}{n} = a - B_2^n - 2B_3^n - \cdots - (n-2)B_{n-1}^n - \frac{(n-1)}{n}B_n^n$$

where $n = k+1$ is the number of robots, a the area covered by a single robot and B_i^j the area of the division i dependent on the number of robots j.

8.1.1 Proof of formula

we firstly have to define more formally how the divisions are labeled. To this end we define the graph of the divided surface as follow: a vertex represents a region of the plane defined by the rosace and there is an edge between two vertices if the regions are adjacent (see figure 8.3).

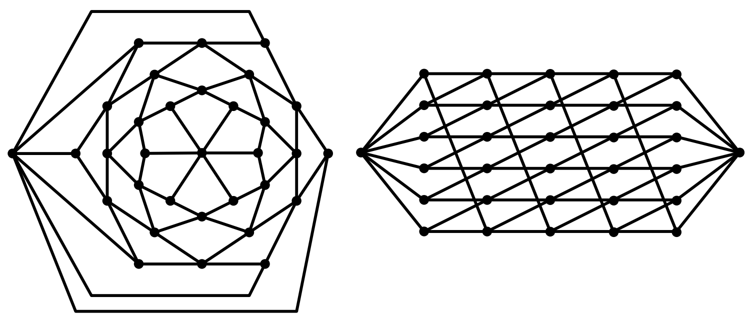


Figure 8.3: two representations of the graph of the rosace with $k = 5$

We then define a *labeling path* from the vertex representing the outermost region to the vertex representing the innermost region - *i.e.* a path from outside the rosace to the center - as a path that starts from the outmost vertex and is allowed to go through an edge if this edge represents the boundary of a circle the path has not yet crossed. As in a rosace all circles intersect in the center, such a path is obviously well defined and it ends in the center of the rosace.

Now we take a labeling path as defined and label the regions starting with 0 for the outmost vertex and increasing by 1 for each vertex along the path. To check whether these labels are well defined we need to be sure that two different labeling paths going through the same region will give the same label to that region. In order to do that we need the general graph of the rosace (see figure 8.4).

On such a representation of the graph, a labeling path is a path that starts on the left and always go towards right until the center vertex is reached. Indeed to go towards left would make the path longer than n edges and the rosace has only n different circles. The path would therefore not be a labeling path. The well-definition of the labels directly follows from that remark.

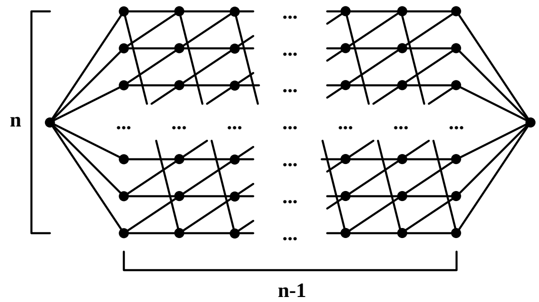


Figure 8.4: the graph of a rosace with $k = n - 1$

In order to compute the area, we can just add all the regions and get the following formula:

$$A_k = n \sum_{i=1}^{n-1} B_i^n + B_n^n$$

Alternatively and because the addition of many non-exact small quantities can lead to noticeable errors, we prefer to proceed by subtraction of all overlaps from the area covered by a single robot. As each time we cross a new circle we increase it by one, the label actually represents the number of circles overlapping a region. The regions with a label equal to one are already counted in the single robot area. The innermost region is counted n times; that means $(n-1)$ too many times. Finally the regions with label $2 \leq i \leq n-1$ are each counted i times; that is $(i-1)$ too many and there are n such regions in the area. The formula becomes now:

$$A_k = na - n \sum_{i=2}^{n-1} (i-1) B_i^n - (n-1) B_n^n$$

as stated earlier.

What remains to be proved is that the general graph of the rosace is indeed the one depicted in figure 8.4.

8.1.2 Proof of the graph

To prove the general graph, we define an *E-set* $E \subset \mathbb{Z}_n$ such as:

$$E = \begin{cases} \varnothing \\ i, \bar{i} \in \mathbb{Z}_n \\ E \quad s.t. \quad if \quad \bar{i} \in E \quad then \quad \bar{i} \in \mathbb{Z}_n \\ \qquad and \quad \bar{i}, \bar{j} \in E \Rightarrow \bar{i}+1 \quad or \quad \bar{j}+1 \in E \end{cases}$$

More intuitively, an E-set E is a "connected" subset of $\mathbb{Z}_n$. Using the obvious bijection between $\mathbb{Z}_n$ and the centers of the robots' communication circles on the base circle [1], there is a connected interval of the base circle containing all centers in E and no others (see figure 8.5).

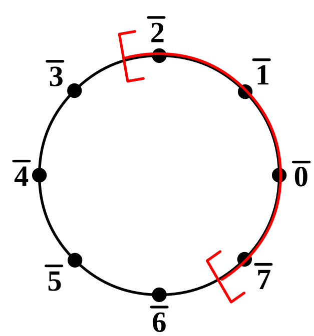


Figure 8.5: An E-set on the circle

Now we define a subset of the plane Π as follow:

$$I_E = \bigcap_{\bar{i} \in E} \mathcal{C}_{\bar{i}} \setminus \{ \bigcup_{\bar{j} \notin E} (\bigcap_{\bar{i} \in E} \mathcal{C}_{\bar{i}}) \cap \mathcal{C}_{\bar{j}} \}$$

Let our graph be $G = (V; E)$ with V the set of vertices corresponding to a region of the plane Π, and E the set of edges. Let also E_n be the set of all E-set of $\mathbb{Z}_n$. We then define the bijection

$$\begin{array}{rccc} f: & V & \longrightarrow & E_n \\ & v & \longmapsto & E \\ & \text{outmost} & \longmapsto & \varnothing \end{array}$$

it is well-defined as if $E \neq E'$ then $I_E \cap I_{E'} = \varnothing$ by definition.

[1] obtained by labeling the robots in the trigonometric order starting from the trigonometric origin, for instance

surjection

we have to show $\forall E \in E_n \Rightarrow I_E \neq \varnothing$. In particular , we have to find a $x \in \Pi$ such that $\forall \bar{i} \in E, \quad d(x, c_i) \leq r \quad and \quad \forall \bar{j} \notin E d(x, c_j) > r$. We define the *boundary* of E

$$\partial E = \bar{i} \in E \quad and \quad \bar{i} = \bar{j} + 1, \bar{j} \notin E \quad or \quad \bar{i} + 1 \notin E$$

We have by definition of an E-set that $|\partial E| \leq 2$. Remember the intuitive representation an E-set.

If $|\partial E| = 1$, let us define $d_{\bar{i}\perp}$ as the straight line passing in $\bar{i}$ perpendicular to $\partial \mathcal{C}_i$. Then $d_{\bar{i}||} \perp d_{i\perp}$ in $\bar{i}$ divides the plane in two parts $A \cup B$ where all $\bar{i} \in E$ are A (wlog). Hence $x = \partial \mathcal{C}_i \cap d_{\bar{i}\perp} \cap A$ fulfills $d(x, c_{\bar{j}}) > r$ as $d(x, d_{\bar{i}||}) = d(x, \bar{i}) = r$ (see figure 8.6).

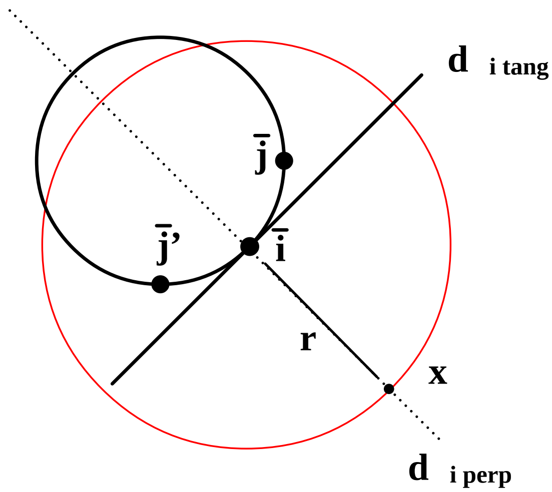


Figure 8.6: case $|\partial E| = 1$

If instead $|\partial E| = 2$ ($E = \{\bar{i}, \bar{i}'\}$) then $\bigcup_{\bar{i} \in \partial E} \mathcal{C}_{\bar{i}}$ has two elements x and x'. But again by definition of an E-set, $d_{\bar{i},\bar{i}'}$ divides the plane in two parts $\Pi = A \cup B$, where one (wlog A) contains all $\bar{j} \notin E$. Define $x = B \cap \bigcup_{\bar{i} \in \partial E} \mathcal{C}_{\bar{i}}$ the element of the circles generated by the boundary of E that stands in B. Then the only points $y \in A$ with $d(x, y) < r$ are in the shaded area $D = A \cap \mathcal{C}(x, r)$ in figure 8.7. Alternatively $\mathcal{C} \cap \mathcal{C}(x, r) = \{\bar{i}, \bar{i}'\}$. As a result and because the intersection of two distinct circles can have at most two elements, $\nexists \bar{j} \notin E$ with $d(x, \bar{j}) < r$.

injection

What needs to be proved is that $\forall E$, I_E is connected. We begin with three remarks:

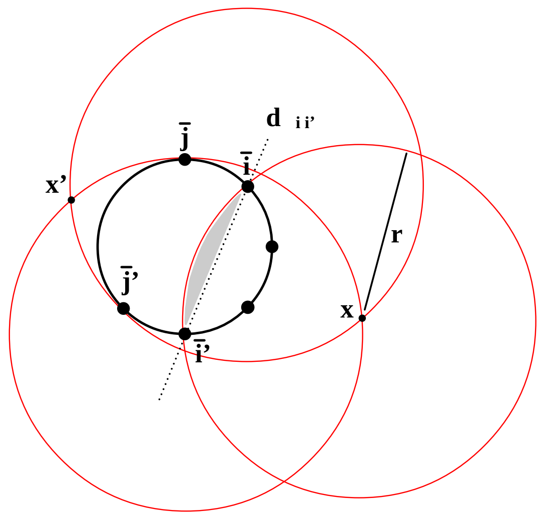


Figure 8.7: case $|\partial E| = 2$

- $\bigcap_{\bar{i}\in\mathcal{P}(\mathbb{Z}_\backslash)} \mathcal{C}_{\bar{i}}$ is *convex, i.e.* $\forall x, y \in \bigcap \mathcal{C}_{\bar{i}}, \quad \bar{xy} \subset \bigcap \mathcal{C}_{\bar{i}}$

- $\bigcap_{\bar{i}\in E} \mathcal{C}_{\bar{i}} \subset \mathcal{C}(c_M(E), l)$, $\forall E \subset \mathbb{Z}_n$, where $c_M(E)$ is the center of mass of the elements in E and l is the maximal diameter of an intersection of $\mathcal{C}_{\bar{i}}$ (see figure 8.8).

- $\bigcup_{\bar{j}\notin E} \mathcal{C}_{\bar{j}}$ overlaps $\mathcal{C}(c_M(E), l)$, $\forall E \subset \mathbb{Z}_n$, as the minimal diameter of a union of $\mathcal{C}_{\bar{i}}$ is greater than l.

The maximal diameter of the intersection of several circles is found with the intersection of two nearest circles. In our case, we take adjacent $c_{\bar{i}}$ as in figure 8.8 where $a = 2\pi/n$, and compute $d = 2R\sin\frac{pi}{n}$ and $l = 2\sqrt{3}R\sin\frac{\pi}{n}$. Then we use the fact that for all B convex, we have $B \setminus (A \cap B)$ connexe if the minimal diameter of A is greater or equal to the maximal diameter of B. Indeed A cannot separate B in two components because it is too large.

Then the graph is proved as follow: an edge in the graph corresponds on the plane Π to crossing a circle. Also to cross a circle corresponds in terms of E-set, through the bijection defined above, to adding or subtracting an element. But to remain in the set of E-sets there are

- 4 possibilities if $1 < |E| < n-1$

- 3 possibilities if $|E| = 1$ or $|E| = n-1$

- n possibilities if $E = \varnothing$ or $E = \mathbb{Z}_n$

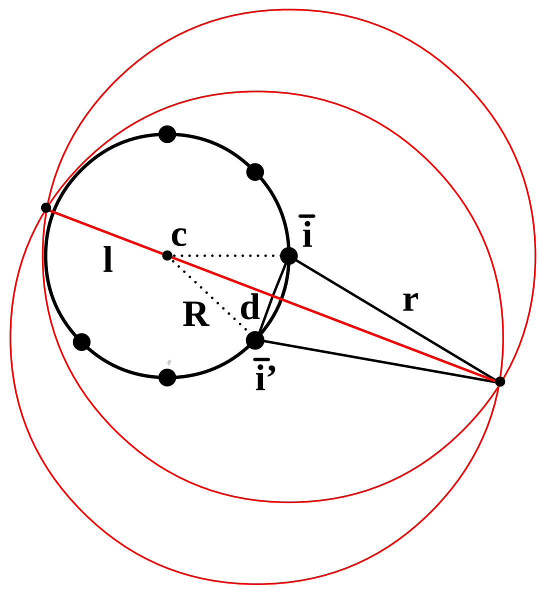


Figure 8.8: l maximal diameter of an intersection of $\mathcal{C}_i$

These possibilities correspond to the degrees of the general graph.

To represent the graph we draw the vertices in columns according to the number of elements in E. There are n different E-set with k elements, hence n rows, and $n+1$ columns (n elements in $\mathbb{Z}_n$ and $\varnothing$). $\square$

8.1.3 computation of B_i^n, $2 \leq i \leq n-2$

Figure 8.9 depicts the situation with the center of the circle labeled with A, and the four robots contributing to the area B_i^n labeled with B, C, D and E.

Half the area B_i^n can be computed using simple geometric tools:

$$B_i^n = 2(A_{\text{triangle } FGH} + A_{\text{segment } CGH} - A_{\text{segment } EFH})$$

where a *segment* is the area of the circular sector defined by an isosceles triangle minus the area of this triangle (see figure 8.10).

In order to compute this area we make use of the available information from the disposition of the robots on the circle. From the polygon $BCDE$ we get

$$\alpha = \frac{i\pi}{n} \qquad \text{and} \qquad \omega = \frac{\pi}{n}$$

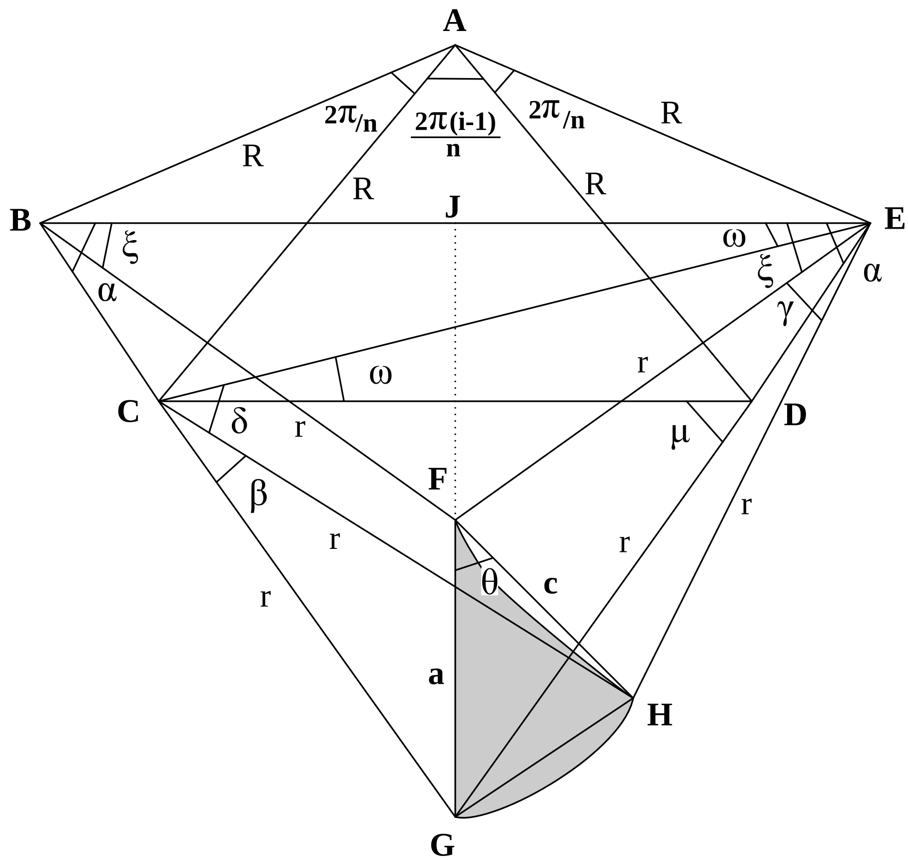


Figure 8.9: schema for the computation of B_i^n

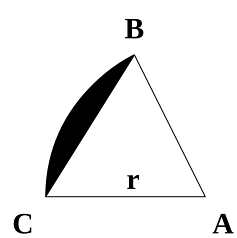


Figure 8.10: a segment area from a triangle ABC

from polygon $EJFGH$,

$$\theta = \xi + \frac{\gamma}{2}$$

from polygon $ACHE$,

$$\delta = \arccos(\frac{R}{r} \sin \frac{i\pi}{n}) \qquad \text{and} \qquad \gamma = \delta + \omega - \xi$$

from polygon $ABFE$,

$$\xi = \arccos(\frac{R}{r}\sin\frac{(i+1)\pi}{n})$$

From polygon $ACGD$,

$$\mu = \arccos(\frac{R}{r}\sin\frac{(i-1)\pi}{n}) \qquad \text{and} \qquad \beta = \mu + \omega - \delta$$

from polygon $BCGDEJ$,

$$a = 2R\sin\frac{\pi}{n}\sin\frac{i\pi}{n} + r\sin(\mu) - r\sin(\xi)$$

and finally from triangle EFH

$$c = 2r\sin\frac{\gamma}{2}$$

Then area B_i^n is computed as follows:

$$B_i^n = ac\sin\theta + r^2(\beta - \sin\beta) - r^2(\gamma - \sin\gamma)$$

computation of R

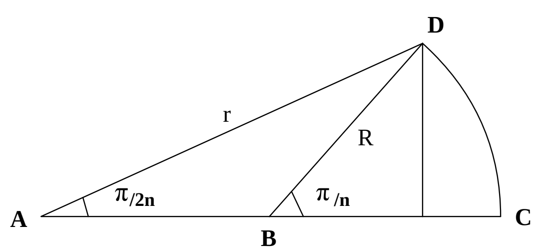


Figure 8.11: calculation of R for n odd

If n is even, robots are placed on the circle in front of each other, (*i.e.* robots are distributed symmetrically on the circle with circular symmetry of center B). In this case the radius of the maximal circle is

$$R = \frac{r}{2}$$

where r is the communication radius.

From figure 8.11 which depicts the odd case with a robot situated in A and another in B, follows

$$R = \frac{r}{2\cos\frac{\pi}{2n}}$$

8.1.4 computation of B_{n-1}^n

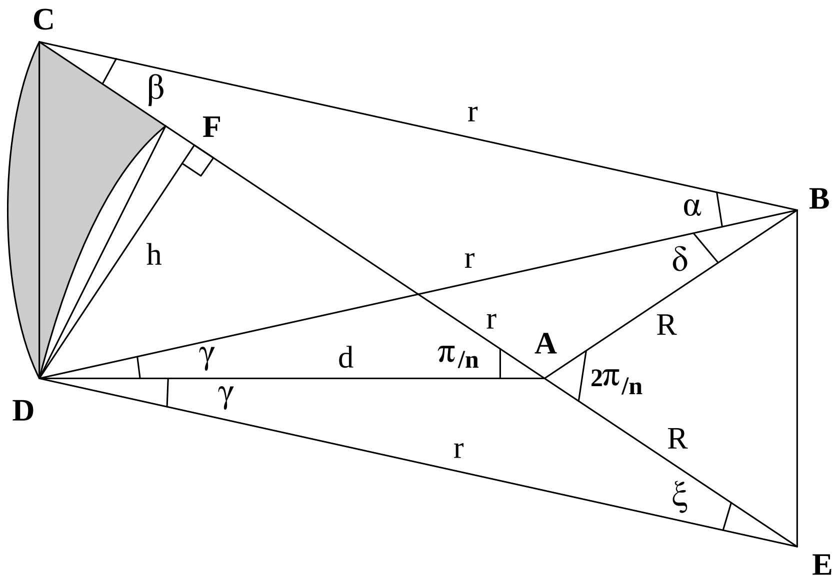


Figure 8.12: schema for the computation of B_{n-1}^n

Half the area B_{n-1}^n can be computed by considering the communication disks of two neighbouring robots on the circle. Figure 8.12 depicts the situation with the center of the circle labeled with A, and both robots contributing labeled with E and B. For the computation of n even or odd, we use the same schema with a different R value.

We firstly note from the isosceles triangle BDE that $\delta = \frac{pi}{n} - \gamma$. Next we compute from the triangle ABC,

$$\begin{aligned}\frac{\sin\beta}{R} &= \frac{\sin(\pi - \frac{2\pi}{n})}{r} \\ \beta &= \arcsin(\frac{R}{r}\sin(\pi - \frac{2\pi}{n}))\end{aligned}$$

from the triangle ABD,

$$\begin{aligned}\frac{\sin\gamma}{R} &= \frac{\sin(\pi - \frac{\pi}{n})}{r} \\ \gamma &= \arcsin(\frac{R}{r}\sin(\pi - \frac{\pi}{n}))\end{aligned}$$

from the triangle BCE,

$$\alpha = \frac{\pi}{n} - \beta + \gamma$$

and

$$\begin{aligned}\frac{\sin(\frac{\pi}{2}-\frac{\pi}{n})}{r} &= \frac{\sin(\frac{\pi}{2}+\frac{\pi}{n}-\beta)}{L} \\ L &= r\frac{\cos(\beta-\frac{\pi}{n})}{\cos\frac{\pi}{n}}\end{aligned}$$

where L is the segment $\overline{CE}$; and finally we compute from the triangle CDE,

$$\begin{aligned}h &= r\sin(\delta) \\ &= r\sin(\frac{\pi}{n}-\gamma)\end{aligned}$$

Then half the area B^n_{n-1} is computed as follows:

$$\frac{1}{2}B^n_{n-1} = A_{\text{triangle } CDE} - A_{\text{sector } DEF} + A_{\text{segment } BCD}$$

where A stands for area. Hence,

$$\begin{aligned}B^n_{n-1} &= 2\Big(\frac{r}{2}L\sin(\frac{\pi}{n}-\gamma)) - \frac{r^2}{2}(\frac{\pi}{n}-\gamma) + \frac{r^2}{2}(\alpha-\sin\alpha)\Big) \\ &= r^2\Big(\frac{\cos(\beta-\frac{\pi}{n})}{cos\frac{\pi}{n}}\sin(\frac{\pi}{n}-\gamma) - \sin(\frac{\pi}{n}-\beta+\gamma) + 2\beta-\gamma\Big)\end{aligned}$$

8.1.5 computation of B^n_n

For the computation of B^n_n, we distinguish between the odd and even case.

n odd

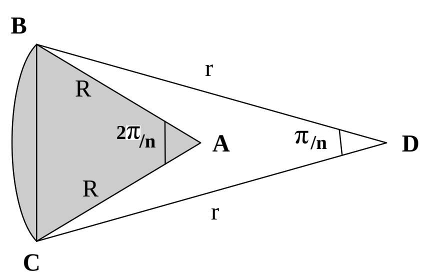


Figure 8.13: schema for the computation of $B^n_n, oddcase$

The area B^n_n can be computed as follows:

$$B^n_n = A_{\text{polygon}} + nA_{\text{segment}}$$

As the polygon is formed of n isosceles triangles with isosceles edges of length R,

$$B_n^n = n(A_{\text{triangle}} + A_{\text{segment}})$$

Figure 8.13 depicts the triangle situation with the center of the circle labeled with A, and the three robots involved labeled with B,C and D. Remember that for the computation of n odd, $R = \frac{r}{2\cos\frac{\pi}{2n}}$.

Then,

$$\begin{aligned} B_n^n &= \frac{R^2}{2}\sin\frac{2\pi}{n} + n\frac{r^2}{2}(\frac{\pi}{n} - \sin\frac{\pi}{n}) \\ &= \frac{nr^2}{2}\left(\frac{\pi}{n} - \sin\frac{\pi}{n} + \frac{\sin\frac{2\pi}{n}}{4\cos\frac{\pi}{2n}}\right) \end{aligned}$$

n **even**

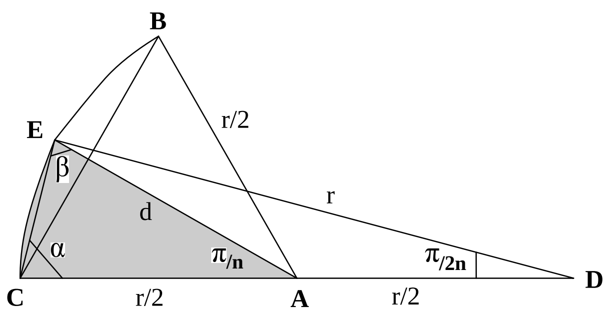


Figure 8.14: schema for the computation of B_n^n, even case

Half the area can be computed using the same technique as used for the computation of B_{n-1}^n:

$$B_n^n = 2n(A_{\text{triangle } AEC} + A_{\text{segment } DEC})$$

Figure 8.14 depicts the n even case with the center of the circle labeled with A, and the three robots involved labeled with B,C and D.

From triangle DEC, we get

$$\alpha = \frac{\pi}{2} - \frac{\pi}{4n}$$

From triangle ABC,

$$\beta = \frac{\pi}{2} - \frac{3\pi}{4n}$$

and

$$\begin{aligned}
\frac{\sin\beta}{\frac{r}{2}} &= \frac{\sin\alpha}{d} \\
d &= \frac{r}{2}\frac{\sin\frac{\pi}{4n}}{\sin\frac{3\pi}{4n}}
\end{aligned}$$

Then the area is computed as follows:

$$\begin{aligned}
B_n^n &= 2n\Big(\frac{r}{2}\frac{d\sin\frac{\pi}{n}}{2} + \frac{r^2}{2}(\frac{\pi}{2n} - \sin\frac{\pi}{2n})\Big) \\
&= 2n\Big(\frac{r^2}{8}\frac{\sin\frac{4\pi}{n}\sin\frac{\pi}{n}}{\sin\frac{3\pi}{4n}} + \frac{r^2}{2}(\frac{\pi}{2n} - \sin\frac{\pi}{2n})\Big) \\
&= nr^2\Big(\frac{\sin\frac{4\pi}{n}\sin\frac{\pi}{n}}{4\sin\frac{3\pi}{4n}} + \frac{\pi}{2n} - \sin\frac{\pi}{2n}\Big)
\end{aligned}$$

8.1.6 computation of special case $n = 2$

As the formula for the rosace does not hold for $n = 2$, this case has to be dealt with separately. In fact this case is equivalent to the case presented in section 8.2.5 with $r = R$. We therefore get an expression of the maximal area covered by two robots with communication radius r:

$$A = r^2\Big(\frac{2}{3}\pi + \frac{\sqrt{3}}{4}\Big)$$

thus,

$$\alpha(1) = \frac{r^2}{2}\Big(\frac{2}{3}\pi + \frac{\sqrt{3}}{4}\Big)$$

8.2 Appendix B: lower bound for area coverage

As the robots have proximity sensors with lower range than the communication range, it is possible, with analogous reasoning as used in the computation of the upper bound, to compute a lower bound of the area coverage of the swarm. Using simple geometric properties we again compute the minimal contribution of a single robot to the global area in all possible cases of degree of connections. Then the lower bound is given by the following formula:

$$A = \sum_{i=1}^{Nrobots} \lambda(k_i)$$

where k_i is the degree of connections and $\lambda(n)$ is a function that returns the minimal contribution according to degree n.

8.2.1 growth

If the robots have to keep a minimal distance, the area will be minimal if they are situated on an equilateral grid. Hence to compute the minimal area for a degree value of k we put $k+1$ robots on an equilateral grid of edge length R - the avoidance radius, such that they are all interconnected and cover an area as symmetrical as possible. To this extent, we start with three robots forming the first equilateral triangle and then grow the shape according to the growth rule depicted in figure 8.15 (the notation corresponds to the number of equilateral length R on the edges of the 6-polygon). We then compute the area and divide by $k+1$ to get the individual contribution to the area.

Unlike for the computation of the upper bound, we will start from the area covered by two robots standing at minimal distance to each other, and then compute the new contribution of each added robot on the equilateral grid. Then the expression of the minimal contribution is

$$\lambda(n) = \frac{1}{n+1}\Big(\text{growth}_2 \ + \sum_{i=3}^{n+1} \text{growth}_i\Big)$$

If $n \geq 3$, growing the previous group by adding a robot corresponds to the addition of a vertex on the equilateral grid occupied by the group. According to the position of this added vertex, the growth of the overall covered area can be different. Following the growth rule of figure 8.15, we have to compute 5 different cases (figure 8.16).

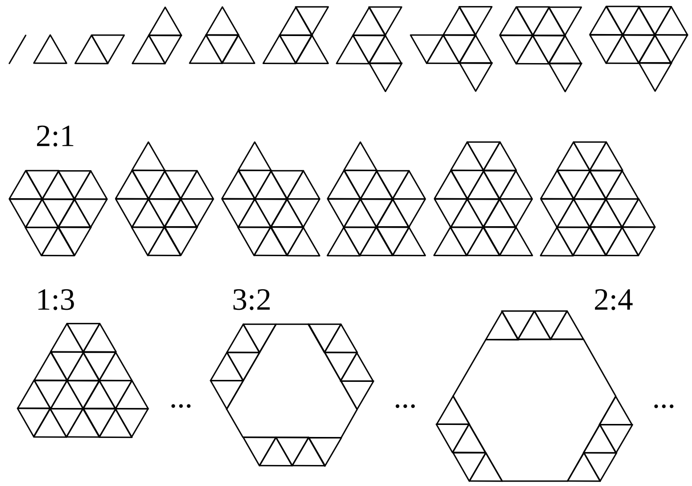


Figure 8.15: minimal area growth

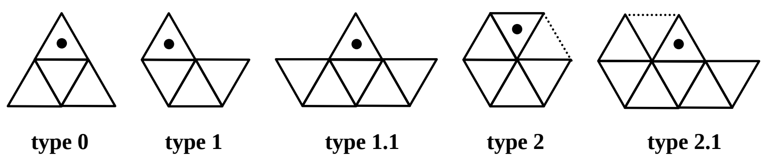


Figure 8.16: area growth types

Following the growth rule, we compute the first cases and show the behaviour of the sum in figure 8.17. No proof is given as the communication radius/avoidance radius ratio constrains in our case the number of robots to $N \leq 36$.

To compute the contribution to the growth of the overall covered area for each growth type, we consider figure 8.18 and derive the following equations

$$
\begin{aligned}
A_0 &= A \\
A_1 &= A - C \\
A_{1.1} &= A - 2C \\
A_2 &= B \\
A_{2.1} &= B - C
\end{aligned}
$$

What remains to be computed are the areas A, B and C.

		growth types				
notation	nb of robots	0	1	1.1	2	2.1
	3	1				
	6	4				
	9	4	3			
2:1	12	4	3		3	
1:3	18	4	6		6	
3:2	27	4	6	3	12	
2:4	36	4	6	6	18	
4:3	66	4	6	9	24	3
3:5	60	4	6	12	30	6
5:4	75	4	6	15	36	12
4:6	90	4	6	18	42	18
6:5	106	4	6	21	48	27
5:7	126	4	6	24	54	36
7:6	147	4	6	27	60	48
6:8	168	4	6	30	66	60

Figure 8.17: computation of the first terms of the sum

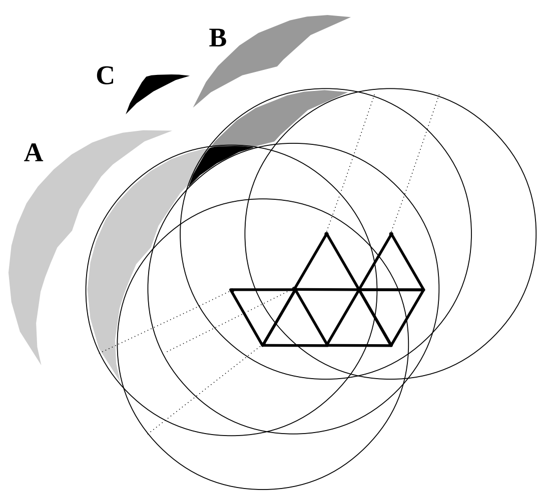


Figure 8.18: computation of area types

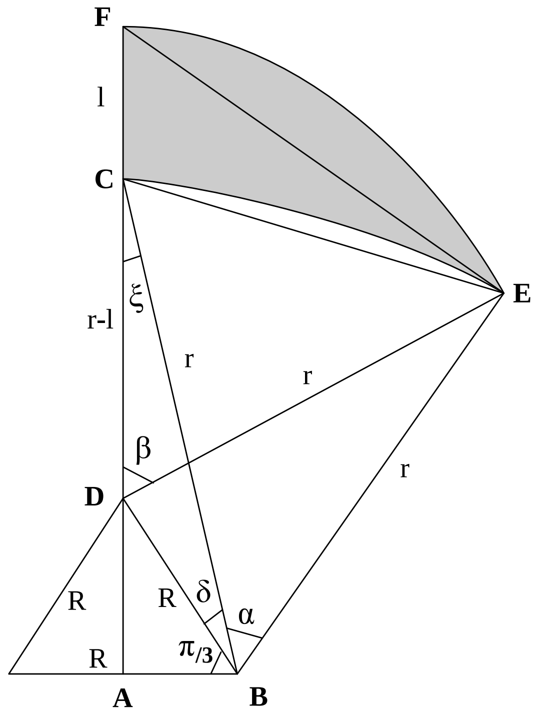


Figure 8.19: schema for the computation of A

8.2.2 computation of A

Figure 8.19 depicts schematically the situation with two robots labeled B and D; they stand at a distance R of each other, on the equilateral grid. In order to compute the different terms of this equation, we consider several polygons. From triangle ABC we firstly get

$$\delta = \frac{\pi}{6} - \xi \qquad \text{and} \qquad \xi = \arcsin \frac{R}{2r}$$

from triangle BED

$$\alpha = \arccos \frac{R}{2r} \frac{\pi}{6} + \xi$$

from triangle BCD

$$\begin{aligned} \alpha + \delta + \beta + \xi + \delta &= \pi \\ \beta &= \tfrac{2\pi}{3} + \xi - \alpha \end{aligned}$$

and

$$\begin{aligned}
\frac{\sin(\frac{\pi}{6} - \xi}{r - l} &= \frac{\sin \xi}{R} \\
r - l &= 2r \sin(\frac{\pi}{6} - \xi) \\
l &= r\Big(1 - 2\sin(\frac{\pi}{6} - \xi)\Big)
\end{aligned}$$

Then an expression for the area A is given by

$$\begin{aligned}
A &= 2\Big(A_{\text{triangle } CEF} + A_{\text{segment } DEF} - A_{\text{segment } BCE}\Big) \\
&= 2\Big(\frac{l}{2}\sin(\frac{\pi}{2} - \frac{\beta}{2})2r\sin\frac{\alpha}{2} + \frac{r}{2}(\beta - \sin\beta) - \frac{r}{2}(\alpha - \sin\alpha)\Big) \\
&= r\Big(2l\sin(\frac{\pi}{2} - \frac{\beta}{2})\sin\frac{\alpha}{2} + (\beta - \alpha) - (\sin\beta - \sin\alpha)\Big)
\end{aligned}$$

8.2.3 computation of B

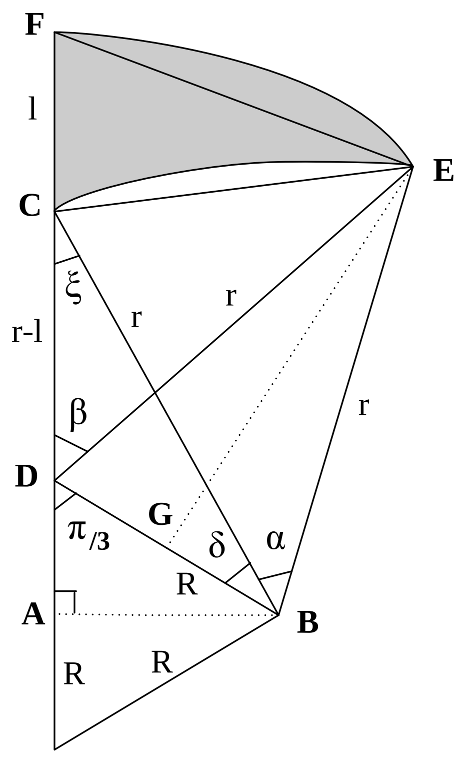


Figure 8.20: schema for the computation of B

Figure 8.20 depicts schematically the situation with two robots labeled B and D; they stand at a distance R of each other. In order to compute the different terms of this equation, we consider several polygons. From triangle ABC we firstly get

$$\begin{aligned} \sin\xi &= \frac{R\sin\frac{\pi}{3}}{r} \\ \xi &= \arcsin\left(\frac{\sqrt{3}R}{2r}\right) \end{aligned}$$

and

$$\delta = \frac{\pi}{3} - \xi$$

from triangle BEG

$$\alpha = \arccos\frac{R}{2r} + \xi - \frac{\pi}{3}$$

from triangle BCD

$$\begin{aligned} \delta + \xi + \beta + \alpha + \delta &= \pi \\ \beta &= \tfrac{\pi}{3} + \xi - \alpha \end{aligned}$$

and

$$\begin{aligned} \frac{\sin(\frac{\pi}{3} - \xi}{r - l} &= \frac{\sin\xi}{R} \\ r - l &= \frac{2r}{\sqrt{3}}\sin(\frac{\pi}{3} - \xi) \\ l &= r\left(1 - \frac{2}{\sqrt{3}}\sin(\frac{\pi}{3} - \xi)\right) \end{aligned}$$

Then an expression for the area B is given by

$$\begin{aligned} B &= 2\left(A_{\text{triangle } CEF} + A_{\text{segment } DEF} - A_{\text{segment } BCE}\right) \\ &= 2\left(\frac{l}{2}\sin(\frac{\pi}{2} - \frac{\beta}{2})2r\sin\frac{\alpha}{2} + \frac{r}{2}(\beta - \sin\beta) - \frac{r}{2}(\alpha - \sin\alpha)\right) \\ &= r\left(2l\sin(\frac{\pi}{2} - \frac{\beta}{2})\sin\frac{\alpha}{2} + (\beta - \alpha) - (\sin\beta - \sin\alpha)\right) \end{aligned}$$

8.2.4 computation of C

Figure 8.21 depicts schematically the situation with three robots labeled A, D and H. In order to compute the different terms of this equation, we consider several polygons. From the polygon $AHCD$ we firstly get

$$\delta = \frac{\pi}{3}$$

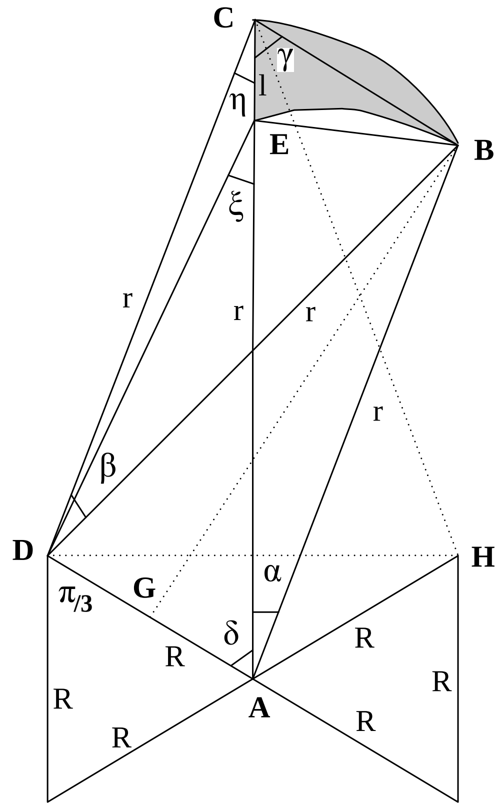


Figure 8.21: schema for the computation of C

as the robots are placed on the hexagonal grid; from triangle BCD

$$d = 2r \sin \frac{\beta}{2}$$

from triangle ABG

$$\alpha = \arccos \frac{R}{2r} - \delta$$

from triangle ACD

$$\begin{aligned} \frac{\sin \delta}{r} &= \frac{\sin \eta}{R} \\ \eta &= \arcsin \left(\frac{\sqrt{3}R}{2r} \right) \end{aligned}$$

and

$$\beta = \frac{\pi}{3} - \eta - \alpha$$

from triangle ABC

$$\begin{aligned}
\frac{\sin\gamma}{r} &= \frac{\sin\alpha}{d} \\
\sin\gamma &= \frac{\sin\alpha}{2\sin\frac{\beta}{2}} \\
\gamma &= \arcsin\Big(\frac{\sin\alpha}{2\sin\frac{\beta}{2}}\Big)
\end{aligned}$$

back to triangle ACD, we get

$$\begin{aligned}
\frac{\sin(\beta+\alpha+\delta)}{r+l} &= \frac{\sin\delta}{r} \\
r+l &= \frac{2r}{\sqrt{3}}\sin(\beta+\alpha+\frac{\pi}{3}) \\
l &= r\Big(\frac{2}{\sqrt{3}}\sin(\beta+\alpha+\frac{\pi}{3})-1\Big)
\end{aligned}$$

Then an expression for the area C is given by

$$\begin{aligned}
C &= 2\Big(A_{\text{triangle } BCE} + A_{\text{segment } BCD} - A_{\text{segment } ABE}\Big) \\
&= 2\Big(\frac{ld}{2}\sin\gamma + \frac{r}{2}(\beta-\sin\beta) - \frac{r}{2}(\alpha-\sin\alpha)\Big) \\
&= ld\sin\gamma + r(\beta-\sin\beta) - r(\alpha-\sin\alpha)
\end{aligned}$$

8.2.5 computation of special case $n = 2$

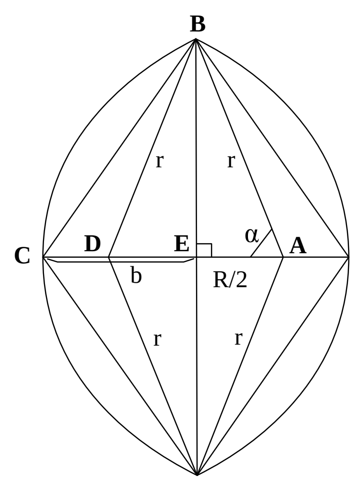


Figure 8.22: schema for the computation of the special case of two robots

Again, the case $n = 2$ has to be dealt with separately. Figure 8.22 depicts schematically the situation with the two robots labeled A and D standing at a distance R of each other. An expression for the area covered by the pair is given by

$$A = 2\pi r^2 - 4\Big(A_{\text{triangle } EBC} + A_{\text{segment } ABC}\Big)$$

which then gives

$$\lambda(1) = \pi r^2 - 2\Big(A_{\text{triangle } EBC} + A_{\text{segment } ABC}\Big)$$

In order to compute the different terms of this equation, we consider several polygons. From the triangle ABE we firstly get

$$\alpha = \arccos\frac{R}{2r} \qquad \text{and} \qquad h = \sqrt{r^2 - \frac{R^2}{4}}$$

Then the area is given by

$$\begin{aligned}
A &= 2\pi r^2 - 4\Big(\frac{1}{2}(r - \frac{R}{2})h + \frac{r^2}{2}(\alpha - \sin\alpha)\Big) \\
&= 2\Big(\pi r^2 - (r - \frac{R}{2})\sqrt{r^2 - \frac{R^2}{4}} - r^2(\alpha - \sin\alpha)\Big) \\
\lambda(1) &= \pi r^2 - (r - \frac{R}{2})\sqrt{r^2 - \frac{R^2}{4}} - r^2(\alpha - \sin\alpha)
\end{aligned}$$

Bibliography

[Abelson et al., 2000] Abelson, H., Allen, D., Coore, D., Hanson, C., Homsy, G., Knight, T., Nagpal, R., Rauch, E., Sussman, G., and Weiss, R. (2000). Amorphous computing. *Communication of the ACM*, 43(5):74–83.

[Ackermann, 2001] Ackermann, E. (2001). Piaget's constructivism, papert's constructionism: What's the difference. In de la recherche en education, S., (Ed.), *Constructivismes: usages et perspectives en education.*, volume 8, pages 85–94.

[Adams, 1997] Adams, D. G. (1997). Cyanobacteria. In Dworkin, M. and Shapiro, J., (Eds.), *Bacteria as Multicellular Organisms*, pages 109–148. Oxford University Press.

[Adler and Gordon, 1992] Adler, F. and Gordon, D. (1992). Information collection and spread by network of patrolling ants. *American Naturalist*, 140(3):373–400.

[Albiez et al., 2002] Albiez, J., Luksch, T., Berns, K., and Dillmann, R. (2002). An activation based behaviour control architecture for walking machines. In Hallam, B. et al., (Eds.), *From Animals to Animats SAB'7*, volume 7, pages 118–126.

[Arai et al., 1993] Arai, T., Yoshida, E., and Ota, J. (1993). Information diffusion by local communication of multiple mobile robots. In *IEEE int. Conf. on Systems, Man & Cybernetics*, pages 535–540.

[Arkin, 1998] Arkin, R. (1998). *Behaviour-Based Robotics.* MIT Press.

[Artaud et al., 2004] Artaud, G., Plancke, P., Magness, R., Durrant, D., and Plummer, C. (2004). Ieee 802.15.4: Wireless transducer networks. In *Datasystems In Aerospace, DASIA'04*, Nice.

[Bäck et al., 1991] Bäck, T., Hoffmeister, F., and Schwefel, H.-P. (1991). A survey of evolution strategies. In *4th Int. Conf. on Genetic Algorithms*, pages 2–9.

[Balch and Arkin, 1994] Balch, T. and Arkin, R. (1994). Communication in reactive multiagent robotic systems. *Autonomous Robots*, 1:1–25.

[Balch and Arkin, 1998] Balch, T. and Arkin, R. (1998). Behaviour-based formation control for multi-robot teams. *IEEE Transactions on Robotics & Automation*, 14(6):926–939.

[Balch and Hybinette, 2000] Balch, T. and Hybinette, M. (2000). Social potentials for scalable multi-robot formations. In *Proc.Int.Conf. on Robotics and Automation ICRA'00*, volume 1, pages 73–80.

[Baldassare et al., 2003] Baldassare, G., Nolfi, S., and Parisi, D. (2003). Evolving mobile robots able to display collective behaviours. *Artificial Life*, 9(2):255–267.

[Banzhaf, 2003] Banzhaf, W. (2003). On the dynamics of an artificial regulatory network. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life ECAL'03*, volume 7, pages 217–227.

[Beckers et al., 1994] Beckers, R., Holland, O., and Deneubourg, J.-L. (1994). From local actions to global tasks: Stigmergy and collective robotics. In Press, M., (Ed.), *Artificial Life IV*, pages 181–189.

[Ben-Jacob and Cohen, 1997] Ben-Jacob, E. and Cohen, I. (1997). Cooperative formation of bacterial patterns. In Dworkin, M. and Shapiro, J., (Eds.), *Bacteria as Multicellular Organisms*, pages 394–416. Oxford University Press.

[Beni and Wang, 1991] Beni, G. and Wang, J. (1991). Theoretical problems for the realisation of distributed robotic systems. In *IEEE Int. Conf. on Robotics & Automation*, pages 1914–1920.

[Bentley, 2003] Bentley, P. (2003). Evolving fractal gene regulatory networks for robot control. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life ECAL'03*, volume 7, pages 753–762.

[Billard et al., 1999a] Billard, A., Ijspeert, A., and Martinoli, A. (1999a). Adaptive exploration of a frequently changing environment by a group of communicating robots. In Floreano, D. et al., (Eds.), *Advances in Artfificial Life, ECAL'99*, volume 1674, pages 596–605.

[Billard et al., 1999b] Billard, A., Ijspeert, A., and Martinoli, A. (1999b). A multi-robot system for adaptive exploration of a fast changing environment: Probabilistic modeling and experimental study. *Connection Science Special Issue on Adaptive Robots*, 11(3 and 4):359–379.

[Bisset, 2003] Bisset, D. (2003). Doing robot science. In *TIMR 2003*.

[Bollobás, 1998] Bollobás, B. (1998). *Modern Graph Theory*. Springer.

[Bonabeau et al., 1999] Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm Intelligence - From Natural to Artificial Systems*. Oxford Univ. Press.

[Bonabeau and Theraulaz, 1994] Bonabeau, E. and Theraulaz, G. (1994). Why do we need artificial life? *Artificial Life*, 1:303–325.

[Bongard, 2002] Bongard, J. (2002). Evolving modular genetic regulatory networks. In *Proceedings of the IEEE 2002 Congress on Evolutionary Computation (CEC2002)*, pages 1872–1877.

[Bongard and Paul, 2000] Bongard, J. and Paul, C. (2000). Investigating morphological symmetry and locomotive efficiency using virtual embodied evolution. In Meyer, J.-A. et al., (Eds.), *From Animals to Animats: The Sixth International Conference on the Simulation of Adaptive Behaviour*, pages 420–429.

[Bongard and Pfeifer, 2001] Bongard, J. and Pfeifer, R. (2001). Repeated structure and dissociation of genotypic and phenotypic complexity in artificial ontogeny. In Spector, L. et al., (Eds.), *Proceedings of The Genetic and Evolutionary Computation Conference, GECCO-2001*, pages 829–836.

[Braitenberg, 1984] Braitenberg, V. (1984). *Vehicules - Experiments in Synthetic Psychology*. MIT Press.

[Brodie, 2000] Brodie, S. (2000). Phase transition phenomena in robotic communication, msc dissertation. Technical Report MSc in Distributed Systems and Networks, Computing Laboratory, University of Kent.

[Brooks, 1986] Brooks, R. (1986). A robust layered control system for a mobile robot. *Jour. of Robotics & Automation*, 2:14–23.

[Brooks, 1991] Brooks, R. (1991). Intelligence without reason. In *Proc. Int. Joint Conf.on Artificial Intelligence*, pages 569–595.

[Bulusu et al., 2001] Bulusu, N., Estrin, D., Girod, L., and Heidemann, J. (2001). Scalable coordination for wireless sensor networks: Self-configuring localization systems. In *Proc. Int. Symp. on Communication Theory & Application ISCTA'01.*

[Cao et al., 1995] Cao, Y., Fukunaga, A., Kahng, A., and Meng, F. (1995). Cooperative mobile robotics: Antecedents and directions. In *IEEE Int.Conf. on Intelligent Robots & Systems*, pages 226–234.

[Christensen et al., 2004] Christensen, D., Østergaard, E., and Lund, H. (2004). Metamodule control for the atron self-reconfigurable robotic system. In Groen, P. et al., (Eds.), *8th Int.Conf. on Intelligent & Autonomous Systems, IAS-8*, pages 685–692.

[C.Kube and H.Zhang, 1994] C.Kube and H.Zhang (1994). Collective robotics: From social insects to robots. *Jour. of Adaptive Behaviour*, 2:189–218.

[d'Arcy Thompson, 1917] d'Arcy Thompson, E. (1917). *On Growth and Form.*

[Dedeoglu and Sukhatme, 2000] Dedeoglu, G. and Sukhatme, G. (2000). Landmark-based matching algorithm for cooperative mapping by autonomous robots. In *Distributed Autonomous Robotic Systems*, volume IV, pages 251–260.

[Defago, 2001] Defago, X. (2001). Distributed computing on the move: From mobile computing to cooperative robotics and nanorobotics. In *Proc. 1st ACM Annual Workshop on Principles of Mobile Computing (POMC'01)*, pages 49–55.

[Defago and Konagaya, 2002] Defago, X. and Konagaya, A. (2002). Circle formation for oblivious anonymous mobile robots with no common sense of orientation. In *Mobile Computing, POMC'02*, pages 97–104.

[Delgado and Sole, 1997] Delgado, J. and Sole, R. (1997). Collective-induced computation. *Physical Review E*, 55:2338–2344.

[Dreyfus, 1992] Dreyfus, H. (1979/1992). *What Computers Still Can't Do: A Critique of Artifical Reason*. MIT Press.

[Dudek et al., 1993] Dudek, G., Jenkin, M., Milios, E., and Wilkes, D. (1993). Robust positioning with a multi-agent robotic system. In *Proceedings of the International Joint Conference of Artificial Intelligence (IJCAI-93) Workshop on Dynamically Interacting Robots*, pages 118–123.

[Eggenberger, 1996] Eggenberger, P. (1996). Cell interactions for development in evolutionary robotics. In Maes, P. et al., (Eds.), *From Animals to Animats, SAB 4*, volume 4, pages 440–448.

[Eggenberger, 1997] Eggenberger, P. (1997). Evolving morphologies of simulated 3-d organisms based on differential gene expression. In Husbands, P. and Harvey, I., (Eds.), *Advances in Artificial Life, ECAL'97*, volume 4, pages 205–213.

[Endler, 1993] Endler, J. A. (1993). Some general comments on the evolution and design of animal communication systems. *Phil. Trans. Royal Society London B*, 340:215–225.

[Erdös and Renyi, 1960] Erdös, P. and Renyi, A. (1960). On the evolution of random graphs. *Publ. of the Mathematical Institute of the Hungarian Acad.of Science*, 5:17–61.

[Estrin et al., 1999a] Estrin, D., G.Bekey, Mataric, M., Govindan, R., and Heidemann, J. (1999a). Dynamic adaptive wireless networks with autonomous robot nodes. In *NSF Grant ANI-9979457*.

[Estrin et al., 1999b] Estrin, D., Govindan, R., Heidemann, J., and Kumar, S. (1999b). Next century challenges: Scalable coordination in sensor networks. In *ACM/IEEE Int.Conf. on Mobile Computing and NetworksMOBICOM'00*, pages 263–270.

[Evans, 2000] Evans, D. (2000). Programming the swarm. In *NSF Grant CAREER ANI-0092945*.

[Even and Tarjan, 75] Even, S. and Tarjan, E. (75). Network flow and testing graph connectivity. *SIAM journal of Computation*, 4(4):507–518.

[Floreano and Nolfi, 1998] Floreano, D. and Nolfi, S. (1998). *Evolutionary Robotics*. MIT Press.

[Foreman et al., 2003] Foreman, M., Prokopenko, M., and Wang, P. (2003). Phase transitions in self-organising sensor networks. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life ECAL 2003*, volume 7, pages 781–791.

[Franklin et al., 1995] Franklin, D., Kahng, A., and Lewis, A. (1995). Distributed sensing and probing with multiple search agents: Towards system-level landmine detection solutions. In *Proc. SPIE Aerosense-95: Detection Technologies for Mines and Minelike Targets.*

[Fraser et al., 1995] Fraser, C., Gocayne, J., White, O., Adams, M., Clayton, R., et al. (1995). The minimal gene comlement of mycoplasma genitalium. *Science*, 270:397–402.

[Fredslund and Mataric, 2001] Fredslund, J. and Mataric, M. (2001). Robot formations using only local sensing and control. In *Proc. of IEEE Int.Symp. on Computational Intelligence for Robotics & Automation*, pages 308–313.

[Frehland et al., 1985] Frehland, E., Kleutsch, B., and Markl, H. (1985). Modeling a two-dimensional random alarm process. *BioSystems*, 18:197–208.

[Frutiger et al., 2002] Frutiger, D., Bongard, J., and Iida, F. (2002). Iterative product engineering evolutionary robot design. In Bidaud, P. and Amar, F., (Eds.), *Proceedings of the Fifth International Conference on Climbing and Walking Robots*, pages 619–629.

[Gage, 1993] Gage, D. (1993). How to communicate with zillions of robots. In *SPIE Mobile Robots VIII*, volume 2058, pages 250 – 257.

[Gazi and Passino, 2003] Gazi, V. and Passino, K. (2003). Stability analysis of swarms. *IEEE Trans. on Automatic Control*, 48:692–697.

[Genovese et al., 1992] Genovese, V., Dario, P., Magni, R., and Odetti, L. (1992). Self-organising behaviour and swarm intelligence in a pack of mobile miniature robots in search of pollutants. In *IEEE Int.Conf. on Intelligent Robots & Systems*, pages 1575–1582.

[Gershenson and Heylighen, 2003] Gershenson, C. and Heylighen, F. (2003). When can we call a system self-organizing. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life ECAL 2003*, volume 7, pages 606–614.

[Gil et al., 2003] Gil, R., Silva, F., Zientz, E., Delmotte, F., Gonzalez-Candelas, F., Latorre, A., et al. (2003). The genome sequance of blochmannia floridanus: Comparative analysis of reduced genomes. *PNAS*, 100(16):9388–9393.

[Gilbert, 2002] Gilbert, S. (2002). *Developmental Biology.*

[Greenfield, 1994] Greenfield, M. (1994). Synchronous and alternating choruses in insects and anurans: Common mechanisms and diverse functions. *American Zoologist*, 34:605–615.

[Hackwood and G.Beni, 1992] Hackwood, S. and G.Beni (1992). Self-organization of sensors for swarm intelligence. In *IEEE int. Conf. on Robotics & Automation*, pages 819–829.

[Harvey, 2000] Harvey, I. (2000). Robotics : Philosophy of the mind using a screwdriver. In Gomi, T., (Ed.), *Evolutionary Robotics: From Intelligent Robots to Artificial Life*, volume 3, pages 207–230.

[Hayes et al., 2000] Hayes, A., Martinoli, A., and Goodman, R. (2000). Comparing distributed exploration strategies with simulated and real robots. In *Distributed Autonomous Robotic Systems*, volume IV, pages 261–270.

[Hobbs, 1989] Hobbs, A. (1989). Computing edge-toughness and fractional arboricity. In *Contemporary Mathematics*, volume 4, pages 89–106.

[Hogeweg, 2000a] Hogeweg, P. (2000a). Evolving mechanisms of morphogenesis: On the interplay betwen differential adhesion and cell differentiation. *Journal of Theoretical Biology*, 203:317–333.

[Hogeweg, 2000b] Hogeweg, P. (2000b). Shapes in the shadow: Evolutionary dynamics of morphogenesis. *Artificial Life*, 6:85–101.

[Hogeweg, 2002] Hogeweg, P. (2002). Computing an organism: On the interface between informatic and dynamic processes. *BioSystems*, 64:97–109.

[Holland, 1998] Holland, J. H. (1998). *Emergence - From Chaos to Order.* Oxford Univ. Press.

[Holland and Melhuish, 1996] Holland, O. and Melhuish, C. (1996). Getting the most from the least : Lessons for the nanoscale from minimal mobile agents. In *5th Int. Conf. on Artificial Life.*

[Honary and McFarland, 2003] Honary, E. and McFarland, D. (2003). Flock distortion: A new approach in mapping environmental variables in deep water. *Robotica*, 21:part 4.

[Hornby et al., 2001] Hornby, G., Lipson, H., and Pollack, J. (2001). Evolution of generative design systems for modular physical robots. In *IEEE Int.Conf. on Robotics & Automation (ICRA).*

[Hutchison et al., 1999] Hutchison, C., Peterson, S., Gill, S., Cline, R., White, O., Fraser, C., Smith, H., and Venter, C. (1999). Global transposon mutagenesis and a minimal mycoplasma genome. *Science*, 286:2165–2169.

[Iijima et al., 2000] Iijima, D., Yu, W., Yokoi, H., and Kakazu, Y. (2000). Obstacle avoidance for a distributed autonomous swimming robot by interaction-based learning. In Press, M., (Ed.), *From Animals to Animats SAB'00*, pages 297–306.

[Jacob and Monod, 1961] Jacob, F. and Monod, M. (1961). Genetic regulatory mechanisms in the synthesis of proteins. *Jour. of Molecular Biology*, 3(318).

[Jakobi et al., 1995] Jakobi, N., Husbands, P., and Harvey, I. (1995). Noise and the reality gap : the use of simulation in evolutionary robotics. In Moran, F. et al., (Eds.), *Advances in Artificial Life, ECAL'95*, pages 704–720.

[J.Broch et al., 1998] J.Broch, D.Maltz, D.Johnson, Y.-C.Hu, and J.Jetcheva (1998). A performance comparison of multi-hop wireless ad-hoc network routing protocols. In *4th ACM/IEEE Int.Conf on Mobile Computing and Networking (MobiCom'98)*, pages 85–97.

[Johnson, 1994] Johnson, D. B. (1994). Routing in ad hoc networks of mobile hosts. In *IEEE Workshop on Mobile Computing Systems & Applications*, pages 158–163.

[Kauffman, 1989] Kauffman, S. (1989?). *The Origins of Order.* Oxford Univ. Press.

[Kauffman and Goodwin, 1994] Kauffman, S. and Goodwin, B. (1994). *At Home in The Universe.* Oxford Univ. Press.

[Kawai and Hara, 1994] Kawai, N. and Hara, F. (1994). Formation of morphology and morph-function in a linear-cluster robotic system. In Press, M., (Ed.), *From Animals to Animats SAB'94*, pages 459–464.

[Knoll et al., 2001] Knoll, A., Bekey, G., and Henderson, T. (2001). Cui bono robo sapiens. *Robotics and Autonomous Systems*, 27:73–80.

[Kobayashi et al., 2003] Kobayashi, K., Ehrlich, S., Albertini, A., Amati, G., Andersen, K., Arnaud, M., Asai, K., et al. (2003). Essential bacillus subtilis genes. *PNAS*, 100(8):4678–4683.

[Kotay and Rus, 1999] Kotay, K. and Rus, D. (1999). Locomotion versatility through self-reconfiguration. *Jour. of Robotics & Autonomous Systems*, 26:217–232.

[Krieger and Billeter, 2000] Krieger, M. and Billeter, J.-B. (2000). The call of duty: Self-organised task allocation in a population of up to twelve mobile robots. *Jour. of Robotics & Autonomous Systems*, 30:65–84.

[Kube. and Zhang, 1992] Kube., C. and Zhang, H. (1992). Collective robotic intelligence. In *2nd Int.Conf. Simulation of Adaptive Behaviour*, pages 460–468.

[Kubik, 2003] Kubik, A. (2003). Distributed genetic algorithm: Learning by direct exchange of chromosomes. In Springer, (Ed.), *Advances in Artificial Life ECAL'03*, volume 7, pages 346–356.

[Kurokawa et al., 2004] Kurokawa, H., Yoshida, E., Tomita, K., Kamimura, A., S.Murata, and Kokali, S. (2004). Deformable multi m-tran structure works as walker generator. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems*, pages 746–753.

[Lachmann et al., 2000] Lachmann, M., Sella, G., and Jablonka, E. (2000). On information sharing and the evolution of collective. *Proc.Roy.Soc of London Biological Series*, 267(1450):1287–1293.

[Lam and Pochy, 1993] Lam, L. and Pochy, R. (1993). Active-walker models: Growth and form in nonequilibrium systems. *Computer in Physics*, 7(5):534–541.

[Lam et al., 1997] Lam, L., Shu, C.-Q., and Bødefeld, S. (1997). Active walks and path dependent phenomena in social systems.

[Langton, 1989] Langton, C., (Ed.) (1989). *Artificial Life.* MIT Press.

[Lee and Sanderson, 2001] Lee, W. and Sanderson, A. (2001). Dynamic analysis and distributed control of the tetrobot modular reconfigurable robotic system. *Autonomous Robots*, 10:67–82.

[Lerman and Galstyan, 2001] Lerman, K. and Galstyan, A. (2001). A general methodology for mathematical analysis of multi-agent systems. Technical Report ISI-TR-529, USC Information Sciences.

[Lewis and Bekey, 1992] Lewis, A. and Bekey, G. (1992). The behavioral self-organization of nanorobots using local rules. In *IEEE Int.Conf. on Intelligent Robots & Systems*, pages 1333–1338.

[Lewis and Tan, 1997] Lewis, A. and Tan, K.-H. (1997). High precision formation control of mobile robots using virtual structures. *Autonomous Robots*, 4:387–403.

[Lim and Kim, 2001] Lim, H. and Kim, C. (2001). Flooding in wireless ad hoc networks. *Computer Communication*, 24:352–363.

[Lindenmayer, 1968] Lindenmayer, A. (1968). Mathematical models for cellular interaction in development: Parts i and ii. *Jour. of Theoretical Biology*, 18:280–299,300–315.

[Lipson and Pollack, 2000] Lipson, H. and Pollack, J. (2000). Automatic design and manufacture of robotic lifeforms. *Nature*, 406:974–978.

[Lungarella et al., 2002] Lungarella, M., abd R. Pfeifer, V. H., and Yokoi, H. (2002). Wisking: An unexplored sensory modality. In Hallam, B. et al., (Eds.), *From Animals to Animats SAB'7*, volume 7, pages 58–60.

[Madina et al., 2003] Madina, D., Ono, N., and Ikegami, T. (2003). Cellular evolution in a 3d lattice artificial chemistry. In Springer, (Ed.), *Advances in Artificial Life ECAL'03*, volume 7, pages 59–68.

[Maes, 1991] Maes, P. (1991). A bottom-up mechanism for behaviour selection in an artificial creature. In Meyer, J.-A. et al., (Eds.), *From Animals to Animats SAB'1*, volume 1, pages 238–247.

[Maes, 1993] Maes, P. (1993). Behaviour-based artificial intelligence. In Meyer, J.-A. et al., (Eds.), *From Animals to Animats SAB'2*, volume 2, pages 2–10.

[Martinoli et al., 2003] Martinoli, A., Easton, K., and Agassounon, W. (2003). Modeling swarm robotic systems: A case study in collaborative distributed manipulation. *Int. Journal of Robotics Research, Special Issue on Experimental Robotics*, 23(4):415–436.

[Martinoli et al., 1999a] Martinoli, A., Franzi, E., and Matthey, O. (1999a). Towards a reliable set-up for bio-inspired collective experiments with real robots. In Casals, A. and de Almeida, A., (Eds.), *Proc. of the Fifth Int. Symp. on Experimental Robotics ISER-97, Lecture Notes in Control and Information Sciences*, volume 232, pages 597–608.

[Martinoli et al., 1999b] Martinoli, A., Ijspeert, A., and Gambardella, L. (1999b). A probabilistic model for understanding and comparing collective aggregation mechanisms. *Proc. Euro.Conf. on Artificial Life ECAL'99*, pages 575–584.

[Martinoli et al., 1999c] Martinoli, A., Ijspeert, A. J., and Mondada, F. (1999c). Understanding collective aggregation from probabilistic modelling to experiments with real robots. *Special Issue on Distributed Autonomous Robotic Systems, Robotics and Autonomous Systems*, 29:51–63.

[Martinoli and Mondada, 1998] Martinoli, A. and Mondada, F. (1998). Probabilistic modelling of a bio-inspired collective experiment with real robots. In uth, T. L. et al., (Eds.), *Distributed Autonomous Robotic Systems DARS-98*, pages 289–308.

[Mataric, 1992] Mataric, M. (1992). Designing emergent behaviours: From local interactions to collective intelligence. In *From Animals To Animats*, pages 432–441.

[Mataric, 1994] Mataric, M. (1994). Learning to behave socially. In Cliff, D. et al., (Eds.), *From Animals To Animats*, volume 3, pages 453–462.

[Matsushita, 1997] Matsushita, M. (1997). Formation of colony patterns by a bacterial cell population. In Dworkin, M. and Shapiro, J., (Eds.), *Bacteria as Multicellular Organisms*, pages 366–393. Oxford University Press.

[McGeer, 1990] McGeer, T. (1990). Passive dynamical walking. *International Journal of Robotics Research*, 9:62–82.

[Melhuish, 1999a] Melhuish, C. (1999a). Controlling and coordinating mobile micro-robots: Lessons from nature. In *IMEC'99*.

[Melhuish, 1999b] Melhuish, C. (1999b). Employing secondary swarming with small scale robots: a biologically inspired collective approach. In *Proc. of the 2nd Int.Conf. on Climbing & Walking Robots CLAWAR*.

[Melhuish, 1999c] Melhuish, C. (1999c). *Strategies for Collective Minimalist Mobile Robots.* PhD Thesis.

[Melhuish et al., 1998] Melhuish, C., Holland, O., and Hoddell, S. (1998). Collective sorting and segregation in robots with minimal sensing. In *From Animals to Animat*, volume 5, pages 465–470. MIT Press.

[Melhuish et al., 1999a] Melhuish, C., Holland, O., and Hoddell, S. (1999a). Convoying : Using chorusing to form travelling groups of minimal agents. In *Robotics Autonomous Systems*, volume 28, pages 207–216.

[Melhuish et al., 1999b] Melhuish, C., J.Welsby, and C.Edwards (1999b). Using templates for defensive wall building with autonomous mobile ant-like robots. In *Proc. TIMR'99 Towards Intelligent Mobile Robots, Technical Report Series.*

[Mesot and Teuscher, 2003] Mesot, B. and Teuscher, C. (2003). Critical values in asynchronous random boolean networks. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life ECAL'03*, volume 7, pages 367–376.

[Miller, 2003] Miller, J. (2003). Evolving developmental programs for adaptation, morphogenesis and self-repair. In Springer, (Ed.), *Advances in Artificial Life ECAL'03*, volume 7, pages 256–265.

[Mombach and Glazier, 1996] Mombach, J. and Glazier, J. (1996). Single cell motion in aggregates of embryonic cells. *Physical Review Letters*, 16(76):3032–3035.

[Mondada et al., 2004] Mondada, F., Bonani, M., Magnenat, S., Guignard, A., and Floreano, D. (2004). Physical connections and cooperation in swarm robotics. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems.*

[Murata et al., 2000] Murata, S., Yoshida, E., Tomita, K., Kurokawa, H., Kamimura, A., and Kokaji, S. (2000). Hardware design of modular robotic system. In *Proc. of IEEE/RSJ Int.Conf. on Intelligent Robots & Systems IROS'2000.*

[Naffin and Sukhatme, 2004] Naffin, D. and Sukhatme, G. (2004). Negotiated formations. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems*, pages 181–190.

[Nagpal, 1999] Nagpal, R. (1999). Organizing a global coordinate system from local information on an amorphous computer. Technical Report AI memo 1666, MIT.

[Nagpal and Coore, 1998] Nagpal, R. and Coore, D. (1998). An algorithm for group formation in an amorphous computer. In *Proc. of the 10th International Conference on Parallel and Distributed Computing Systems (PDCS'98).*

[Nehaniv et al., 1999] Nehaniv, C., Dautenhahn, K., and Loomes, M. (1999). Constructive biology and approaches to temporal grounding in post-reactive robotics. In McKee, G. and P, S., (Eds.), *Sensor Fusion and Decentralized Control in Robotics Systems II, Proceedings of SPIE*, volume 3839, pages 156–167.

[Nehaniv et al., 2003] Nehaniv, C., Dautenhahn, K., and Newton, A. (2003). The robot in the swarm: an investigation into agent embodiment within virtual robotic swarms. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life, ECAL 2003*, volume 7, pages 829–838.

[Nehmzow, 2001] Nehmzow, U. (2001). Physically embedded genetic algorithm learning in multi-robot scenarios: The pega algorithm. In *Proc. 2nd Int.Work. on Epigenetic Robotics: Modeling Cognitive Development in Robotic Systems.*

[Nembrini et al., 2002] Nembrini, J., Winfield, A., and Melhuish, C. (2002). Minimalist coherent swarming of wireless networked autonomous mobile robots. In *From Animals to Animats SAB'02*, pages 373–382. MIT Press.

[Nicolis and Prigogine, 1977] Nicolis, G. and Prigogine, I. (1977). *Self-Organisation in Non-Equilibrium Systems.* New York, Wiley & Sons.

[Noble, 2000] Noble, J. (2000). Talk is cheap : Evolved strategies for communication and action in asymmetrical animal contests. In Press, M., (Ed.), *SAB 2000*, pages 481–490.

[Nolfi and Floreano, 2000] Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: Biology, Intelligence and Technology of Self-Organising Machines.* MIT Press.

[Nusslein-Volhard, 1996] Nusslein-Volhard, C. (1996). Gradients that organise embryo-development. *Scientific American*, august:38–43.

[Østergaard and Lund, 2004] Østergaard, E. and Lund, H. (2004). Distributed cluster walk for the atron self-reconfigurable robot. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems, IAS-8*, volume 8, pages 291–298.

[Ottery and Hallam, 2004] Ottery, P. and Hallam, J. (2004). Steps towards self-reconfigurable robot systems by modelling cellular adhesion mechanisms. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems, IAS-8*, pages 720–728.

[Papert and Harel, 1991] Papert, S. and Harel, I. (1991). Situating constructionism. In Pub.Corp., A., (Ed.), *Constructionism*, page 1st chapter.

[Parker, 1996] Parker, L. (1996). On the design of behaviour-based multi-robot teams. *Advanced Robotics*, 10(6):547–578.

[Parker, 2000] Parker, L. (2000). Current state of the art in distributed autonomous mobile robotics. In *Distributed Autonomous Robotic Systems*, volume 4, pages 3–12. Springer.

[Parker and Touzet, 2000] Parker, L. and Touzet, C. (2000). Multi-robot learning in a cooperative observation task. In *Distributed Autonomous Robotic Systems*, volume 4, pages 391–401. Springer.

[Penn, 2002] Penn, A. (2002). Steps towards a quantitative analysis of individuality and its maintenance: a case study with multi-agent systems. In Polani, D. et al., (Eds.), *Fifth German Workshop on Artificial Life: Abstracting and Synthesizing the Principles of Living Systems*, pages 125–134.

[Penn, 2003] Penn, A. (2003). Modelling artificial ecosystem selection: A preliminary investigation. In Springer, (Ed.), *Advances in Artificial Life ECAL'03*, volume 7, pages 659–668.

[Pfeifer, 2000] Pfeifer, R. (2000). On the role of morphology and materials in adaptive behaviour. In Meyer, J.-A. et al., (Eds.), *From Animals to Animats 6, SAB2000*, pages 23–32.

[Pfeifer and Scheier, 1999] Pfeifer, R. and Scheier, C. (1999). *Understanding Intelligence*. MIT Press.

[Piaget, 1954] Piaget, J. (1954). *The Construction of Reality in the Child*. Basic Books.

[Pollack et al., 2000] Pollack, J., Lipson, H., Funes, P., Hornby, G., and Watson, R. (2000). Evolutionary techniques in physical robotics. In *Third International Conference on Evolvable Systems: From Biology to Hardware (ICES2000)*.

[Prencipe and Gervasi, 2002] Prencipe, G. and Gervasi, V. (2002). On the intelligent behavior of stupid robots. In *VIII Convegno AI*IA*, Siena.

[Prigogine, 1994] Prigogine, I. (1994). *Les Lois du Chaos.* Champs Flammarion.

[Qi et al., 2001] Qi, H., Iyengar, S., and Chakrabarty, K. (2001). Distributed sensor networks-a review of recent research. *Jour. of Franklin Institute*, 338:655–668.

[Quick et al., 1999a] Quick, T., Dautenhahn, K., Nehaniv, C., and Roberts, G. (1999a). The essence of embodiment : a framework for understanding and exploiting structural coupling between system and environment. In *3rd Int.Conf. on Computer Anticipatory Systems.*

[Quick et al., 1999b] Quick, T., Nehaniv, C., Dautenhahn, K., and Roberts, G. (1999b). Evolving embodied genetic regulatory network-driven control systems. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life ECAL'03*, volume 7, pages 366–377.

[Quinn et al., 2002a] Quinn, M., Smith, L., Mayley, G., and Husbands, P. (2002a). Evolving formation movement for a homogeneous multi-robot system: Teamwork and roe-allocation with real robots. In *Cognitive Science Research Papers*, volume 515.

[Quinn et al., 2002b] Quinn, M., Smith, L., Mayley, G., and Husbands, P. (2002b). Evolving team behaviour for real robots. In *Walter Grey Walter International Workshop on Biologically-Inspired Robotics*, pages 217–224.

[Ray, 1992] Ray, T. (1992). An approach to the synthesis of life. *ALife*, II:371–408.

[Resnick, 1999] Resnick, M. (1999). *Turtles,Termites and Traffic Jams : Explorations in Massively Parrallel Microworlds.* MIT Press.

[Reynolds, 1987] Reynolds, C. (1987). Flocks, herds and schools : a distributed behavioral model. In *Computer Graphics*, volume 21, pages 25–34.

[Ronald and Sipper, 2001] Ronald, E. and Sipper, M. (2001). Surprise versus unsurprise: Implications of emergence in robotics. *Robotics and Autonomous Systems*, 37(1):19–24.

[Savill and Hogeweg, 1997] Savill, N. and Hogeweg, P. (1997). Modelling morphogenesis: From single cells to crawling slugs. *J. of Theor. Biol.*, 184:229–235.

[Schreiber, 1999] Schreiber, T. (1999). Interdisciplinary application of non-linear time series methods. *Physics Reports*, 308(2).

[Shapiro, 1988] Shapiro, J. (1988). Bacteria as a multicellular organism. *Scientific American*, June 1988:62–69.

[Shapiro and Dworkin, 1997] Shapiro, J. and Dworkin, M., (Eds.) (1997). *Bacteria as a Multicellular Organism*. Oxford University Press.

[Sims, 1994] Sims, K. (1994). Evolving 3d morphology and behavior by competition. *Artificial Life VI*, pages 28–39.

[Sipper, 1997] Sipper, M. (1997). *Evolution of Parallel Cellular Machines: The Cellular Programming Approach*. Springer.

[Stanley and Miikkulainen, 2003] Stanley, K. and Miikkulainen, R. (2003). A taxonomy for artificial embryogeny. *Artificial Life*, 9:93–130.

[Støy, 2001a] Støy, K. (2001a). Developing a solution to the foraging task using multiple robots and local comunication. In *IEEE CIRA2001*.

[Støy, 2001b] Støy, K. (2001b). Sound localization using intensity in mobile autonomous robotics. In *7th Scandinavian Conf. on AI*.

[Støy, 2001c] Støy, K. (2001c). Using situated communication in distributed autonomous mobile robotics. In *7th Scandinavian Conf. on AI*, pages 44–52.

[Støy, 2004] Støy, K. (2004). Controlling self-reconfiguration using cellular automata and gradients. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems, IAS-8*, pages 693–702.

[Støy et al., 2002] Støy, K., Shen, W.-M., and Will, P. (2002). Global locomotion from local interaction in self-reconfigurable robots. In *7th Int.Conf. on Intelligent & Autonomous Systems (IAS7)*, pages 309–316.

[Streichert et al., 2003] Streichert, F., Spieth, C., Ulmer, H., and Zell, A. (2003). Evolving the ability of limited growth and self-repair for artificial embryos. In Banzhaf, W. et al., (Eds.), *Advances in Artificial Life, ECAL'03*, volume 7, pages 289–298.

[Sugihara and Suzuki, 1990] Sugihara, K. and Suzuki, I. (1990). Distributed motion coordination of multiple mobile robots. In *5th IEEE Int. Symposium on intelligent Control*, pages 138–143.

[Suzuki and Yamashita, 1994] Suzuki, I. and Yamashita, M. (1994). Distributed anonymous mobile robots - formation and agreement problems. Technical Report TR-94-07-01, Department of Electrical Engineering and Computer Science, University of Wisconsin, Milwaukee.

[Takahashi et al., 2001] Takahashi, N., Nagai, T., Yokoi, H., and Kakazu, Y. (2001). Control system of flexible structure multi-cell robot using amoeboid self-organisation mode. In *Proc.6th Europ.Conf. on Artificial Life, ECAL 2001*, pages 563–572. Springer.

[Takahashi et al., 2004] Takahashi, N., Yu, W., Yokoi, H., and Kakazu, Y. (2004). Amoeba like multi-cell robot control system. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems, IAS-8*. IOS Press.

[Tangamchit et al., 2000] Tangamchit, P., Dolan, J., and Khosla, P. (2000). Learning-based task allocation in decentralized multi-robot systems. In *Distributed Autonomous Robotic Systems*, volume IV, pages 381–390 and 483–484.

[Taylor, 2004] Taylor, T. (2004). A genetic regulatory network-ispired real-time controller for a group of underwater robots. In Groen, P. et al., (Eds.), *Proc. Int.Conf. on Intelligent & Autonomous Systems, IAS-8*, pages 403–412.

[Toffoli, 2000] Toffoli, T. (2000). What you always wanted to know about genetic algorithms but were afraid to hear. In *Festschrift Conference in Honor of J.H. Holland.*

[Trianni et al., 2002] Trianni, V., Labella, T., R.Gross, Sahin, E., Dorigo, M., and Deneubourg, J.-L. (2002). Modelling pattern formation in a swarm of self-assembling robots. Technical Report TR/IRIDIA/2002-12, IRIDIA.

[Trianni et al., 2004] Trianni, V., Nolfi, S., and Dorigo, M. (2004). Hole avoidance: Experiments in coordinated motion on rough terrain. In *Proc. Int.Conf on Intelligent & Autonomous Systems.*

[Turing, 1952] Turing, A. (1952). The chemical basis of morphogenesis. *Phil.Trans.Roy.Soc. series B: Biological Sciences*, 237:37–72.

[Turkle and Papert, 1992] Turkle, S. and Papert, S. (1992). Epistemological pluralism and the reevaluation of the concrete. *Journal of Mathematical Behavior*, 11(1):3–3.

[Unsal et al., 2000] Unsal, C., Kilivcote, H., Patton, M., and Khosla, P. (2000). Motion planning for a modular self-reconfiguring robotic system. In *Distributed Autonomous Robotic Systems*, volume IV, pages 165–175. Springer.

[Vaughan et al., 2000a] Vaughan, R., Støy, K., Sukhatme, G., and Mataric, M. (2000a). Blazing a trail : Insect inspired resource transportation by a robot team. In *Distributed Autonomous Robotic Systems*, volume IV, pages 111–120.

[Vaughan et al., 2000b] Vaughan, R., Sukhatme, G., Mesa-Martinez, F., and Montgomery, J. (2000b). Fly spy : Lightweight localization and target tracking for cooperative air and ground robots. In *Distributed Autonomous Robotic Systems*, volume IV, pages 315–324.

[von Glasersfeld, 1995] von Glasersfeld, E. (1995). *Radical Constructivism - A Way of Knowing and Learning*. Falmer Press.

[Wang and Premvuti, 1994] Wang, J. and Premvuti, S. (1994). Resource sharing in distributed robotic systems based on a wireless medium access protocol (csma/cd-w). In *IEEE RSJ IROS*, pages 784–791.

[Wang et al., 1995] Wang, J., Premvuti, S., and Tabbara, A. (1995). A wireless medium access protocol (csma/cd-w) for mobile robot based distributed robotic system. In *IEEE Int. Conf. on Robotics & Automation*, pages 2561–2566.

[Ward and Mellor, 1985] Ward, P. and Mellor, S., (Eds.) (1985). *Structured Development for Real-Time Systems*. PrenticeHall/Yourdon Press.

[Watson et al., 1999] Watson, R., Ficici, S., and Pollack, J. (1999). Embodied evolution: A response to challenges in evolutionary robotics. In *8th Europ.Workshop on Learning Robots*, pages 14–22.

[Watson et al., 2002] Watson, R., Ficici, S., and Pollack, J. (2002). Embodied evolution : Embodying an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1):1–18.

[Watts, 1999] Watts, D. (1999). *Small Worlds - the Dynamics of Networks beween Order and Randomness*. Princeton Univ. Press.

[Webb et al., 2003] Webb, B., Reeve, R., A.Horchler, and Quinn, R. (2003). Testing a model of cricket phonotaxis on an outdoor robot platform. In *Proc. of TIMR'03*.

[Weiss and Knight, 2000] Weiss, R. and Knight, T. (2000). Engineered communications for microbial robotics. In *Proceedings of the Sixth International Meeting on DNA Based Computers, DNA'6*.

[Weßnitzer, 2001] Weßnitzer, J. (2001). Distributed computation in networks of mobile robots. Technical report, University of the West of England, IASLab.

[Weßnitzer et al., 2001] Weßnitzer, J., Adamatzky, A., and Melhuish, C. (2001). Towards self-organising structure formations: A decentralised approach. In *Proceedings of ECAL 2001*, pages 573–581. Springer.

[Weßnitzer and Melhuish, 2003] Weßnitzer, J. and Melhuish, C. (2003). Collective decision-making and behaviour transitions in distributed ad hoc wireless networks of mobile robots: Target hunting. In *Advances in Artificial Life ECAL 2003*, pages 893–902. Springer.

[Winfield, 2000] Winfield, A. (2000). Distributed sensing and data collection via broken ad hoc wireless connected networks of mobile robots. In *Distributed Autonomous Robotic Systems*, volume IV, pages 273–282.

[Winfield, 2004] Winfield, A. (2004). Towards dependable swarms. In *Swarm Robotics Workshop, SAB'04*, Santa Monica.

[Winfield and Holland, 2000] Winfield, A. and Holland, O. (2000). The application of wireless local area network technology to the control of mobile robots. *Microprocessors and Microsystems*, 23:597–607.

[Wright et al., 2001] Wright, W., Smith, R., Danek, M., and Greenway, P. (2001). A generalisable measure of self-organisation and emergence. In Dorffner, G. et al., (Eds.), *Artificial Neural Networks - ICANN 2001*, pages 857–864.

[Wuensche, 1998] Wuensche, A. (1998). Discrete dynamical networks and their attractor basins. Technical Report SFI 98-11-101, Santa Fe Institute.

[Yamaguchi et al., 2001] Yamaguchi, H., Arai, T., and Beni, G. (2001). A distributed control scheme for multiple robotic vehicle to make group formation. *Robotics and Autonomous Systems*, 36:125–147.

[Yanco and Stein, 1992] Yanco, H. and Stein, L. (1992). An adaptive communication protocol for cooperating mobile robots. In *From Animals to Animats SAB'92*, pages 478–485.

[Yokoi et al., 1998] Yokoi, H., Yu, W., Hakura, J., and Kakazu, Y. (1998). Morpho-functional machine: Robotics approach of amoeba model based on vibrating potential method. In Pfeifer, R. et al., (Eds.), *Proceedings of 5th Int.Conf. on Simulation of Adaptive Behaviour: From Animals to Animats.*

[Yoshida et al., 1998] Yoshida, E., Arai, T., Yamamoto, M., and Ota, J. (1998). Local communication of multiple mobile robots : Design of optimal communication area for cooperative tasks. *Journal of Robotic Systems*, 15(7):407–419.

[Yoshida et al., 2000] Yoshida, E., Murata, S., Kokaji, S., Tomita, K., and Kurokawa, H. (2000). Micro self-reconfigurable robotic systems using shape memory alloy. In *Distributed Autonomous Robotic System*, volume IV, pages 145–154.

[Ziemke, 2001] Ziemke, T. (2001). The construction of reality in the robot: Constructivist perspectives on situated artificial intelligence and adaptive robotics. *Foundations of Science: Radical Constructivism.*