Evolving Communication in Evolutionary Robotics

Joachim de Greeff

Master's thesis

June 2007

Cognitive Artificial Intelligence
Utrecht University

First supervisor: dr. Marco Wiering
Second supervisor: dr. Stefano Nolfi
Third supervisor: dr. Frank Dignum

Contents

Contents i

Abstract iii

Acknowledgments v

1 Introduction 1
- 1.1 Outline 1
- 1.2 Background 1
- 1.3 Research Questions 2
- 1.4 Methodology 3
- 1.5 Relation to Cognitive Artificial Intelligence 4
- 1.6 Terminology 5
- 1.7 Thesis Outline 6

2 Evolutionary Robotics 7
- 2.1 Outline 7
- 2.2 Historical Overview 7
- 2.3 Motivations for Doing ER 9
  - 2.3.1 Design by Hand versus Automatic Design of Control Programs 9
- 2.4 Technique 10
  - 2.4.1 Control and Evolutionary Techniques 10
  - 2.4.2 Artificial Neural Networks 10
  - 2.4.3 Classifier Systems 12
  - 2.4.4 Genetic Algorithms 12
  - 2.4.5 Genetic Programming 13
- 2.5 Environments 13
  - 2.5.1 Different Environments 13
  - 2.5.2 Simulators 14
  - 2.5.3 Hybrid Approaches 14
  - 2.5.4 Embodied Evolution 14
- 2.6 Summary 15

3 EvoRobot 17
- 3.1 Outline 17
- 3.2 Simulators 17
- 3.3 Features of EvoRobot 18

3.3.1 The World . . . 18
3.3.2 The Robots . . . 18
3.3.3 The Neural Network Controller . . . 20
3.3.4 The Simulation Process . . . 21
3.3.5 The Evolutionary Process . . . 21
3.4 Modifications Made in EvoRobot . . . 23
3.5 Using EvoRobot with Real Robots . . . 23
3.6 Summary . . . 24

4 Evolution of Communication 25
4.1 Outline . . . 25
4.2 Motivations for Studying EoC through ER . . . 25
4.3 About Natural Communication Systems . . . 26
4.4 About Evolving Artificial Communication . . . 27
4.5 The Characteristics of a Suitable Task . . . 28
4.6 A Model of Communication Prerequisites . . . 29
4.7 Forms and Modalities of Communication . . . 31
4.8 Summary . . . 32

5 Experiments 33
5.1 Outline . . . 33
5.2 First Series of Experiments . . . 33
5.2.1 Experimental Setup . . . 33
5.2.2 Results . . . 37
5.3 Second Series of Experiments . . . 45
5.3.1 Experimental Setup . . . 45
5.3.2 Results . . . 47
5.3.3 Concluding Remarks . . . 56
5.4 Summary . . . 56

6 Application in e-Puck Robots 57
6.1 Outline . . . 57
6.2 The e-Puck Robot . . . 57
6.3 Transfer from EvoRobot to e-Puck . . . 58
6.3.1 Two Approaches . . . 58
6.3.2 Modifications in EvoRobot . . . 59
6.3.3 Modifications in SerCom . . . 60
6.4 Performance of the e-Pucks . . . 61
6.4.1 Two Evolved Behaviors Tested . . . 61
6.4.2 Problematic Issues . . . 61
6.5 Summary . . . 62

7 Conclusions 63
7.1 Summary . . . 63
7.2 Discussion and Future Research . . . 64

Bibliography 65

Abstract

In order to solve a collective task, a team of mobile autonomous robots needs to communicate with each other and with the environment. The creation of suitable control programs for these robots through evolutionary search requires knowledge about the prerequisites and environmental factors which drive the evolutionary process toward the development of effective communication. Since a lot of key features about the nature of the task, the capacities of the robots and the evolutionary forces are relatively unknown, it is not trivial which characteristics of a task stimulate a group of robots to develop a functional communication system. To gain insight in these questions a model is proposed which embodies some of the (presumably) key aspects that influence the development of communication. From this model a series of experimental setups is derived which is tested in a simulator (EvoRobot). Results show that it is hard to isolate different task dimensions because the presence of one aspect influences the usefulness of communication in relation to the other aspects. Nevertheless, it can be concluded that the most influential aspect that boosts the use of communication is the possibility for robots to have access to information which is useful for other robots. Next to this, the aim of this thesis is to investigate the characteristics of evolved communication systems and the level of complexity that can be achieved. Thus, various aspects of complexity on both the task setup and the resulting communication system are discussed and based on this a more complex task is developed. This results in a rich communication system which includes selective attention, different functional roles and integration of multiple communication channels. In order to strengthen findings in simulation, evolved behaviors are then transferred onto e-Puck robots.

Acknowledgments

First of all I would like to thank my supervisors. Marco Wiering for the discussions about the contents and progression of this thesis, Stefano Nolfi for the numerous good ideas and critical reflections on the experiments conducted and Frank Dignum for proofreading and some critical remarks. Thanks to Marieke for her love and support. I also would like to thank Bas for the stimulating discussions about a lot of things and my friends and family for support and understanding.

Joachim
2007

“  ”

Robot R1, 2nd series of experiments, replica 4, epoch 0, cycle 801, 2007

Chapter 1

Introduction

1.1 Outline

The study of Artificial Intelligence investigates among others how robots can become 'intelligent'. With intelligent is meant that robots are able to perform all sorts of duties and participate in society. Because robots may need to cooperate with both humans and other robots, they should also be able to communicate. While talking and understanding natural language is something which robots cannot do (yet), they may use their own language in order to communicate with other robots. Because it is rather hard to think of every possible situation in which robots might find themselves it is very impractical to pre-program all robot behavior, including communication, in advance. Hence, within Artificial Intelligence techniques are developed which allow for automatic generating robot behavior. One of these techniques is the use of evolutionary forces to develop advanced robot control programs. By applying a 'survival of the fittest' mechanism on a group of robots, their performance can increase because only the best performing robots are allowed to survive. These are then used as a basis for a new generation on which a selection is applied again. Such an approach should also be able to develop a communication system which robots may apply to solve tasks that require cooperation. This thesis investigates how such an evolutionary system can be used to develop a functional communication system.

In this chapter a background is sketched first, along with the formulation of three research questions. Then the methodology is discussed, as well as the relation of research conducted in this thesis to the study of Cognitive Artificial Intelligence as taught at Utrecht University. Last, some key concepts are explained.

1.2 Background

Artificial Intelligence is a relatively young discipline but has developed into a mature research enterprise since the pioneer work in the 1950's. Emphasis has shifted from an all-including artificial model of human cognition, commonly referred to as 'GOFAI'[1], inspired by the 'Language of Thought' hypotheses (Fodor and Pylyshyn, 1988) toward a perspective that emphasizes the importance of embodiment and situatedness (Brooks, 1991; Clark, 1996)[2] and

[1]Good Old-Fashioned Artificial Intelligence; term introduced by Haugeland (1985).

[2]An artificial system is always realized in a physical or virtual body (embodiment) and is always placed in an environment with which it interacts (situatedness).

adaptivity (Nolfi and Floreano, 2000)[3]. In this new paradigm, commonly referred to as 'Embodied Cognition' (EC), cognition is viewed as a dynamical process happening in a body interacting with its environment. Inspiration was taken from biology, neuroscience and complex system theory. It was hypothesized that EC could produce artificial systems which are more adaptive and reactive toward their environment than those resulting from the GOFAI approach.

In the wake of EC, the field of Evolutionary Robotics (ER) matured. ER advocates the same line of thought in the sense that it strives to create physically embodied robots functioning in an ever changing environment. Next to this, emphasis is on adaptivity which is realized by exploiting evolutionary techniques. By subjecting suitable control architectures like neural networks to evolutionary search the aim is to obtain well performing robotic controllers which are adapted toward their environment. This can be done for a individual robot but also for a team of robots which need to cooperate. In order to have multiple robots operate collectively in an effective manner there is a need to communicate. Thus, recent interest within ER has also focused on the development of teams of robots which are able to function in real world environments and are capable of solving collective tasks. This type of robot needs to communicate with its surroundings and with other agents in order to maintain itself and solve a given task in an efficient manner. For this purpose research has been conducted to investigate artificial communication systems. Again, inspiration can be found in biology. Theory about the evolution of natural communication and various types of animal communication can help the development of artificial communication systems. Because this research is conducted within the realm of ER there is a need to understand how evolutionary principles can be used effectively to obtain robots which use communication. This requires knowledge about the prerequisites and environmental factors which drive an evolutionary process toward the development of an effective communication system.

While considerable work has been done, a lot of key features about the nature of the task the robots are faced with, the capacities of the robots and the evolutionary forces are not properly understood yet. For instance, it is not trivial which characteristics of a task stimulate a group of robots to develop a functional communication system. This thesis hopes to contribute to the insight of how communication within a team of robots can be obtained and what it is capable of.

1.3 Research Questions

As described, many aspects of the evolution of communication in ER are not properly understood yet. Regarding the use of ER to simulate the evolution of communication one of the most basic questions is what the capacities of ER are to simulate communication and how this can be achieved. Because robot control systems resulting from ER techniques are heavily shaped toward the task for which they are evolved it is important to understand the characteristics of a task which may cause the development of communication. Furthermore, it would be interesting to gain insight in the type and capacities of communication which can be evolved. Next, to strengthen findings in simulation, it would be valuable to transfer evolved behaviors onto real robots.

[3] An artificial system should be adaptive toward its environment in order to operate effectively.

Thus, three research questions can be formulated:

What is the nature of tasks which may trigger the development of communication?

And based on insights from the first research question:

What are the characteristics of evolved communication systems and what level of complexity can be achieved?

Furthermore, in order to confirm findings in simulation and to do justice to the emphasis of EC on embodiment and situatedness, a third research question is:

Can behaviors developed in simulation be transferred onto real robots?

1.4 Methodology

Research within Artificial Intelligence and especially ER quite often makes use of computational simulations. This stems from the fact that the phenomena to be studied are sometimes not observable in nature in a practical manner. For instance, different stages in an evolutionary process related to certain capacities of natural species might be deductible from fossil evidence, but this evidence may very well be rather poor. A computer model simulating this evolutionary process could make up for the lack of natural evidence. Of course, findings from a computer model can never provide a complete alternative for natural evidence, but they may nevertheless provide valuable insights which would otherwise be hard to obtain. Also, various implementations and parameter settings can be tested efficiently for their influence on the phenomenon which is studied. One needs to be careful however, not to construct within the model precisely that which is wished to be observed. Because a model is always a simplification of reality certain assumptions and presuppositions are inevitably, but these should be as plausible as possible and not bias the model toward a specific direction[4].

Conducting proper research using simulations has been the subject of debate. Di Paolo (1999) proposed three different phases for conducting proper research using simulations. First, there is an exploratory phase in which various interesting cases can be examined, observables be formulated and patterns be explored. Then there is an experimental phase in which hypotheses are formulated, crucial experiments are tested and the simulation is explained. Last, there is an explanatory phase in which the phenomena observed in simulation are related to existing theories about natural phenomena which motivated the construction of the model in the first place. The obtained experimental data may have the same status as a

[4]This is trivial of course, as one always tries to have a model as close to reality as possible. Nevertheless it is not always straightforward in practice, since sometimes implementation shortcuts are used which may seem natural but in fact do bias the model.

'thought experiment',[5] with the difference that evidence from simulation may be more solid than mental journeys alone since it is easier to constrain computer simulations in a plausible manner.

Next to this, simulations may also provide a 'proof of concept', which does not necessarily entails a comprehensive theory for certain natural phenomena but proposes a possible explanation of how such phenomena could be. This can function as an illustration of a possible solution and as inspiration to engage in more thorough theory building.

In order to answer the first research question a model is proposed which embodies some of the (presumably) key aspects which influence the development of communication. These key aspects are based on a review of the current literature on the evolution of communication in ER. From this model a series of experimental setups is derived which is tested in ER fashion: robots placed in an environment are evolved for their ability to solve a given task which embodies the key aspects of the proposed model. The resulting evolved behaviors are then analyzed for the functional role of communication and compared with non-communication solutions for the same task.

Based on findings of the first series of experiments, an advanced task is designed which is meant to test whether an advanced level of complexity can be achieved in an evolved communication system. Different aspects of complexity are discussed and a series of experiments based on this more advanced task is conducted. The second research question is addressed by analyzing two of the most interesting replicas from this series of experiments. A description is given of the functionality of different evolved signals and how these signals are grounded in sensory-motor experiences.

The third research question is addressed by transferring the resulting evolved neural controllers of the second research question onto e-Puck robots (E-Puck robot development group, 2007).

1.5 Relation to Cognitive Artificial Intelligence

The study of Cognitive Artificial Intelligence (CAI) as taught at Utrecht University is comprised of multiple research areas which all contribute to the development of intelligent artificial systems. Motivations and objectives to build such systems are on the one hand the wish to model natural cognitive systems as encountered in animals and humans in order to understand them, and on the other hand the wish to develop practical applications in (robotic) software which may be applied in various branches. Thus, one line of research tries to understand natural cognitive systems by making use of AI techniques and the other line of research uses natural systems as inspiration for developing better AI systems.

This twofoldness of CAI in general applies to ER as well. On the one hand we can test various models of natural cognitive systems by observing how well they perform in artificial systems. Since there is full knowledge of the inner workings of these artificial systems as a test ground (after all, we developed them ourselves), such experiments may help to better understand a natural system which might not be very accessible in itself. On the other hand we would like to have functional robots which 'understand' the natural world and society well enough to perform all sorts of duties. Since humans have demonstrated for ages that they

[5] A thought experiment is a hypothetical consideration which is meant to prove a certain point. Because no empirical experiment is conducted, the thought experiment needs to be formulated as plausible as possible in order to be convincing.

can understand the natural world to various degrees, they can serve as source of inspiration to develop artificial systems with equal or better[6] capacities.

Further down the line, the same twofoldness can be observed with respect to the evolution of communication. On the one hand research of the evolution of communication can help to understand how communication systems in nature developed and on the other hand nature again is used as a source of inspiration to develop functional communication in artificial systems.

Next to this motivational twofoldness, which can be observed in all layers of the research enterprise undertaken in this thesis, studying evolution of communication through ER also touches many of the subfields of CAI: it uses theory from philosophy, psychology and biology, techniques from computer science and robotics and it briefly flirts with linguistics as well[7]. Therefore, by embracing comparable motivations and operating within the realm of nearly all subfields of CAI, studying evolution of communication through ER may be called representative for CAI.

1.6 Terminology

In order to clarify some of the non-trivial terms used in this thesis their intended meaning is discussed here.

- *agent:* an agent is viewed as a system that is placed in an environment with which it interacts. In general an agent may be a human, an animal, a software program or physical robot. In this thesis hardware agents are usually called 'robots'. The term 'agent' may still include robots, but emphasis is meant to be on the interactive and situated aspects, rather than on the material the agent is made of.

- *signal:* a signal is an action of an agent which affects the behavior of another agent. The medium over which the signal is conveyed may be arbitrary, so a signal can either consist of a body movement which is perceived by another agent or a 'sound signal'. With sound signal is meant the value of a neural network output unit of an evolved robot controller which is received by other robots and affects their behavior. This signal can be viewed analogue to a sound signal, but is in essence a real number in the range from 0.0 to 1.0. A signal is viewed as explicit when it consists of behavior exclusively serving as a signal, or implicit when it is behavior not exclusively serving as a signal[8].

- *complexity:* complexity is used to describe the characteristics of both a task and an evolved communication system. A task is conceived as more complex when it cannot

[6]Because of various ethical considerations it is always tricky to consider artificial systems which perform 'better' than humans. Mankind seems to possess a somewhat remarkable eagerness to compare himself and compete with machinery, as long as there is the possibility to win the battle. Man-machine chess games for instance are considered highly interesting, while man-machine weight lifting matches would be considered foolish by most people even though in both type of games the machines mostly rely on brute force strategies.

[7]It needs to be noted that linguistics in general is about the properties of human language, whereas the communication signals of natural and artificial systems discussed in this thesis are usually much more simple: they usually do not exhibit language-like features, with some notable exceptions in higher order animal communication.

[8]A behavior executed by an agent might consist of remaining on a target area which serves the agent itself, while at the same time this stopping behavior can signal the location of the target area to another agent. Hence, this is viewed as an implicit signal because it is realized through another behavior.

be solved in a simple manner, i.e. robots have to rely on more than one individual behavior. A communication system is conceived as more complex when it is rich in nature, i.e. it consists of different signals conveying different information and producing different behaviors. A comprehensive list of both task and signal complexity is given in Chapter 4.

1.7 Thesis Outline

The outline of this thesis is as follows. In this 1^{st} chapter a background was sketched, along with the postulation of research questions, a description of the methodology and a justification of the relevance to CAI.

The 2^{nd} chapter consists of a theoretical description of the field of ER, which is used as a tool and as a source of inspiration. It includes a historical background, motivations and discussion about different techniques and environments.

Chapter 3 describes the details of EvoRobot as a simulation program for ER and how it can be used for experiments. It contains a detailed description about the simulated environment and robots, as well as the characteristics of the evolutionary process.

In the 4^{th} chapter various aspects of the evolution of communication are addressed. This includes a discussion about the evolution of natural communication systems, the characteristics of a suitable task, the introduction of a task space model of some prerequisites for communication and the description of a sophisticated task which triggers the development of a complex communication system.

In chapter 5 the experimental setups are discussed along with the results. It also contains an analysis of the effectiveness of the task space model and the nature of the more complex communication systems evolved.

Chapter 6 describes the results of the transfer of the neural controllers evolved in simulation onto e-Puck robots, along with some technical aspects and problems encountered.

Chapter 7 consist of a summary of the work conducted and a discussion about future research.

Chapter 2

Evolutionary Robotics

2.1 Outline

This chapter provides an overview of the field of Evolutionary Robotics. ER is a relatively new field in Artificial Intelligence in which the main idea is to create adaptive control programs and sometimes hardware for robots by exploiting evolutionary principles. Such a control program can consist of a neural network, which takes sensor information from the robot's sensors as input and produces motor actions to be executed by the robot as output. The free parameters of the control program that are subjected to variation and selection typically include the connection weights and biases of the neural network and, possibly, other parameters that determine the architecture of the neural network or the characteristics of the robots' body and sensory-motor system. Advantages are the fact that a large search space with many dimensions can be searched in an effective manner and often solutions are found that are intuitively not so straightforward (from a human perspective) but are nevertheless effective. Drawbacks are the fact that a neural network controller functions essentially as a black box and the fact that training can be very time consuming, especially when hardware is also subjected to the evolutionary process.

2.2 Historical Overview

The principals of ER were first mentioned by Alan Turing who spoke about building brain-like networks through genetical search. However, for decades thereafter research tended to focus more on the GOFAI approach or computational view. In this view a robot or agent is controlled by a program which typically consists of a 'sense-plan-act' loop which is executed in serial. First the environment is perceived through the sensors, then an internal map is built based on these measured values and finally, based on preprogrammed objectives, the most suitable action is selected and executed. Such a system is conceived as a symbol manipulating entity, inspired by a philosophical doctrine which views human cognition as a logical process of symbol manipulation. A typical example is the robot Shakey[9](Nilsson, 1984). Such robots would perform rather well in a sterile and predictable environment. Their performance was less effective in an unpredictable dynamic environment (e.g. an office setting), but most of these deficiencies were judged to be caused by a lack of computational power and should eventually be overcome.

[9]This robot was nicknamed 'Shakey' because of its peculiar way of moving.

It was assumed that cognitive tasks could be decomposed in smaller subtasks (sense, plan and act) which could be solved independently in a heuristic fashion. So in order to solve a rather complex task, all a robot needed to do was to decompose it in manageable subtasks, find solutions for them and 'glue' it all together. The decomposition into subtasks however, is not always trivial. Also, issues with the subtasks themselves were problematic. For instance, how could sensory information be translated into a proper symbolic representation, how could discrete symbols be translated into continuous motor actions and how could planning happen on the fly (to anticipate a changing environment) while it mostly required a lot of time. Nevertheless this computational view dominated AI research for decades.

It was only when in the 1980's Brooks (1986) proposed his radical new view of the 'subsumption architecture' that these underlying assumptions of GOFAI were challenged. The subsumption architecture consisted of multiple layers of behavior which operated in parallel. Sensors where directly connected to motor outputs to allow for fast low level behaviors like obstacle avoidance. On top of this other layers of behavior could be added incrementally which could influence lower behavior but also incorporated higher order objectives. By employing parallel processing of all layers, the robot controller was postulated to be more reactive and adaptive toward changing environments than its GOFAI competitor.

This led to a broader movement, known as Embodied Cognition, which propagated the notion that computational, rule based systems were just not suitable to capture the most basic characteristics of intelligence. Alternatives proposed a sense of 'embeddedness' and 'embodiment' as being more proper views to achieve genuine intelligent systems. Thus, the environment and the physical robot cannot be seen independently but should be studied as a whole. The subsumption architecture was meant to capture this. Behavioral layers were hierarchically structured, where the simplest layers at the bottom would serve as reactive behavior (e.g. obstacle avoidance) and more advanced layers which incorporated the higher order objectives could inhibit or exhibit the output of lower layers and hence strive for more long term goals. Although the subsumption architecture was inspired by nature (low level behavior as 'reflexes', high level reasoning as higher cognition), each layer was still designed and tested by human programmers. Since each layer could influence all pre-existing layers complexity accumulated rapidly. This made the design of programs with a lot of layers rather impractical. It was acknowledged by various authors that the forming of a hierarchical structure of different behaviors should rely more on a self-organizing process which is adaptive to the environment and less on intuition of human programmers.

Even though the subsumption architecture was not the ultimate answer to the problems of the GOFAI approach, Brooks cleared the path for an alternative line of thought. The view was broadened and researchers would focus on biological inspired techniques like neural networks and evolutionary algorithms because such techniques provided the means for the development of adaptive control programs. Also methodological tools of dynamic system theory served the newly formed paradigm well. Dynamic system theory describes the development of typically large systems over time. In this view the robot control program, the robot phenotype[10] and the environment are all regarded as one dynamic system in which interactions take place continuously.

[10]The robots body; simulated or real.

2.3 Motivations for Doing ER

Different motivations give rise to the study of ER. One of them is the wish to study cognition on a minimal level. Since (sophisticated) human cognition is largely built upon pre-existing primitive forms similar to animal cognition it makes sense to start with such basic forms of cognition (Harvey et al., 2005). ER provides a platform for employing such a bottom-up approach in contrast to the top-down GOFAI.

Another motivation is the wish to develop robots which are robust toward an ever changing dynamical environment. Explicit programming and modeling suffers from the 'Frame Problem', which was introduced by McCarthy and Hayes (1969) as a logical problem but later taken into broader context as a general relevance problem for artificial systems which try to maintain knowledge about a changing environment[11]. In contrast, ER emphasizes the significance of the environment. This is achieved by letting robots evolve *within* a world and thus relying on interactions between the robot and the environment, rather than defining the control program in isolation on paper first and testing it in a real world setting later (only to find it does not function very well). By taking a dynamic environment into account a robot can be evolved to be more robust to unpredictable changes.

2.3.1 Design by Hand versus Automatic Design of Control Programs

The relation between design by hand and automatic design needs to be taken into consideration. Hand designing a robot control program can be conceived as specifying a task in an algorithmic fashion, analogue to human reasoning. To be effective, all situations which can possibly be encountered need to be anticipated. Subtasks of the algorithm are to be tackled in the same fashion. To not explicitly describe on an electrical circuit level every step of the algorithm (it would most likely be undoable) some abstraction is required. This can be achieved by describing the algorithm in terms of higher order objectives. The robot should then decompose the objective in parts relying on pre-programmed knowledge about these subparts. In theory this is possible but very impractical in a realistic dynamic environment, simply because it is almost impossible to adequately anticipate every possible situation in which a robot can find itself.

Thus, various automation techniques were put forth which allow for the automatic development of a robot control system. It needs to be noted though, that an automatic design mechanism is not a replacement for specifying what exactly a robot needs to do. It merely constitutes a shift in abstraction. That is to say, where a design-by-hand requires explicit descriptions of every subtask (or references to pre-programmed behaviors, which need to be explicitly designed as well), an automatic design mechanism allows for an autonomous translation (implementation) from higher order objectives to low level implementation. The higher order objective however, still needs to be specified by a programmer.

For instance, in Genetic Algorithms, much depends on carefully choosing a fitness function because the evolved solutions will be shaped toward this. Likewise in Genetic Programming, not only the fitness function is of importance but also the choice of proper function primitives

[11]The relevance problem consist of the fact that it is very hard for an artificial system to determine which of all the perceivable features of the world are relevant for a specific problem. Because not *everything* in the world can be considered (too time consuming), only the relevant aspects should be taken into consideration. However, how can a artificial system know which aspects are relevant for its current problem without examining them first?

and terminal values[12]. It can be very difficult to come up with a proper fitness function because in general there is no heuristic approach toward this. A proper fitness function should reflect the task for which the controllers are trained. It needs to be specific enough to 'point' toward the right direction in which the controller should evolve, but one should refrain from explicitly rewarding certain behavior which from a human perspective seems effective but could prevent the development of truly innovative solutions.

So, although such automation techniques do not provide the whole package from a 'perform well' requirement in general toward low level implementation of robot controllers, they can be very effective and produce powerful solutions to complex problems which are typically encountered within ER.

2.4 Technique

2.4.1 Control and Evolutionary Techniques

Different techniques used within ER can roughly be divided in two categories: control and evolutionary. Control mechanisms are the actual control programs which run on the robot. Evolutionary mechanisms provide the means to obtain these control programs in an effective and adaptive manner.

Control programs are, similar to GOFAI, in essence a mapping from sensory input to motor output. However, in GOFAI this translation consists of quite some steps; from sensory input via symbolic representation and integration with planning to motor output. In contrast, in ER control programs typically provide a direct coupling of sensory input to motor output. Moreover, control programs in ER need to be suitable to be handled by an evolutionary process. Hence, they need to be generic and it should be possible to encode them in an 'evolution-friendly' manner (e.g. a bit string in which bits can be altered easily), so that mutation and recombination operators can be applied. Well tailored for this end is for example a Artificial Neural Network (ANN) or a Classifier System (CS).

Evolutionary mechanisms are used to find those parameters that cause the control program perform in an effective manner. An often used and well studied evolutionary mechanism is a Genetic Algorithm (GA). Within a GA sets of parameters are tested for their effectiveness and those most effective are selected for reproduction. In this way the GA allows for an efficient search in parameter search space. Likewise in Genetic Programming (GP) evolutionary mechanisms are applied not on parameter sets but whole control programs.

2.4.2 Artificial Neural Networks

ANN's are biologically inspired function approximators. They can be trained to learn non-linear functions. A typical and often used ANN is a multilayer feed-forward neural network (also known as a multilayer perceptron) which consists of an input layer, a hidden layer and an output layer. It is called 'feed-forward' because data is processed from input layer to output layer with no 'looped' connections. Each layer consists of one or multiple nodes which have an activation function. Different activation functions can be used, like a logistic function which has a continuous output, a threshold function which fires 1 when the threshold is reached and 0 otherwise or a 'leaky integrator' function in which the activation also depends on the activation of the previous time step. All nodes in the input layer are connected to all nodes

[12] These techniques will be discussed in more detail in Section 2.4.4 and Section 2.4.5.

in the hidden layer which in turn are connected to all nodes in the output layer. Depending on the activation function a node 'fires' its activation value to every other node connected to it. The activation value of a node is simply calculated through addition of all incoming values multiplied by the associated weights and subjected to the activation function. In general a node in a multilayer perceptron takes the form of:

$$y(x) = g\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{2.1}$$

Where $y(x)$ is the output value, g is some activation function, w denotes a vector of weights, x denotes a vector of inputs and b denotes a bias. The use of a bias value ensures the ability to always generate an output, independent of input values which could be zero. In order to obtain more powerful neural networks the activation function should be non-linear (a network with a linear activation function cannot learn the rather simple XOR function). In this fashion the activation of all hidden and output nodes is calculated successively.

Other forms of ANN's are recurrent networks in which a 'looped' connection does exits. Contrary to feed-forward networks they have one or more connections from the hidden or output layers back to the input layer. This allows the recurrent network to recognize structures over time since in every time step internal information is processed again. This is impossible for feed-forward networks since they process input to output in a single time step and no information is kept for the next time step. Simple recurrent networks were first described by Elman (1990). Also continuous time recurrent neural networks (CTRNN) provide a form of memory at the node level, by employing leaky integrator nodes. See Beer (1995) for a detailed description.

To have a network produce a certain output for a given input, the connection weights producing this desired output need to be found. This can be done by training the network. A number of different training methods exist like supervised learning, reinforcement learning and evolutionary training. Supervised learning requires knowledge about the goal states for which the network is trained. The neural network is exposed to training data and learning consists of altering the weights in order to have the output match the training data as optimal as possible. If the training data is representative the neural network should perform equally when confronted with real data. A popular method is backpropagation (Werbos, 1974), in which weights of neurons are adjusted relative to their error rate. This method however is not very well suited for ER, since typically the exact desired output of a neural network is not known. Hence, automated evolutionary search is applied for this end.

Other forms of learning can be applied in a more autonomous manner and are achieved by subjecting the weights of the network to reinforcement learning techniques or evolution. Again the weights of the network are modified to find the best possible solution. In these methods however, the weights of the network are altered based on the global performance of an agent embodying the network, rather then on specific output values[13].

Mostly these networks have a fixed topology in which only weights are evolved. Since a fully connected network can approximate any continuous function (Cybenko, 1989) a fixed topology does not seem to impose restrictions on the capacities of an ANN in principal. However, to optimize the search for effective networks, techniques have been proposed in which not only the weights of a network are evolved but its structure as well. An example

[13]Instead of explicitly stating desired output values, the network is evaluated on how well an agent equipped with this network is able to solve a specific task.

of this is called NeuroEvolution of Augmenting Topologies (NEAT), done by Stanley and Miikkulainen (2002b). It has been shown that NEAT can be more efficient in finding good solutions than networks with fixed topology.

2.4.3 Classifier Systems

A CS (Holland, 1986) consists of a set of rules (the classifiers) which take a general [condition $\rightarrow$ action] form. Conditions and actions are formed using the alphabet {0,1,#}, where 1 = true, 0 = false and # is a wildcard. If the condition is satisfied the action is selected for potential execution. By having multiple classifiers running in parallel, the system allows for a potentially wide range of available actions. A higher order system is used to determine which of the classifiers (provided the entailed condition is satisfied) is executed, both checking for conflicting actions and applicability. An executed action can take the form of an explicit motor action for the robot or can be flagged as a fulfilled condition which then can be processed by the next classifier rule. A CS can be subjected to various learning techniques in order to find the most suitable set of classifiers for a given problem.

Dorigo and Schnepf (1993) proposed to use a CS subjected to evolution to develop effective robot controllers. By encoding the classifier in a generic manner it can be subjected to evolution to find the most effective solution.

2.4.4 Genetic Algorithms

GA's were first proposed by Holland (1975). A GA, or sometimes called evolutionary algorithm, is a biologically inspired technique to search for an optimal solution in a typically large search space which may consist of many dimensions (number of parameters). In Darwinian evolution those members of a population who express the highest fitness are allowed to produce offspring which will presumably share some of its parents' characteristics that allowed them to survive. Likewise a GA can be applied to evolve artificial systems for their ability to survive and reproduce. However, a GA can also be used to evolve systems for the ability to solve any other problem; the ability of a solution to solve a problem then determines its survival rate.

Initially a generation of random solutions is generated and tested. Those solutions which solve the problem best, i.e. score the highest fitness are selected for reproduction. Like nature's example, reproduction methods are recombination and mutation. Recombination of the parent solutions (through various types of cross-over) allows for the combination of partial solutions exhibited by different parents. Mutation can generate novel solutions and hence allows for the possibility to reach all corners of the search space. To effectively apply evolutionary operators on candidate solutions they need to be coded in a generic manner. This can be achieved by encoding the solutions in strings of bits, on which it is easy to apply recombination and mutation.

GA's can be vulnerable for local maxima since the evolutionary forces tend to favor the current optimal solution and thus may prevent the development of a branch which might exhibit less fitness for a while but holds potential to achieve higher fitness on the long run. This can cause a GA to get stuck at sub-optimal solutions. To counter this vulnerability the number of mutations can be increased, thus allowing the solutions to take a leap in search space. The downside of this approach is that extensive mutation is rather destructive for already found schemata which could lead to more optimal solutions. Careful balancing is the key. A method to ensure the preservation of good solutions is 'elitism', in which the best performing solutions of every generation are preserved.

The rate at which a GA converges is strongly problem dependent and may therefore be difficult to predict. However, by keeping track of the 'hamming distance' (the average distance of all candidate solutions within a population) over time, models have been proposed which allow for a fairly accurate prediction (Louis and Rawlins, 1992).

2.4.5 Genetic Programming

Where in GA's candidate solutions are sets of parameters which are successively tested in the same control architecture, in GP the control programs themselves are evolved. To apply evolutionary operators effectively on computer programs, again they need to have a generic form. Suitable for this are tree-like structured programs which consist of function primitives and terminal values. Recombination can be achieved by swapping subtrees of parent programs. Mutation is done through randomly changing function primitives or terminal values in various manners (pruning or growing sub-trees, swapping functions etc.). The functional programming language Lisp is well tailored for GP since it is constituted through tree-like structures. To have GP produce effective control programs, a careful choice of suitable function primitives and terminal values is of crucial importance. The whole set needs to be rich enough to embody a wide range of robot behaviors but there also needs to be a level of abstraction to allow for the creation of powerful programs. The choice of function primitives and terminals again requires creativity on the programmer's part.

2.5 Environments

2.5.1 Different Environments

Within ER the development of both hardware and software can be subjected to evolutionary principles. Different environments can be used for evolution. Of course, in general the best results are achieved when evolution occurs in an environment much like the one in which the evolved robot will need to perform eventually. When control programs for software agents or artificial robots need to be developed it is relatively straightforward to use an environment for evolution which is very similar to the one in which they will be performing. For control programs for real robots on the other hand, best results are achieved when the evolution takes place in the actual robot.

However, applying evolution on real robots can lead to a number of difficulties. For instance, the process of evolution can be very time consuming since multiple individuals need to be tested for a multitude of generations. Also, in terms of cost (energy, maintenance etc.) it can be less attractive to run a lot of trials on robotic hardware. Limitations need to be imposed on the evolutionary algorithm to ensure no behavior is evolved which could actually damage the robot. This however, could potentially cripple the GA's performance. A solution for these types of problems is a hybrid approach in which controllers are first evolved in a simulated environment and later transferred onto the real robot. Differences between simulated and real environments are inevitable, so if necessary the evolution process can be continued for some time to allow the controller to adapt toward these differences. But since "the best model of the world is the world itself" (Brooks, 1991), experiments have also been undertaken in which the whole process of evolution takes place in actual robots.

2.5.2 Simulators

A lot of simulator programs have been developed in the wake of ER. While some being simply an automatic way to expose the controller to realistic sensory input, others embody full 3-D environments including laws of physics. Within a simulation it is relatively easy to also evolve the morphology of robots. This relates to work conducted within Artificial Life (AL) studies. Emphasis differs though from biological plausible simulated systems in Artificial Life to the more practical search for effective robot controllers in ER. More on the relation of AL to ER is discussed in Chapter 3.

The EvoRobot simulator allows for a wide range of experiments in ER. In essence the simulator program consists of a 2-D world in which walls and obstacles can be placed and in which multiple simulated robots can roam. The robot phenotype (its bodily realization within the simulator) consists of a round body, two wheels and various sensors. Its controller is a neural network which can be customized depending on the task demand. Within the simulator populations of robots can be subjected to evolution. A task can be specified through defining a fitness formula and the simulator will select those individuals which perform best. To allow for 'realism' and easy transfer to real physical robot platforms the activation data of the sensors are not simulated (e.g. generated by the program) but based on measurements of actual real world sensors. EvoRobot is described in more detail in Chapter 3.

2.5.3 Hybrid Approaches

In a different study (Nolfi et al., 1994), the EvoRobot program was used to evolve robot controllers in simulation first and than transfer these controllers onto a physical Khepera robot (Mondada et al., 1994, 1999). Recordings of the Khepera sensors and motor behavior were used to have the simulated environment as genuine as possible. After transferring the evolved controllers from simulation to Khepera, performance dropped significantly at first. However, by allowing the controller to evolve in the real environment the same level of performance was reached again in only a few generations. This displays the impossibility to exactly model a real environment, but in general this approach seems feasible since the GA appears to be robust enough to overcome these differences.

2.5.4 Embodied Evolution

One of the first studies in Embodied evolution was done by Floreano and Mondada (1994). In this study the whole evolutionary process took places entirely on a Khepera robot without human intervention. Also, in the study performed by Watson et al. (2002) a population of physical robots was used that autonomously reproduced with one another while situated in their task environment. Within embodied evolution the whole process of evolution takes place in the physical robots. The advantage of this approach is that evolved robots are optimally adapted to the environment; the problem of coping with differences between virtual and real environments is thus avoided. Also, rather than having a centralized setup in which a GA checks which control programs perform best, the evolutionary process is distributed over all physical robots. In essence it is the environment and the interactions in which the robots engage with each other autonomously which determine their fitness. It can be noted that such a setup is more in line with natural evolution since the whole process takes place in the physical realm. However, since in these types of experiments a relatively large emphasis is on technological aspects, it is important to distinguish engineering related challenges from more theoretical challenges. This because the former, interesting as they may be, are not necessary insightful for the study of ER.

2.6 Summary

In this chapter the theory and technique of Evolutionary Robotics was discussed. ER has been developed as an alternative for traditional AI research paradigms in order to achieve more effective control programs for autonomous robots. Emphasis in the paradigm of ER is on embodiedness, situatedness and adaptivity. Techniques can be divided into two categories: control mechanisms and evolutionary mechanisms. Application of both these techniques in simulators and in real robots allow for the development of robots which are well adapted to a wide range of environmental challenges.

Chapter 3

EvoRobot

3.1 Outline

This chapter describes EvoRobot as a software simulator for conducting experiments within the Evolutionary Robotics discipline. EvoRobot is a simulation program which is developed by Nolfi (2000). It is based on the Khepera robotic platform and allows for a wide range of experiments within ER like experiments in simulation, in hardware or as a hybrid approach, as described in Chapter 2. In EvoRobot a two dimensional environment is simulated which can hold a number of objects and in which robots controlled by a simulated neural network can roam. Furthermore, EvoRobot allows for the simulation of an evolutionary process which can find solutions in the form of neural network controllers well adapted for a given task. A simulation can be customized to a high degree with regard to the environment, the type of input sensors of the robot, the task objectives, associated fitness functions and various evolutionary parameters.

3.2 Simulators

A fair number of software programs for the purpose of simulating and evolving autonomous agents has been developed. A lot of these simulations however stem from an Artificial Life (AL) tradition and are only partially applicable to Evolutionary Robotics. This is mostly so because AL emphasizes biological plausibility as opposed to real world application in ER. Furthermore, because AL models are maintained solely in virtual environments and no mapping onto a real world environment is done, they can be more extended and flamboyant. For example, within an AL model it is relatively straightforward to simulate large flocks of flying creatures, but given the current state of technology this would be very challenging to implement in robotic hardware. Because of this, simulators used within ER tend to be relatively sober and simple compared to their AL counterparts. An example of an AL model is the work of Sims (1994), which embodies a full 3D environment in which agents are free to evolve both their neural controller and (simulated) physical body. Interesting as it may be, it would be nearly impossible to implement Sims evolved creatures into functional physical hardware. An example of a more ER based model is the work of Stanley and Miikkulainen (2002a) in which they investigate different aspects of how complexification in genomes can lead to sophisticated solutions. Their robotic model is based on the Khepera robot which makes it suitable for porting and more realistic at the same time. EvoRobot can be placed in the ER category of simulators because it also strives for realistic correspondence to physical hardware implementations.

3.3 Features of EvoRobot

3.3.1 The World

The world of EvoRobot consists of a two-dimensional plane. Typically this plane is surrounded by four walls which effectively define the outer boundaries of the world. The size of the plane can be infinite but in typical experiments boundaries are applied to keep the robots 'focused' on the task at hand and not have them wander around aimlessly in the vast nothingness of a virtual plane. The size is specified in *mm* so a world of 1000 x 1000 corresponds to 1^2 meter. Various objects can be placed in a world. These objects are:

- Walls specifying the outer boundaries of the world. They can be placed inside a world as well, but robots cannot pass through them. Parameters: from x_1, y_1 to x_2, y_2.
- Round obstacles consisting of circular objects with a standard radius of 25 mm. It is impossible for robots to pass through an obstacle. Parameters: coordinates x, y, radius r.
- Small round obstacles are the same as round obstacles, only with half their standard radius: 12.5 mm. Parameters: coordinates x, y, radius r.
- Target areas consisting of a circle which a robot can enter. In essence a target area is a set of coordinates with a specific label. Depending on its sensors the robot may or may not 'know' it is on a target area. Fitness function related actions can be triggered by checking whether a robot is on a target area or not. Target areas can act as a food zone by holding a number of 'food items' which can be foraged by the robots. Parameters: coordinates x, y, radius r, fitness value f, food amount v.
- Lights can be perceived by a robot (provided it has light sensors enabled). A light has a source on a specific coordinate on which the activation of light sensors is maximized. The activation gradually declines as a function of the distance of the robot to the light source. Robots can pass through lights. Parameters: coordinates x, y, radius r.
- Landmarks are objects of various size and shape which can be perceived by a virtual camera. Parameters: coordinates x, y, radius r, object o, color c.

3.3.2 The Robots

Evorobot allows to run experiments based on the Khepera or on the e-Puck robotic platform. Depending on which platform is used, the simulated robot consists of a spherical body of 55 mm (Khepera) or 75 mm (e-Puck) in diameter. It is equipped with two wheels on each side which rotate independently, allowing the robot to explore the environment. Various input sensors provide the robot with information about the perceived surroundings.

These sensors can include:

- Infrared sensors to perceive walls, obstacles and other robots.
- Light sensors to perceive light sources.
- Ground sensors to perceive whether or not the robot is located on a target area.
- ‘Microphone’ or signal sensors to perceive the sound signals emitted by other robots.
- A Camera, which can detect various landmarks and other robots.
- Motor memory, which encodes the activation of its own motor output at $time - 1$.

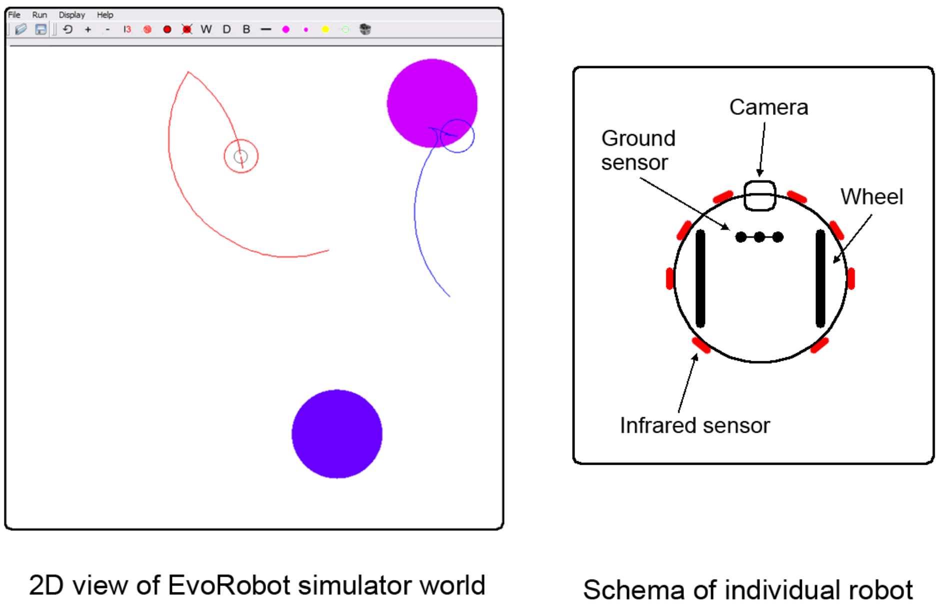


Figure 3.1: *Screenshot of EvoRobot and a schema of an individual robot.*

Robots have two motor outputs, specifying the speed of the left and right wheel and a signaling output emitting a sound signal which can be perceived by other robots. Robots cannot move through walls, obstacles or other robots. When robots encounter various objects in the world, their appropriate sensors are activated. The activation may be binary or continuous, depending on the type of object. Obstacles and other robots can be perceived up to a distance of about 4 cm. with infrared sensors. Other robots can also be perceived at longer distances through the camera and are identified through the color of their LEDs. The activation function of the sensors is based on actual recordings of existing sensors from both Khepera and e-Puck robots in order to try to capture the characteristics of real sensor activation as genuine as possible. To allow for some robustness, a certain percentage of noise can be applied on the sensor data. This drives evolutionary search toward solutions less affected by small variations in sensor data which will inevitably be present in hardware application. Likewise motor activation is also based on recorded values from real motors. Within the simulator a robot can be selected to visualize the current activation of its sensors in a graphical manner and sensor data may be saved for analytical purpose.

Figure 3.1 shows a screenshot of EvoRobot with two robots searching for two target areas, and a schematic picture of an individual robot and its sensors.

3.3.3 The Neural Network Controller

The neural network controller of a robot consists of a multi-layered feed-forward neural network, possibly augmented with recurrent connections. Its structure is typically three-layered: one input layer, one hidden layer and one output layer. The first layer consists of sensory neurons that encode the state of the robot's sensor. The second layer consists of internal neurons. The third layer consists of motor neurons that encode the state of the motors controlling the two wheels and the type of 'sound signal' produced by the robot. Sensory input to motor output is processed as follows. First, the output of the input layer is calculated according to Equation 3.1:

$$O_j = O_j^{(t-1)}\tau_j + I_j(1-\tau_j) \tag{3.1}$$

With O_j being the output of the j^{th} input neuron, $O_j^{(t-1)}$ the output of the j^{th} neuron at the previous time step, τ_j the time constant of the j^{th} neuron and I_j the activity of the j^{th} sensors.

Then, the activation of units in the hidden layer is calculated according to Equation 3.2 and the output is calculated according to the logistic function (3.3):

$$A_j = t_j + \sum_i w_{ij} O_i \tag{3.2}$$

$$O_j = O_j^{(t-1)}\tau_j + (1+e^{-A_j})^{-1}(1-\tau_j) \tag{3.3}$$

With A_j being the activation of the j^{th} neuron, t_j being the bias of the j^{th} neuron, w_{ij} the weight of the incoming connections from the i^{th} to the j^{th} neuron, O_i the output of the i^{th} neuron, O_j being the output of the j^{th} neuron, $O_j^{(t-1)}$ the output of the j^{th} neuron at the previous time step and τ_j the time constant of the j^{th} neuron. The sensory neurons and internal neurons are leaky integrators that hold a certain amount of their activation from the previous time step. They tend to vary their output at different speed, depending on the time constant τ.

Last, the output of the motor units is calculated according to Equation 3.2 and the logistic function (3.4):

$$O_j = \frac{1}{1+e^{-A_j}} \tag{3.4}$$

With O_j being the output of the j^{th} neuron and A_j being the activation of the j^{th} neuron.

These functions are based on previous experiments on neural models conducted by Nolfi (2002). Outputs are real values ranging from 0.0 to 1.0 where 1.0 codes for 100% motor activation. The neural network controller may have recurrent connections, typically realized by using motor outputs and internal unit outputs as input values at $t-1$. This allows the network to maintain a form of 'memory' which could be beneficial for certain tasks. The connection weights and biases are in the range of [-5, 5] and the time constants in the range of [0.0, 1.0]. Time constants encode how much the activation of a unit is influenced by its previous activation state, with high values causing the activation of a unit to change relatively slow over time.

3.3.4 The Simulation Process

The lifetime of individual robots (during which their fitness value is evaluated) consists of a certain number of trials lasting a certain number of time steps. Each time step corresponds to 100 ms in real time. For each time step all dynamic objects in a world are updated. The update process consists of calculating every change in the world along with the effects. Most objects in a world are static and need no update, but target areas for instance may contain a specific amount of food which can be consumed by robots. The robots of course are the most dynamical objects. The update procedure for the robot consists of calculating all input activations through the neural network. The output values of the robot controller are then applied in the world. For instance, a certain activation pattern of the infrared sensors may cause the neural controller of a robot to output a value of 0.3 for the right motor unit and 1.0 for the left motor unit. The speed of the motors is then set accordingly (based on a recorded sample from a real robot) causing the robot to turn. When all objects in the world are updated and all robots have executed their movements 1 time step has passed. In principle the simulation trial will continue until the maximum number of time steps is reached but conditions may be specified which terminate the trial prior to the total number of time steps. Such a condition might be a robot crashing into a wall or obstacle.

3.3.5 The Evolutionary Process

EvoRobot allows for evolving neural controllers of robots to obtain effective control programs for specific tasks. This is done by applying a genetic algorithm on the free parameters (the weights, biases and time constants) of the robots' neural controllers. The initial population consists of 100 genotypes which are randomly initialized and correspond to 100 different type of robots. A generation may hold an arbitrary number of individuals but typically 100 individuals are used[14]. The performance of the robots is evaluated during their lifetime on the basis of a fitness function that computes for every robot how well it is able to solve a given task. For example, the performance with respect to an ability to produce an obstacle avoidance behavior can be measured by calculating the average activation state of the robots' infrared sensors on every time step. The performance with respect to the ability to reach a particular area of the environment can be measured by calculating the percentage of trials in which the robot is located in the right area at the end of the trial. The score of each individual (usually an average of multiple trials) is compared to all other individuals within

[14] A population of 100 individuals is large enough to ensure variation within the population but small enough to run the evolutionary process with computationally realistic requirements.

the population and based on this a number of the best performing individuals (typically 20) is used to generate offspring (Truncation selection).

A certain percentage of the weights, biases and time constants of the neural network which make up the individuals is randomly changed. This is done through subsequentially mapping all the weights, biases and time constants onto strings of 8 bits. Then every bit is subjected to a 50% chance to flip with a specified random chance. So if the mutation percentage is set to 2%, the chance of a bitflip is 2% for every bit. Mutation is the only recombination operator used because other common methods, like cross-over, are rather disruptive for behavior delicately distributed over a network. The way the connection weights, biases, and time constants are encoded within the genotype does not allow for effective recombination through cross-over.

Again, the number of generations and thus the runtime of the genetic algorithm may be arbitrarily long but is usually limited because of practical considerations and the fact that the evolutionary process mostly converges to a specific solution which may be suboptimal but nevertheless constitutes the end of the evolutionary search. During the evolutionary run the progress of the population can be monitored through a graph displaying the best and average fitness scores for every generation. The best performing individual of every generation can be saved for later use. A wide range of evolutionary parameters can be set. These include:

- Number of individuals in a population.
- Number of trials in which one individual is tested.
- Number of environments in which the individuals are tested.
- Number of individuals allowed to reproduce.
- Mutation rate, i.e. percentage of bits encoding weights, biases and time constants which is subjected to a bitflip (mutation).
- Specifying whether or not elitism[15] is used.
- Number of generations for which the evolutionary process runs.
- Specifying whether or not a trial ends when an individual collides with an obstacle.
- The number of replications of the whole evolutionary process using a different random seed for every replication.

The evolved neural controllers embody the best performing solutions for the specified task. Depending on research needs tasks can be customized to a high degree with respect to the number of robots, their capacities and their objectives. Examples of tasks are 'obstacle avoidance', 'search for target areas', 'predator-prey scenarios' and 'spatial distribution coordinated through signaling'.

[15]When elitism is used the best performing individuals are preserved for the next generation. This can increase the performance of the evolutionary search, because without elitism well performing individuals can easily be lost due to mutation.

3.4 Modifications Made in EvoRobot

To conduct the experiments described in this thesis, some modifications were made in EvoRobot. First, new fitness functions were developed in order to evaluate the performance of robots in the experimental setups described in Section 5.2 and Section 5.3. Second, a virtual camera routine was added, which simulates the detection of other robots through camera, based on the functioning of the camera on the e-Puck robot. Also, some new type of input neurons were defined, like the 'memory sensors' used in the experiments of Section 5.3, which allow a robot to remember which target area it has visited last. Furthermore, extra parameter settings were added which also facilitate the described experiments. To transfer evolved behavior onto real robots, the interface between EvoRobot and e-Puck robots was further developed and tested.

During the period in which the experimental work of the thesis was conducted, Nolfi and collaborators were developing a new version of EvoRobot, called EvoRobot*, which includes new features and which allows to run experiments based on the e-Puck platform. EvoRobot* will be released under an open source license when it is sufficiently tested and documented.

3.5 Using EvoRobot with Real Robots

EvoRobot allows for a tight integration with robotic hardware. The robots in the simulator are modeled after the Khepera robot originally, but closely resemble e-Puck robots too. When the option 'connect to real robots' is selected, the simulator will issue motor commands to the (wireless) connected robots instead of simulating the virtual environment. The connected robot executes the motor commands and sends its sensor readings back to EvoRobot. These sensor readings are fed into the neural controller in order to obtain the motor outputs for the next time step.

The neural controller of the robots is run in EvoRobot on the main computer, which is convenient for data analysis. However, when the amount of data which needs to be transferred over the wireless connection is high, the neural controller should run on the robots' on-board computer. In the latter case, the role of the main computer is to hold genetic information of the population, to generate new generations, to send genetic information to the robots and to collect the fitness data at the end of robots' lifetime.

The connection is established over a standard COM port, so in theory the number of connected robots is arbitrary. Since a standard time step in EvoRobot lasts for 100 ms. the sending and receiving process to connected robots ideally is < 100 ms. This is sometimes problematic for large information streams like video information from on-board cameras. To cope with such difficulties it is best to pre-process such large information streams on the robot side and only send relevant abstracted data to EvoRobot.

In the case of the experiments described in this thesis the camera was used to detect the relative position of other robots (when located in the field of view). This method requires that e-Puck robots have their red LEDs enabled. The detection algorithm searches for the largest concentration of red pixels in the captured video data and calculates the relative angle. This angle is communicated to EvoRobot where it may be fed into a sensory input. More details about using EvoRobot in combination with e-Pucks are discussed in Chapter 6.

3.6 Summary

EvoRobot is a simulation program which allows for a wide range of experiments within the Evolutionary Robotics paradigm. Various evolutionary parameters can be tuned and tasks can be customized. EvoRobot is well suited for the simulation of evolving communication systems in a team of robots. Furthermore, by simulating robots based on real robot characteristics, evolved controllers can be transferred onto real robots like the Khepera or e-Puck.

Chapter 4

Evolution of Communication

4.1 Outline

This chapter first discusses the motivations for studying Evolution of Communication (EoC) through ER, along with some key issues about the EoC in both natural and artificial systems. Next, in order to conduct effective experiments on EoC in ER, suitable tasks need to be identified for which robots need to use communication. The identification of these tasks is not trivial, but theoretical and intuitive guidelines can be exploited for this purpose. To facilitate the process of finding suitable tasks a task space model is proposed. Also, more complex and sophisticated forms of communication may be observed in ER models. Experiments are proposed which investigate what type of communication systems can emerge, how these change during the process of evolution and if they progressively become more complex. Some aspects of both task and signal complexity are discussed, along with the postulation of a complex task.

4.2 Motivations for Studying EoC through ER

Motivations for the study of EoC in artificial systems are twofold. On the one hand it may shed light on the development of communication in natural systems as encountered in animals and humans. On the other hand it may serve practical goals like enabling robots to maintain themselves in a real environment and perform various collective tasks in an adequate manner.

Regarding the first motivation, the origin of natural communication is an interesting topic but can be unpractical to study because of the obvious reason that at present natural communication is at the end of the evolutionary chain. This implies that only the end product of evolution can be studied, but no intermediate evolutionary stages. To encounter the evolution of a new communication system 'in the wild' is highly unlikely and most presumably impossible to study because of the time frame and all sorts of modern day disruptive influences which would render the evolutionary process 'unnatural'. Thus, gaining insight in the evolution of communication in natural systems is normally done through comparative studies of various animal species like the work of Hauser (1997). Data on animal communication is always the product of millions of years of evolution, but it is possible to study the evolutionary history of the species and to examine what means for communication the species' ancestors had. However, since the nature of communication systems cannot be derived from fossil records directly but only indirectly by studying physical characteristics which might facilitate communication, empirical data for such studies might be poor. Because actual observation of pre-developed communication systems is not possible, it can be informative to simulate the development of communication in an artificial system. By employing such an approach, it is possible to study the development of such a system as a whole, from premature development to full-blown functionality.

Regarding the second motivation, as described in Chapter 2, the new paradigm of Embodied Cognition requires the next generation of robots to be adaptive and reactive toward an ever changing dynamical environment. As argued, such adaptation can be achieved through application of evolutionary forces on robot controllers. To effectively execute collective tasks which require cooperation between multiple robots the same requirements are imposed on the communication systems such a team of robots would employ. Because the communication system is part of the set of behaviors robots need to exhibit, it should be developed in combination with other required behaviors. By imposing minimal restrictions on the shape and meaning of communication signals and channels, evolution is free to find the most effective solutions shaped toward a given task. For this means it is important to gain insight in the factors and prerequisites which may lead to the evolution of effective communication systems. The practical applicability of effective robot communication systems is potentially large given the not unrealistic assumption of an increase in robotic participation in society.

4.3 About Natural Communication Systems

For the purpose of clarity, some key concepts are discussed here. First, what is communication? Communication is usually assumed to occur between two or more individuals when *information* needs to be conveyed. What exactly constitutes information exchange is still an unsettled dispute, but a general consensus regards the framework of Weaver and Shannon (1949) to be suitable enough. In this framework it is stated that the function of information is to reduce uncertainty about a particular event. This is formalized as follows:

$$H(X) = -\sum_{i=1}^{k} p_i \log_2 p_i \tag{4.1}$$

Here H stands for the uncertainty about whether a series of events $x_1, x_2, .., x_k$ denoted as X will occur and p_i stands for the probability that the i^{th} event of set X will occur. Communication then is the act of exchanging information between a sender and a receiver. Whether or not communication has occurred can be determined by using the above formalism: communication between two individuals is successful when uncertainty H of the receiver about a particular event X is reduced.

As Steels (2005) comments, a second viewpoint about what constitutes successful communication emphasizes the inferential nature of human communication. From this point of view communication is successful when the receiver has correctly identified that part of the world which the sender is communicating about and executes behavior accordingly. Thus communication is not simply a synchronization process of the knowledge base of sender and receiver since the information conveyed does not have to hold the same meaning for sender and receiver or may be merely a trigger to extract the full information from the environment. This implies that determining whether or not communication was successful cannot be done by comparing the sender's and receiver's knowledge bases but instead should be inferred from observable behaviors of the receiver.

In both viewpoints the core component is the exchange of information between individuals, whether this is a literal transmission or used to infer other information. This exchange then, seems to happen when benefit can be gained by sharing knowledge between the members of a group of two or more individuals. Such a group can be a nest of insects, a pack of wolfs, a group of humans, a collection of simulated agents or a team of mobile autonomous robots. Independent of the group constitution, when confronted with a collective task individuals need to relay on the exchange of information. A collective task may be anything; it could be conceived as the more abstract task a natural species has to maintain itself in a hazardous environment, a group of students who need to fulfill a study project or the rather practical task of a team of autonomous robots that needs to collect garbage bins. In all such cases, in order to achieve optimal performance, individual members of the group may need to inform other members about various facts regarding their own present state, relay information they can perceive but others cannot or issue guidance.

Thus, various communication systems are employed for this end. To be effective, these communication systems need to be shaped toward the objectives of the users. This has occurred in natural systems through the Darwinian principle of natural selection. When examining various natural communication systems it is clear these systems are highly adapted toward the tasks and challenges individuals are faced with. Examples of this can be found in the work of Hauser (1997). He applied the model of Tinbergen (1952) on the evolution of communication in the animal kingdom in which *mechanical*, *ontogenetic*, *functional* and *phylogenetic* perspectives "represent a coherent theoretical and methodological framework for both studying and explaining communication (in natural systems)". This resulted in a complete study of various kinds of communication systems as they can be encountered in wildlife and sheds light on the evolutionary origin of such systems and how they were developed toward specific ecological needs.

While natural selection shapes natural communication systems, artificial communication systems can be developed through simulating the process of evolution. ER provides a suitable platform for doing so. Moreover, it can be used to study both the prerequisites and the development of effective communication systems.

4.4 About Evolving Artificial Communication

Within the enterprise of evolving communication in artificial systems, the status of artificial communication needs to be considered. While communication on the one hand functions as a means to exchange information and can be seen as a cooperative action between agents, on the other hand it is also an individual action for which agents are selected through evolution when it is beneficial. Hence, from the perspective of a single agent, the way to learn to be successful in communication is similar to learning any other output. It may be argued that communication is a property which is applied from an outsider's point of view onto observed behavior. From the agent's perspective their is no difference between learning to adjust its motor outputs when its proximity sensors are activated and learning to 'utter' a specific signal over designated communication channels in order to increase its fitness by manipulating the behavior of other agents. Hence, it could be argued there is nothing special about learning such behavior compared to other behavior. The fact that from a human perspective it can be said that agents communicate with each other may be as true as it is to say that agents 'do not like to bump into walls' or 'like to stay on target areas'. In essence it is a result of a specific set of evolutionary constraints which drives a population of agents to develop such behavior.

However, the view adopted in this thesis is that communication can be distinguished as a special case of behavior when it is regarded in a functional manner. It is of course an action of an agent as any other, but by taking into account the dynamics of the interaction between agents which allows them to achieve objectives which could otherwise not be achieved, it appears to be justified that such acts of communication deserve special attention. The label 'communication' may be arbitrary in the sense that it could be disputed whether or not agents use *real* communication comparable to natural species, but on the other hand it must be admitted that observed communication behavior in fact does have characteristics of natural communication systems and also functions in a comparable manner. Therefore artificial communication as a specific behavior observed within artificial systems is judged to be sufficiently equal to natural communication systems, and hence may be functionally considered as such.

4.5 The Characteristics of a Suitable Task

Since the outcome of an artificial evolutionary process is shaped to the task and the associated fitness function individuals are confronted with, it is interesting to examine what type of task drives a population of individuals to use communication. Because communication systems should be functional it is not interesting to conjure tasks which specifically reward communication act through explicit fitness gain. Rather, tasks should be formulated in general terms and fitness functions should evaluate the evolved behavior of individuals as a whole. Communication then can evolve as a facility for effective solutions. In this way communication is not stimulated just 'for the sake of' communicating, but for its functionality. So, what is considered interesting are behavioral solutions which happen to incorporate a communication system which add to the effectiveness of the solution.

From an outsider's point of view it is not straightforward to determine whether or not the solution for a specific task can be enhanced by using communication. There exists no formal description about which conditions require the use of communication and which not. Adaptive as the evolutionary process is, it is known to find 'surprising' novel solutions which may or may not rely on communication. Thus, to find a suitable task for usage in artificial evolution, a combination of inspiration from natural studies, intuition and heuristic approach is used to acquire an experimental setup which may require the use of communication to be solved effectively. Some examples of these sources of inspiration, heuristics and intuition are discussed here.

An important aspect that influences the evolution of communication is whether or not communication provides an adaptive advantage for both the sender and the receiver of a signal. This has been investigated by several researchers who tried to explain how altruistic traits (e.g. the ability to produce a signal that is beneficial for the hearer but not for the speaker) can evolve. In this respect, Dawkins and Krebs (1978) argued that animal communication should largely be seen as a means to manipulate other animals, rather than a mechanism to strive for a collective goal. The signaler is primary selected to benefit from manipulation of the receiver; whether or not the receiver happens to benefit as well is coincidence. Selection pressure is postulated to work on the gene level of individual animals. However, Parisi and Mirolli (2007) argued against this idea, stating that the adaptive advantage for the use of communication stems from both an advantage for the signaler and the receiver and that the viewpoint of manipulation is too shallow to explain all natural communication. In order for a receiver to exhibit behavior which is beneficial for the sender, this behavioral response should be directly beneficial for the receiver as well. Altruistic forms of communication can result as an exploitation of the tendency of the speaker to produce signals that are functionally different in different situations, and also because agents tend to use their own signal as a form of external memory. Experiments on mobile robots performed by Floreano et al. (2007) showed that kin relatedness and group selection tend to stimulate the development of effective communication, whereas unrelatedness combined with individual selection schemes actually resulted in deceptive strategies.

In addition to the issue of whether or not communication should provide an adaptive advantage for both speaker and hearer, several other factors seem determine whether or not communication can evolve. In a survey about the progress in the simulation of emergent communication and language performed by Wagner et al. (2003) it was argued that “in situated simulations where agents interact with an environment in a causal fashion, many other factors have been shown to affect communication, including agent density, food distribution, predator density, signal honesty, and sexual selection”. Thus, there seems to be no exact requirement or condition which is crucial for the development of communication.

The ‘Blue Paper’ of the ECAgents project group (Nolfi et al., 2006) distinguishes three different classes of tasks which may favor the use of communication. First, “cases in which agents have access to different sensory-motor information and in which they might increase their adaptive capabilities by having access to and exploiting other agents information” (ECA class 1). In these cases one agent has access to information which can be useful for another agent who is not in the position to acquire this information. For instance, a robot may be on a specific spatial location where its sensors can capture data which could be used by a second robot whose sensors are not in range. Second, “cases in which agents might increase their adaptive ability by manipulating the behavior of other agents” (ECA class 2). In such a case an agent manipulates the other agent’s actions in favor of its own objectives. Note that this could also include the first case: one agent could manipulate another agent to have it send useful information. Third, “cases in which agents should produce tightly coordinated behaviors that require a continuous bi-directional tuning of agents’ motor and/or signaling behavior” (ECA class 3). This includes cases in which agents may come to a ‘mutual conclusion’ and related actions through extensive bi-directional communication. In some studies this form of communication was observed over infrared channels only (Quinn, 2001), while in others it consist of a combination of infrared and ‘sound’ signals. Such an interaction between two agents can show similarities to duetting as observed in animal communication (Ujhelyi, 1996) and may be “one of the possible prerequisites for the evolution of language” (Marocco and Nolfi, 2007).

4.6 A Model of Communication Prerequisites

Because the three classes of tasks identified in the Blue Paper of the ECAgents project appear to be the most comprehensive they will be used as guideline from here on in order to identify some features of a task which triggers the development of communication. These classes are quite general however and many implementations can be conceived. To obtain a practical implementation a 3-axis task space is proposed which roughly embodies the three classes in a simple manner (see Figure 4.1). This model is by no means sufficient to capture the three different classes of tasks completely, but nevertheless may shed light on some of the basic task characteristics which can trigger the development of a communication system. The model consists of the following dimensions:

- D1: Directional information available for other robots. Robots can have access to information which may be beneficial for other robots. This dimension can be associated with ECA class 1.
- D2: Cooperation required. All robots need to execute a task collectively: fitness is only gained when both robots perform well. This dimension roughly corresponds to ECA class 2.
- D3: Local interaction impossible. Robots need to be in separate locations as part of the task requirements. When the possibility to interact locally through infrared sensors is absent implicit signaling cannot be used, but explicit sound signals are still available to achieve extensive bi-directional communication as described in ECA class 3.

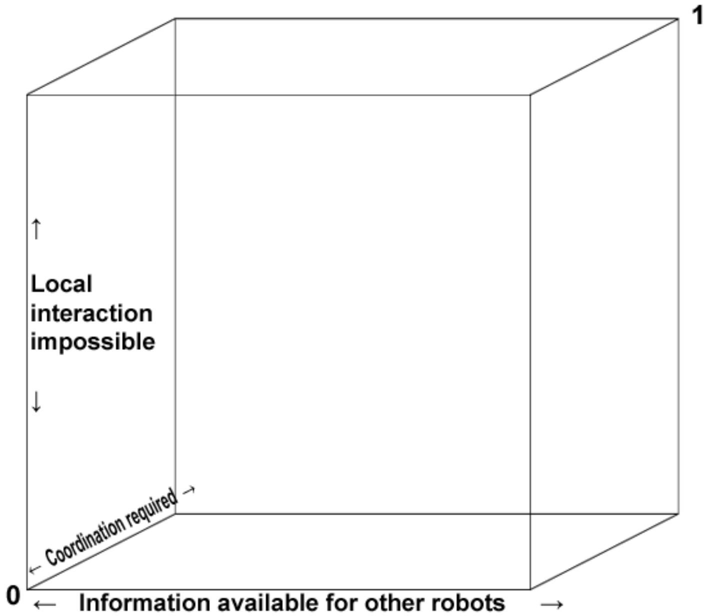


Figure 4.1: *Task space model*

By testing systematically the influence of each dimension on the development of communication, conclusions may be drawn about the necessity of a dimension and the way it interacts with other dimensions. The results of the experiments along with an analysis are described in Chapter 5.

The dimensions are implemented in a binary manner here for clarity and experimental straightforwardness. However, they could be conceived gradually as well by proposing continuous axes and distinguishing cases which increasingly represent or embody the associated class of tasks. For instance, in the first class of tasks in which one robot has access to information which is useful for another robot, successive tasks may be conceived in which the amount of non-global information observable for individual robots increases progressively. By conceiving the task space on a continuous scale (rather than binary) specific regions may be identified that embody those task characteristics crucial for triggering communication.

4.7 Forms and Modalities of Communication

Even though the exact nature of tasks and prerequisites that can cause the emergence of communication are not precisely known as yet, it has already been shown in multiple studies that various forms of communication in fact do emerge in multi robot settings through evolution. In general the communication system is not specified in advance, but emerges as a property of the solution found through evolution. For instance, Di Paolo (2000) showed robots can locate each other using acoustic signals in a synchronized fashion that resembles turntaking. In studies performed by Quinn (2001) it was shown that communication protocols can emerge without the use of dedicated communication channels but through an elaborated activation of other robots' infrared sensors. Marocco and Nolfi (2006) showed how a team of 4 robots can develop a communication system which allows them to solve a spatial task. Individual behaviors and social-signaling behaviors are tightly co-adapted.

However, the pioneering experiments reported above are rather simple in terms of the task to be solved and in term of the sensory-motor apparatus of the agents. In the work described in this thesis a slightly more complex experimental setting was developed. The aim of this experimental setup is to verify if a more complex communication systems can emerge. What constitutes as 'more complex' needs to be clarified. Different levels of complexity can be distinguished, both for the complexity of the evolved communication system and the complexity of the task used to achieve this.

Signal complexity:

- Number of different functional signals.
- Mono directional or bi-directional signals; a signal is used to directly manipulate other robots' behavior or a signal is used to synchronize with another robot's signal in order to achieve a collective action.
- Level of abstraction from the produced signals; a signal may be a direct mapping from single sensor input to signal output or a more abstract signal which is the product of combining multiple sensory inputs (or a result of combined sensory information distributed over time) and thus does not directly encode a single value of a sensor input.
- Implicit or explicit signals; implicit signals are motor actions of an individual (e.g. remaining on a food area) which do not exclusively function as a communication signal, but nevertheless might have a communicative function for other individuals which are able to detect this actions. Explicit signals instead are motor actions (e.g. the production of sounds) that exclusively have a communication function.

Task complexity:

- Number of different individual behaviors required.
- Number of different sensory inputs which can supply information.
- Level of cooperation among robots that is required to solve the given task.

It is naturally to assume that an increase in task complexity would lead to an increase in signal complexity as a more complicated task requires more complicated behavior. However, the relation between task complexity and signal complexity is not that straightforward. When confronted with a complex task, evolution may nevertheless find a surprisingly simple yet effective solution which does not necessary rely on more complex forms of communication. The results and an analysis of the more complex experimental setup is described in Section 5.3.

4.8 Summary

EoC in natural species is hard to study since only the communication systems at the end of evolution are present today. Using ER techniques the process of EoC can be studied in artificial settings, which may help to understand the evolution of natural communication systems as well as provide functional solutions for robots faced with a collective task. For this end it is important to understand which characteristics of an experimental setup may drive a team of robots to develop an adequate communication system. In order to investigate these characteristics, a task space model and associated experimental setup was proposed. Furthermore, next to simple forms of communication, an experimental setup was proposed which may lead to a more complex communication system.

Chapter 5

Experiments

5.1 Outline

This chapter describes two series of experiments performed in EvoRobot. The first series consists of 8 different setups derived from the postulated task space model in order to identify underlying principals which may trigger the development of communication behavior. The influence of three different task dimensions are tested systematically. Results show that it is hard to isolate these three different dimensions, since their effect on the evolution of communication is influenced by the other dimensions. Nevertheless some conclusions can be drawn and the implementation of the dimensions is discussed. The second series of experiments investigate how a more complex task can lead to a more complex communication system. An analysis is given of communicative and behavioral skills that evolved in the two most interesting replicas.

5.2 First Series of Experiments

5.2.1 Experimental Setup

Two simulated robots are placed in a 110x110 cm. world. In this world there are two target areas placed at random positions. The task of the robots is to find and occupy the target areas. The robot controller consists of a feed forward neural network with recurrent connections. Depending on the task, robots may be equipped with a camera which allows them to perceive the relative direction of the other robot if it is in the camera's field of view (36°). This camera provides three inputs (C_1,C_2 and C_3). The first input encodes the relative angle of the other robot in the right visual field (with 1.0 activation when the other robot is at 1° right declining to 0.0 activation when the other robot moves out of the right field of view ($> 18°$)). Likewise the second input encodes the relative angle of the other robot in the left field of view in a mirrored fashion. The third input has a 1.0 activation when the other robot is not perceived in either the right or left field of view. See Figure 5.1 for a graphical representation.

Next to this robots have the ability to send and receive sound signals anywhere in the world. Signals consist of real numbers with values ranging from 0.0 to 1.0. The signal of the other robot is encoded as one input unit and another input unit encodes the robot's own signal. Furthermore robots have 8 infrared sensors and 2 ground sensors encoding binary if a robot is on target 1 or target 2. All inputs are fed into the neural network which consists of an

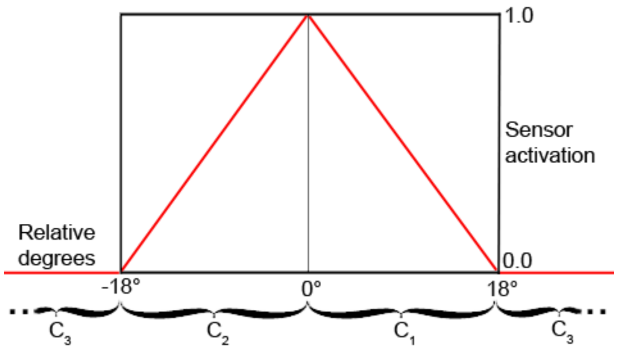


Figure 5.1: *Schema of the activation of the three camera input units C_1,C_2 and C_3 which is based on the relative angle of the observed other robot*

input layer, a hidden layer and an output layer. The hidden layer consist of 2 hidden neurons with connections to themselves, from the input layer and to the output layer. The input layer is also directly connected to the output layer. The output layer specifies the motor output of the left and right wheel of the robot and of the emitted communication signal (Figure 5.2). The sensory and internal neurons consists of leaky integrator neurons. The motor neurons are activated on the basis of the standard logistic function as described in Chapter 3.

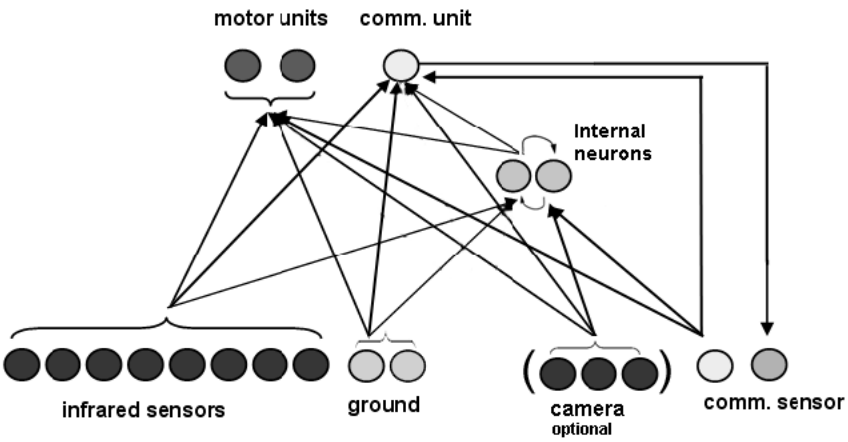


Figure 5.2: *Schema of the neural controller used in the first series of experiments*

The initial population consists of 100 individuals which are successively tested for their ability to solve the given task. One individual consists of a team of two robots which is homogeneous, i.e. the genome of each individual encodes the free parameters of a single neural controller that is duplicated and embodied in two robots. This implies that robots are selected for performing cooperative behaviors, i.e. both robots have the same objective since they embody one individual. Each individual is tested in 10 trials where every trial lasts for 1000 time steps. 20% of the best performing indiviuals are allowed to reproduce with 2% of the bits encoding their neural network weights, biases and time constants mutated. A weight, bias or time constant is encoded in eight bits as a real number ranging from [-5.0 to 5.0] or from [0.0, 1.0]. Every bit is subjected to a 2% chance of mutation with a 50% chance of bitflip when it is mutated. The population is evolved for 1000 generations. 15 replications of this experiment are made.

The three dimensions of the task space model discussed in Section 4.6 are implemented as follows:

- D1: Directional information available for other robots. Robots can have access to information regarding the position of target zones which can be picked up by other robots through camera. Opposed to this is a task in which robots are not equipped with a camera and therefore no directional information can be shared.

- D2: Cooperation required. All robots need to execute the task collectively: fitness is only gained when both robots are located in the proper place. Opposed to this is a task in which a single robot can also gain fitness by being in the proper place. In the latter robots have to develop individual effective behavior and depend less on cooperation (although collective fitness is still increased when both robots perform well), while in the former their fitness also depends on the actions of other robots. Hence it is in the interest of a robot to have other robots behave appropriately which could be achieved by manipulating them.

- D3: Local interaction impossible. Robots need to be in separate locations as part of the task requirements. Opposed to this is a task in which robots have to be in the same location and have the possibility to interact locally through infrared sensors. When the possibility to interact through infrared sensors is absent implicit signaling cannot be used but explicit sound signals are still available to achieve extensive bi-directional communication. However, it is expected that the impossibility to use infrared sensors for this kind of task does influence the development of this form of communication.

In order to test the influences of the three task dimensions robots are subjected to eight different tasks setups which systematically embody these three dimensions. Thus, the list of tasks is as follows:

Task	Description	
1	Robots have no camera Fitness is also scored by single robots occupying a target Robots have to occupy a target together	D1=0 D2=0 D3=0
2	Robots have a camera Fitness is also scored by single robots occupying a target Robots have to occupy a target together	D1=1 D2=0 D3=0
3	Robots have no camera Fitness is only scored when two robots occupy a target Robots have to occupy a target together	D1=0 D2=1 D3=0
4	Robots have no camera Fitness is also scored by single robots occupying a target Robots have to each occupy a different target	D1=0 D2=0 D3=1
5	Robots have a camera Fitness is only scored when two robots occupy a target Robots have to occupy a target together	D1=1 D2=1 D3=0
6	Robots have a camera Fitness is also scored by single robots occupying a target Robots have to each occupy a different target	D1=1 D2=0 D3=1
7	Robots have no camera Fitness is only scored when two robots occupy a target Robots have to each occupy a different target	D1=0 D2=1 D3=1
8	Robots have a camera Fitness is only scored when two robots occupy a target Robots have to each occupy a different target	D1=1 D2=1 D3=1

The performance of an individual controller (comprised of two identical neural networks controlling the two robots) is measured over different trials and averaged in order to rule out ‘lucky’ scores of individuals who happen to be placed in positions from which fitness is easily achieved. In task 3, 5, 7 and 8 where fitness can only be acquired when both robots satisfy the occupation condition (i.e. the occupation condition is satisfied for a robot when it is on the proper target area, depending on the experimental conditions), the associated fitness function is as follows:

$$f(x) = \frac{\sum_{i=1}^{T} \sum_{j=1}^{t} O_{ij}}{T} \tag{5.1}$$

Where $f(x)$ is the fitness score of the individual, T is the total number of trials for which an individual is tested, t is the total number of time steps of one trial and O_{ij} is 1 when the two robots satisfy the target occupation condition on time step j in trial i and 0 when this condition is not satisfied.

In task 1, 2, 4 and 6, where fitness can also be acquired by individual robots satisfying the occupation condition, the associated fitness function is as follows:

$$f(x) = \frac{\frac{1}{2}\sum_{i=1}^{T}\sum_{j=1}^{t} R1_{ij} + R2_{ij}}{T} \tag{5.2}$$

Where $f(x)$ is the fitness score of the individual, T is the total number of trials for which an individual is tested, t is the total number of time steps of one trial, $R1_{ij}$ is 1 when robot 1 satisfies the target occupation condition on time step j in trial i and 0 when this condition is not satisfied and $R2_{ij}$ is 1 when robot 2 satisfies the target occupation condition on time step j in trial i and 0 when this condition is not satisfied.

All experiments were typically run on a Pentium 4 computer (3GHz) with Windows 2000 Service Pack 4. Average runtime of one replica was approximately 2 hours.

5.2.2 Results

Fitness

First the different fitness scores on the 8 tasks are discussed here. In principal the same amount of fitness can be scored in every task, but tasks in which the proper target area for a robot is depending on which target area is occupied by the other robot are harder, i.e. so in these task setups less fitness is scored. See Figure 5.3 for the mean fitness scores on every task. Task 3 turns out to be the hardest task, while in task 2 the most fitness is scored. This is in line with expectations, since in task 3 robots have no means to locate the proper target area (because they are deprived of camera vision) but are yet required to be together on one target area. In task 2 on the other hand robots can use directional information provided by camera vision to locate the proper target area; furthermore, fitness is also gained when one robot has found a target area so less cooperation is required. When the average fitness scores of the different task dimensions are compared, it turns out that the biggest impact on fitness score is on the ‘Cooperation is required (D2)’ dimension because all setups in which cooperation is not required (task 1, 2, 4 and 6) score higher fitness than those setups in which cooperation is required (task 3, 5, 7 and 8). This is not surprising, since it is easier for one robot to locate a target area alone than it is for two robots to find a target area together.

Signal Use

To investigate the influence of the different dimensions on communication usage, performance of robots in all tasks are compared to a control group in which the possibility to use signals was deprived. Robots had to solve the same task, only without the use of explicit sound signals.

When all task dimensions are set to 0 in task 1 it turns out that the average fitness score of the robots is marginally better when signaling is deprived (Figure 5.4). This can be explained by the fact that there is no clear functionality for signal use when solving this task while on the other hand the presence of signals could influence or ‘distract’ the behavior of the robots. So in order to perform optimally robots have to learn not to send signals or ignore them. This could explain why it seems slightly harder to learn the task with signaling enabled compared to with signaling deprived. Indeed, when the best performing individuals are compared, it turns out there is no difference in performance between signaling enabled and deprived.

When robots have the possibility to access useful information through camera in task 2 (D1=1), it is clear this condition provides a suitable base to use signals in an effective manner. As shown in Figure 5.5, performance with signals is better than without signals. Robots can

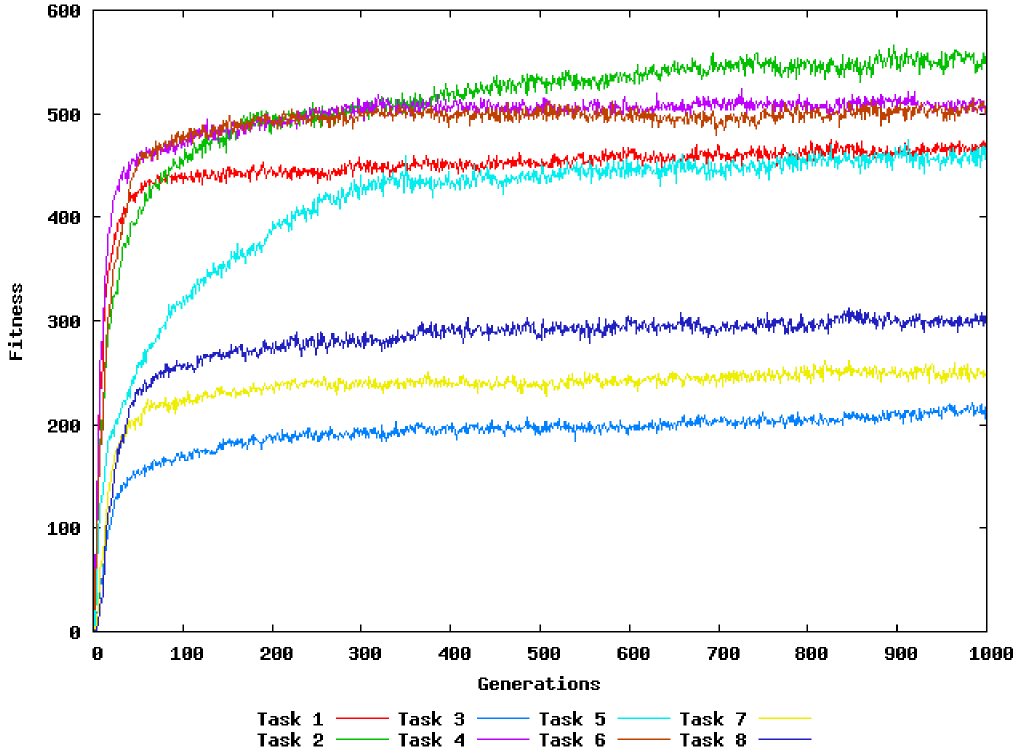
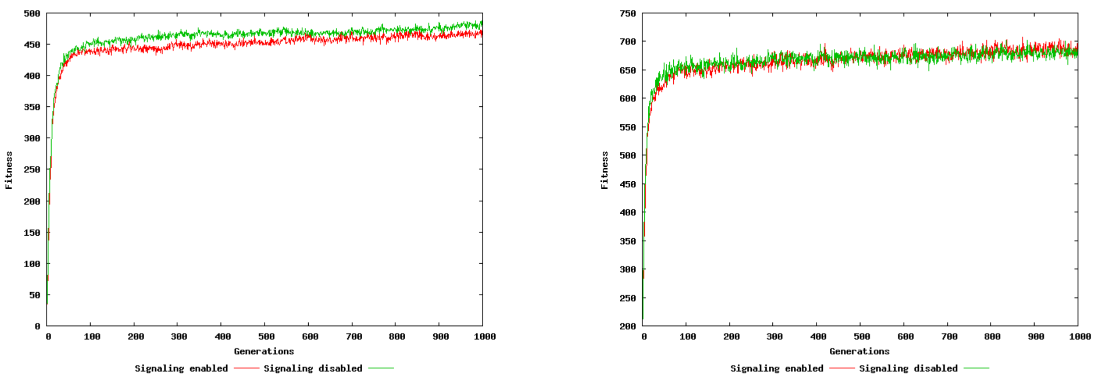
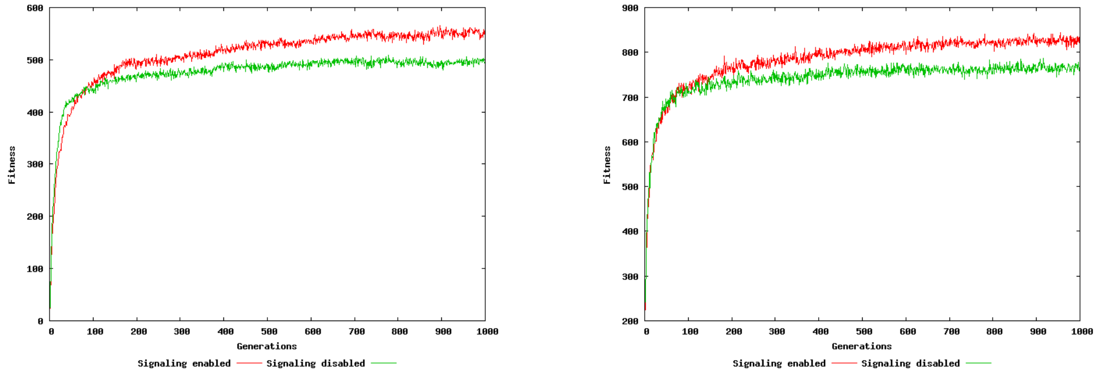


Figure 5.3: *Fitness score of different task setups*



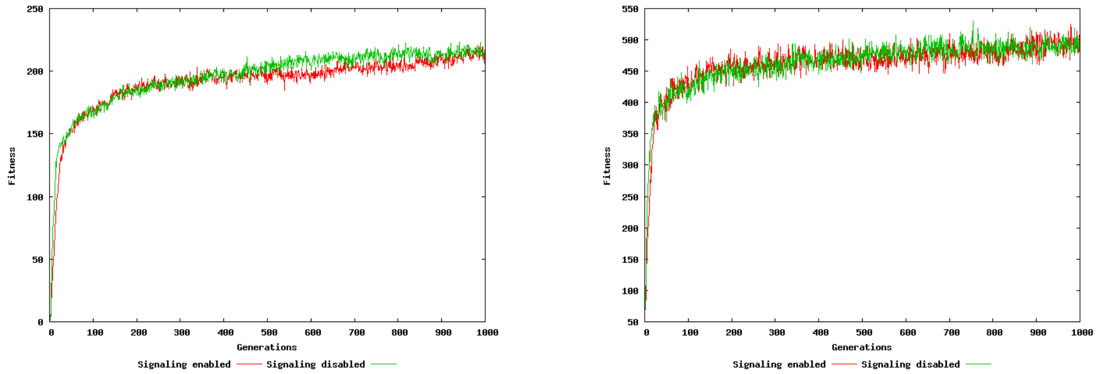
(a) **(b)**

Figure 5.4: *Task 1 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*



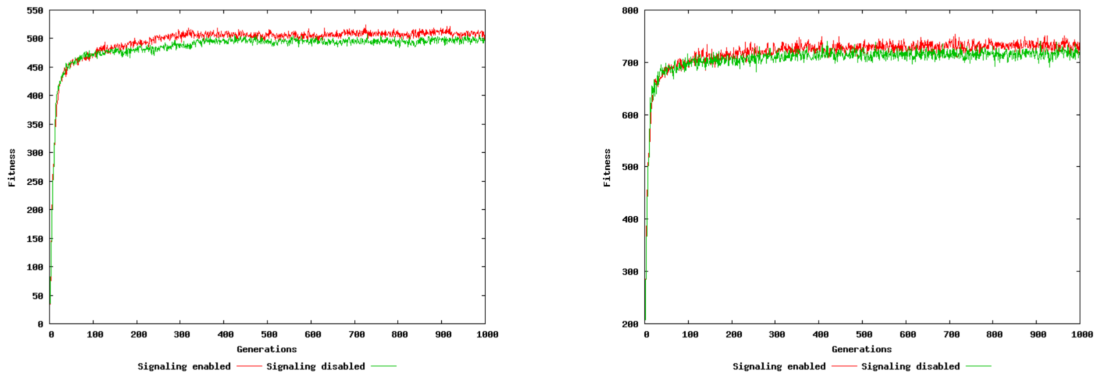
(a) **(b)**

Figure 5.5: *Task 2 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*



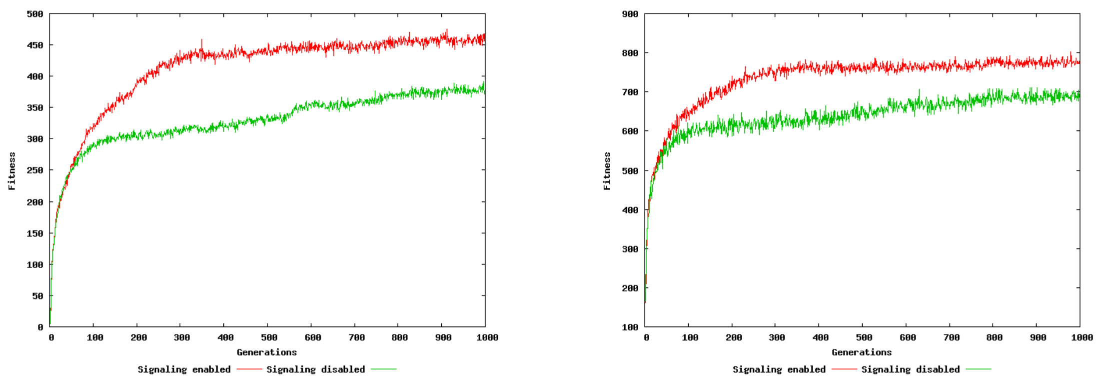
(a) **(b)**

Figure 5.6: *Task 3 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*



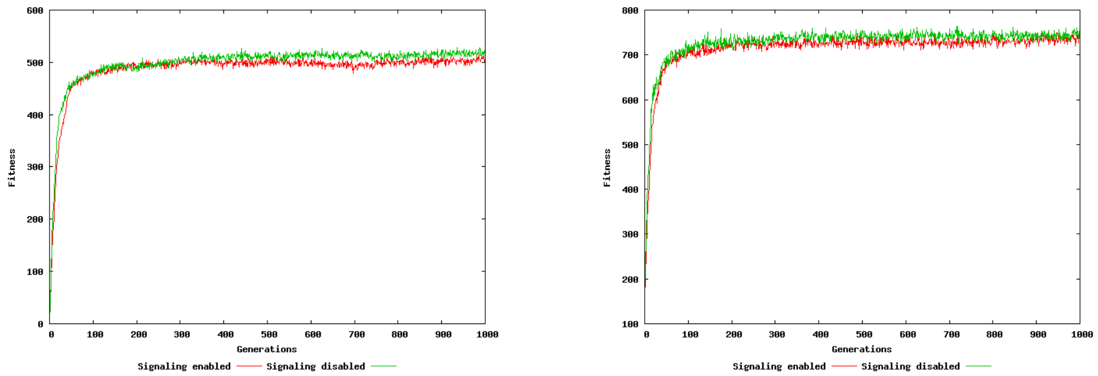
(a) **(b)**

Figure 5.7: *Task 4 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*



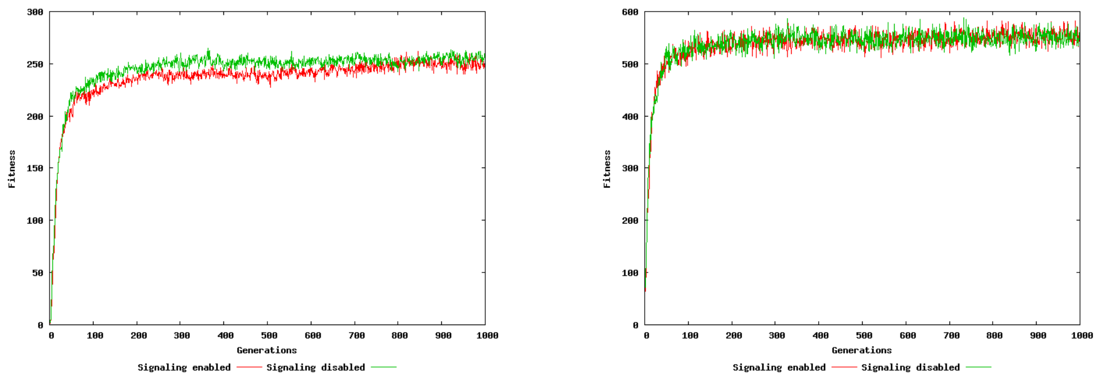
(a) **(b)**

Figure 5.8: *Task 5 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*



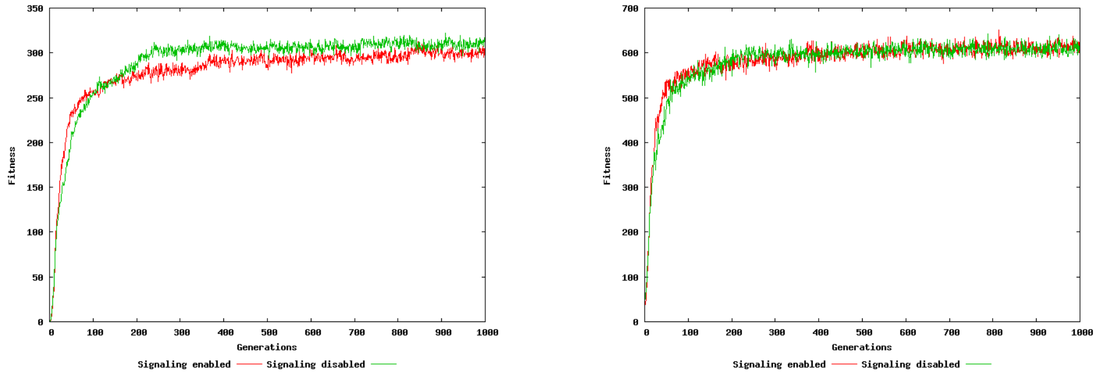
(a) **(b)**

Figure 5.9: *Task 6 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*



(a) **(b)**

Figure 5.10: *Task 7 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*



(a) **(b)**

Figure 5.11: *Task 8 - fitness scores of signaling enabled (red) and deprived (green) for average (a) and best individuals (b)*

perceive other robots so it makes sense to use a signal to convey directional information when a target area is found.

In task 3, where robots also have to occupy a target area together (D3=0) but nevertheless do not have a camera, performance between signal and no signal is equal again (Figure 5.6). Like in task 1, some disturbance causes the non signaling group to perform slightly better on average, but this is not the case when the best individuals are compared. So here the fact that robots do need to relay on cooperation (D2=1) does not boost the use of signals.

In task 4, where robots have to occupy a different target area (D3=1) but have no camera, the use of signals improves the fitness score slightly (Figure 5.7). This improvement can also be observed when the best individuals are compared. When examining the different strategies evolved, it turns out that robots use a signal to drive other robots away when they are both located on the same target area. This happens for one target area only in most replicas. When the two robots are located on the other target area they simply do not use a signal. When they do use a signal to drive the other robot away, the signal cannot hold any clues about the direction of the other target area. Nevertheless it may help the robots to determine if they are both on the same target area which is not good.

When robots both have a camera and need to rely on cooperation as in task 5, the use of signal provides a clear advantage (Figure 5.8). Camera information can be used to locate the proper direction of a found target area, while at the same time pressure to cooperate is high because fitness is only scored when both robots occupy the target areas. In theory this is the situation in which the most can be gained from communication, which can be concluded indeed from the relatively large difference in fitness score between the signaling and non-signaling group.

In task 6 robots do have a camera but can not effectively convey directional information because they need to be on different target areas. Furthermore, pressure is lessened because they do not need to rely on tight cooperation. This explains why performance between signal and no signal is almost equal. The slight difference which can be observed in Figure 5.9 could be due to the fact that the use of signals does disturb robots a little, as is the case in task 1. Likewise, there is no difference in performance between the best individuals with signaling enabled and signaling deprived.

In task 7 robots again do not have a camera but do need to cooperate. Opposed to task 4, the use of signal does not improve performance (Figure 5.10), even though signals could be employed in a likewise fashion to drive other robots away from target areas.

Also in task 8 the use of signals does not improve performance but instead a slight disturbance can be observed (Figure 5.11). This disturbance is not observed when the performance of the best individuals are compared. Robots do have a camera, but this cannot effectively be used in combination with signals because robots need to be on different target areas. Camera is used to avoid the other robot, but for this behavior no signal is required (robots just avoid each other all the time), so the signal-deprived group can use this strategy too.

When robots are deprived of using sound signals, they sometimes use infrared manipulation to drive other robots away from a target area when they are required to each occupy a different target area. This happens occasionally. The fact that fitness score of the signaling and non-signaling group is equal is some tasks can be explained by the fact that the signaling group is not always able to exploit signaling well enough the determine if the robots are located on the proper target area or not.

Signals and Task Space Dimensions

From the results it is clear that signals do improve the performance when information is available for other robots in a useful manner. That is, when both robots need to be on the same target area and they have the possibility to perceive each other. If this is the case, they make effective use of signals to communicate the direction of the target area. However, when each robot needs to be on a different target area robots still make use of the camera by avoiding the other robot, but this does not boost the use of signaling anymore.

The fact that robots need to cooperate does influence the pressure to use communication. This can be concluded when comparing the difference in performance in task 2 and task 5. In both cases the ability to use signals improves performance, but when the robots need to cooperate there is more pressure to use communication and hence a bigger difference between the signaling and non-signaling group can be observed.

Even though it was hypothesized that the possibility to interact locally through infrared sensors would influence the use of signaling, this cannot be concluded from the observed results. Signals could be employed to verify that the same target area is occupied when both robots have found a target area and the objective is to occupy the same target area (as in task 1,2,3 and 5). However, robots mostly solve this by simply ignoring one target area and only occupying the other. This strategy is effective as well and furthermore can also be employed without using communication. Even though less fitness can be scored when only one target area is used, this strategy is easier to learn which is probably why it is the outcome of most replicas. Likewise, signals could be used to drive other robots away when different target areas need to be occupied. This does happen occasionally but mostly only for one target area. Since it is still possible that two robots occupy the other target area for which they have not learned to drive each other away, this strategy is not so effective that it actually improves performance over the non-signaling group. This is somewhat contrary to expectations.

In general it can be concluded that the possibility for robots to have information useful for others (D1) triggers the use of communication. However, this dimension cannot directly be translated in the implementation of equipping the robots with a camera, because the effectiveness of this camera is also influenced by the third dimension (D3: local interaction is impossible). When D3 is in effect, robots cannot effectively hold information for other robots, thus D1 is rendered less influential. It could be argued that robots could still use camera and signaling to indicate that they have found a target area and thus the other robot should avoid them, but this strategy was not observed in the results.

The requirement to use cooperation (D2) seems to boost the use of communication, but only when D1 is also in effect. In cases in which D1 is 0 (robots have no useful information for others), the requirement to cooperate does not seem to boost communication because there is no difference in performance between task 1 and 3 and between task 6 and 8. Thus, the effect of D2 also seems to rely on the state of D1.

In summary: D1 = 1 boosts communication, but only if D3 is 0. D2 = 1 boosts communication, but only if D1 is 1 and D3 = 0.

Implementations

Of course, the implementation of the three dimensions can be subjected to discussion. ‘D1: robots can have information useful for other robots’ can be implemented in many ways, the use of camera being only one of them. Still, it appears to be a useful implementation with the connotation that its effects are influenced by ‘D3: local interaction is impossible’.

Also, it was hypothesized that the impossibility to use local interaction through infrared sensors (D3) would boost the use of signaling because in every task when robots have found a target area they need to figure out if they should stay or abandon the target area, depending on which target area is occupied by the other robot. So it is was assumed that sound signals could be of use. Nevertheless, it turns out that when robots do need to be on the same target area they solve this often not by communicating about it, but by simply ignoring one target area and thus ensuring that they always will be searching for the same target area. When they need to be on separate locations the possibility of using signals still does not boost performance, mostly because robots learn to drive other robots away on one target area only but not on the other.

Likewise, the way ‘D2: cooperation is required’ is implemented can be subjected to discussion. It could be argued that when D2 is 0 (cooperation not required), the performance of the team of robots should be totally independent of the level of cooperation the two robots display, i.e. the task should not require any cooperation at all. However, the way this dimension was implemented, total fitness could still be increased by cooperating but it was not absolutely necessary to cooperate in order to score some fitness. So, suppose D2 was conceived in a more rigid manner. In the task setup as used here, this would mean that a single robot could score all fitness by occupying a target area independent of the actions of the other robot. This however, would yield a problem for the implementation of D3 because the ‘robots need to be on different locations’ requirement has no meaning anymore when one robot already is sufficient to score all fitness. Therefore, a less rigid implementation of D2 = 0 was conceived in which cooperation was not totally not required, but only not strictly necessary. Nevertheless, the relative small influence of D2 found in experiments might be bigger when a more rigid implementation is used.

5.3 Second Series of Experiments

The second series of experiments are performed in EvoRobot in order to investigate how a rich communication system can be developed and what the characteristics of this communication system and the behavioral skills of the robots are. Results show that robots are able to employ such a rich communication system in which sound signals, camera information and infrared sensor readings are tightly interwoven to solve a spatial coordination task. The evolved communication system includes use of explicit and implicit communication channels, switching between different functional roles and selective attention mechanisms that allow the robots to focus on relevant information and to ignore non-relevant information.

5.3.1 Experimental Setup

In this experiment robots have to find two target areas and engage in a switching behavior. In order to perform a switch from one target area to the other it is important to know the location of the other target area. Robots are equipped with a camera which allows them to see the other robot but they cannot perceive a target area with it. Only a ground sensor provides this information when robots are located on a target area. Thus, communication can be used to signify the location of a target area using a combination of information from camera input and ground sensors. This experimental setup realizes an increased level of task complexity. Robots need to perform a relative large number of individual behaviors like world exploration, target area location, signaling when a target area is found and entering and exiting a target area at the appropriate moment. Furthermore multiple sources of information need to be combined; ground sensors to perceive a target area, camera to perceive the other robot and signals to coordinate actions. Also, adequate performance crucially depends on tight coordination between the robots.

Two simulated robots are placed in a 110x110 cm. world. In this world there are two target areas placed at random positions. The task of the robots is to locate these target areas and to occupy them. Once they each occupy a different target area robots have to switch targets as often as possible. A switch is conceived as robot R1 coming from target T1 occupying target T2 and robot R2 coming from target T2 occupying target T1. The performance of an individual (comprised of two identical neural networks controlling the two robots) is measured over different trials and averaged in order to rule out 'lucky' scores of individuals who happen to be placed in positions from which fitness is easily achieved. Thus, the fitness function is as follows:

$$f(x) = \frac{\sum_{i=1}^{T} O_i + S_i}{T} \tag{5.3}$$

Where $f(x)$ is the fitness score of the individual, O_i is 1 when the two robots occupy a different target zone each for the first time in trial i, S_i is the number of combined switches made by the team in trial i and T is the total number of trials for which the individual was tested.

Robots are equipped with a camera which allows them to perceive the relative direction of the other robot if it is in the camera's field of view (36°). This camera provides three inputs (C_1,C_2 and C_3). The first input encodes the relative angle of the other robot in the right visual field (with 1.0 activation when the other robot is at 1° right declining to 0.0 activation when the other robot moves out of the right field of view ($> 18°$)). Likewise the second input encodes the relative angle of the other robot in the left field of view in a mirrored fashion. The third input has a 1.0 activation when the other robot is not perceived in either the right or left field of view. See Figure 5.1 for a graphical representation.

Next to this robots have the ability to send and receive signals anywhere in the world. These signals consist of real numbers with values ranging from 0.0 to 1.0. The signal of the other robot is encoded as one input unit and and another input unit encodes the robots own signal. Furthermore robots have 8 infrared sensors, 2 ground sensors encoding binary if a robot is on target 1 or target 2 and 2 'memory' sensors encoding binary which target area was visited last. All inputs are fed into a neural network which consists of an input layer, a hidden layer and an output layer. The hidden layer consist of 4 hidden neurons with connections to themselves, from the input layer and to the output layer. The input layer is also directly connected to the output layer. The output layer specifies the motor output of the left and right wheel of the robot and of the emitted communication signal (Figure 5.12). The sensory and internal neurons consists of leaky integrator neurons. The motor neurons are activated on the basis of the standard logistic function as described in Chapter 3.

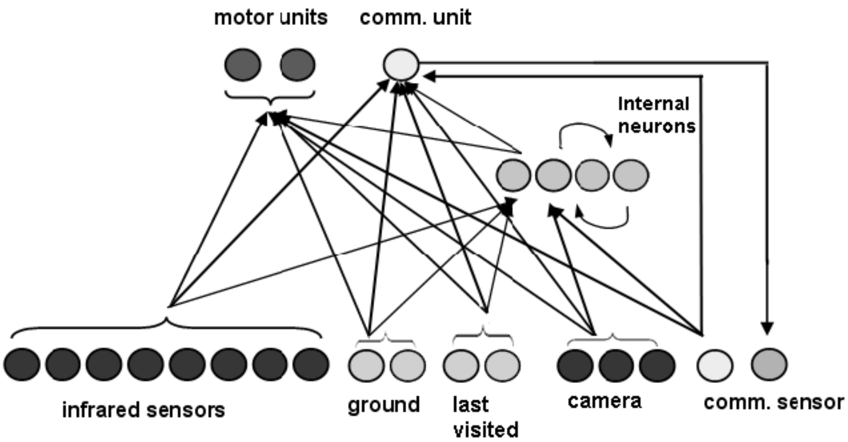


Figure 5.12: *Schema of the neural controller used in the second series of experiments*

The evolutionary parameters are rather similar to those used in the first series of experiments as described in Section 5.2.1, the difference being 10 replicas instead of 15 and a double amount of time steps. For clarity, all evolutionary parameters are mentioned again. The initial population consists of 100 individuals which are successively tested for their ability to solve the given task. Each individual is tested in 10 trials where every trial lasts for 2000 time steps. 20% of the best performing indiviuals are allowed to reproduce with 2% of the bits encoding their neural network weights, biases and time constants mutated. A weight, bias or time constant is encoded in eight bits as a real number ranging from [-5.0 to 5.0] or from [0.0, 1.0]. Every bit is subjeced to a 2% change of mutation with a 50% change of bitflip when it is mutated. The population is evolved for 1000 generations. 10 replications of this experiment are made.

5.3.2 Results

In different replicas some rather effective strategies evolved. See Figure 5.13 for fitness scores of the best and worst performing replicas. The best performing replica and a replica with intermediate fitness score displayed the most interesting communication systems in terms of complexity. Therefore the communication systems evolved in these replicas are discussed in detail. The intermediate performing replica is referred to as 'Replica 1' and the best performing replica is referred to as 'Replica 2'.

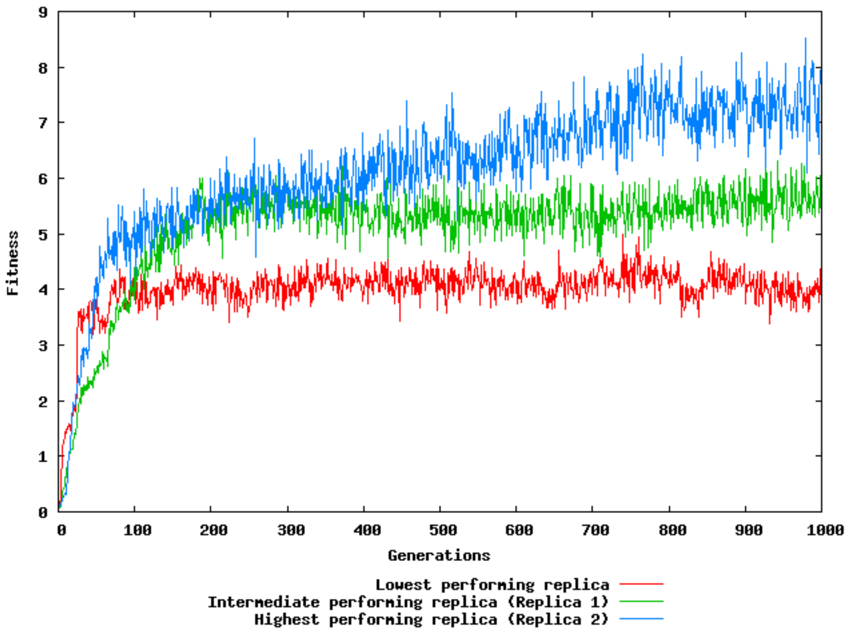


Figure 5.13: *Plot of fitness scores from three replicas*

Compared to an experimental setup in which the use of signaling was deprived, it is clear that the possibility to use signals improves performance (see Figure 5.14).

Replica 1

In this replica robots solve the task by developing an elaborated signaling and guidance strategy. At the beginning both robots explore the environment by avoiding walls and by producing semicircular trajectories. When a robot has located target area 1 it remains there until the other robot has located target area 2. When the second robot also encounters target area 1 before target area 2, it just moves on. When target area 2 is found first it is also ignored. Once both target areas are located one robot stays as a beacon on target 1 while the other robot approaches the beaconing robot (Figure 5.15: a). Once both robots are on the same target area (Figure 5.15: b), the approaching robot steers the beaconing robot away in the direction of the other target area (Figure 5.15: c). The departing robot then heads for the other target area and the steering robot takes the beaconing role (Figure 5.15: d). This behavior is mediated through signaling, the use of camera information to locate the other robot and infrared sensor information to guide the other robot toward the proper direction.

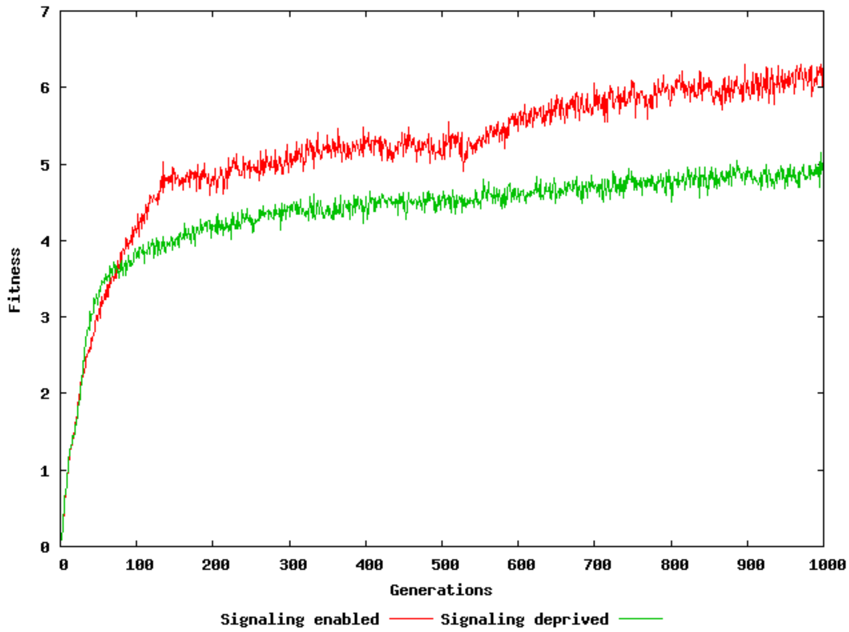


Figure 5.14: *Fitness scores of signaling enabled (red) and deprived (green)*

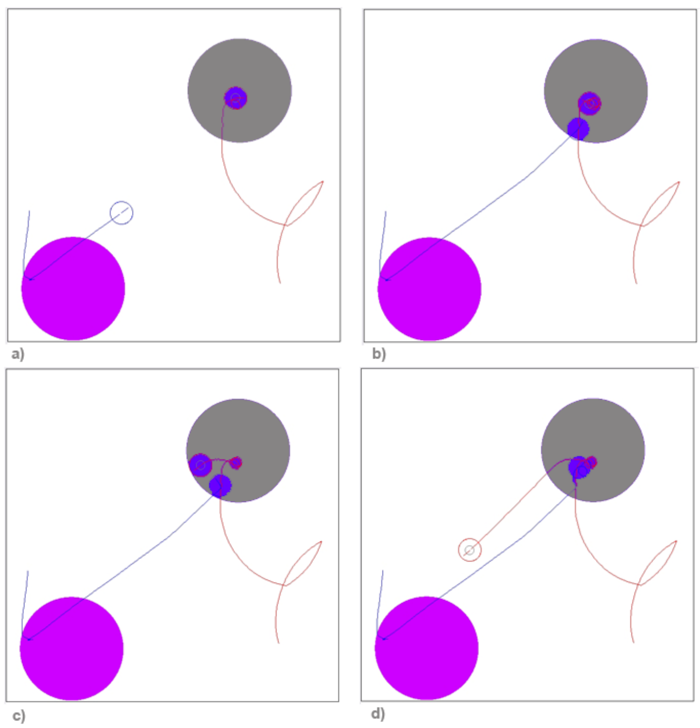


Figure 5.15: *Behavior evolved in replica 1*

Behaviors The following individual behaviors are performed by the robots while solving the task:

1. Exploring the environment by producing semicircular trajectories and by avoiding walls by turning with an angle of about 90° when they are encountered.
2. Exploring the environment by producing straight trajectories and by avoiding walls by turning with an angle of about 90° when they are encountered.
3. Avoiding the other robot by turning clockwise when it is perceived through camera until the other robot is out of sight.
4. Remaining on target area 1 by circling on the spot.
5. Turning on target area 2 until the other robot is perceived through camera.
6. Exiting from target area 2 and moving in the direction of the other robot.
7. Remaining on target area 1 while the other robot enters target area 1.
8. Coordination when entering target area 1 with the robot already occupying this target area.
9. Coordination with the entering robot to acquire the direction from which the entering robot arrived.
10. Exiting from target area 1 by moving straight ahead.

Signals In order to mediate effective execution of this strategy a different number of signals are used by the robots. They affect each other by producing explicit signals, implicit signals and a combination of implicit and explicit signals. Explicit signals consist of sound signals with different frequencies that effect the other robot's behavior in a functional manner. Implicit signals are motor actions executed by a robot that by modifying the perception[16] of the other robot produce a functional modification of the other robot's behavior. Note that robots always influence each others' behavior through sound output, also in initial generations. However, such immature signals do not tend to trigger functional behavior (yet). Below the signals produced by the robots are summarized (see also Figure 5.16):

- A A sound signal with a frequency of 1.0 that is emitted by a robot occupying target area 1.
- B A sound signal with a frequency that increases from 0.1 to 0.9 that is emitted by a robot entering target area 1 which is occupied by the other robot.
- C A sound signal with an frequency oscillating around [0.9] that is emitted by a robot which has just entered target area 1 and has perceived the other robot through its infrared sensors.
- D A sustained activation of the infrared sensors of a robot occupying target area 1 combined with the detection of signal **C** produced by the entering robot that emits this signal and approaches the occupying robot.

[16]Perception that is different than the dedicated 'listening' ability, like infrared or camera information.

With normal exploration no signal is emitted. Once target area 1 is found, signal **A** is emitted. Once robot 1 is entering target area 1 which is occupied by robot 2, robot 1 emits signal **B** while it is approaching robot 2. Once robot 1 perceives robot 2 through its infrared sensors, it emits signal **C** and executes **D**. Robot 1 then performs coordinated alignment with robot 2. When this is done robot 1 emits signal **A** and robot 2 then drops to 0.0.

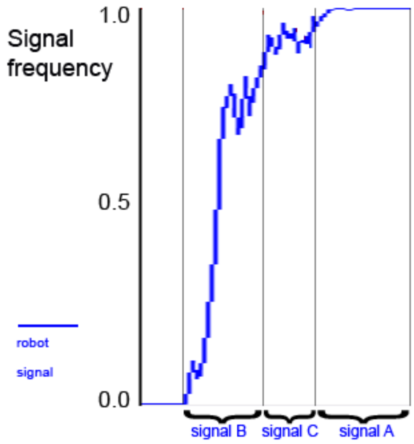


Figure 5.16: *Frequency of different sound signals observed in replica 1*

Effect of Signals on Behavior

A This signal has two effects:

   1 It triggers a specific exploration behavior (based on straight movements and obstacle avoidance behavior) in robots located outside target areas.

   2 It trigger a behavior that consists in exiting from target area 2 and moving toward the direction of the other robot located in target area 1.

B This signal causes the robot occupying target area 1 to stop turning on the spot and move more straight, usually in the direction of the other entering robot.

C This signal causes the occupying robot to remain close to the entering robot and have itself 'guided' in a turn of about 90°.

D This signal facilitates the turning movement executed by the entering robot in order to guide the occupying robot into the proper direction (that of the other target area). Sound communication (signal **C**) and behavioral response (signal **D**) produce an integrated dynamical process which allows the entering robot to convey information about the direction of the other target area to the occupying robot.

Functional Analysis The communication behavior mediated by signal **A** embodies a selective attention mechanism. The detection or lack of detection of this signal regulates whether or not the position of the other robot is used to obtain information about the location of the target area. It also combines two communication channel: a sound channel which is used to communicate whether or not a robot is located on a target area and a visual channel through which the relative position of the target area is conveyed. With regard to the effects of signal **C** and **D**, it also shows that robots effectively integrate two communication channels, sound signals and infrared signals, to convey information about the direction in which the departing robot can find the other target area. The direction of the other target area is encoded implicitly in the orientation of the entering robot, the target area which it just left is at its back. This information is communicated to the occupying robot through a combination of sound signals and spatial alignment mediated by the state of the infrared sensors and by the effect of movements on the infrared sensors of the other robot. The information about the location is not directly encoded in the sound signal (signal **C**), nor is the triggered behavior by the occupying robot enough to find the target area on its own, but the sound signal is used to achieve effective alignment of the two robots which makes the transfer of information through infrared (signal **D**) possible. This coordinated behavior makes the second robot turn toward the direction from which the entering robot arrived.

Evolutionary Overview The strategy of this replica developed over generations as follows. During the first generations (1-10) the robots learn to avoid walls and each other and remain on a target area occasionally. By generation 25 robots have learned to wait on target area 1 until the other robot has found target area 2 and has returned to target area 1. A rudimentary signaling system is used then to drive the occupying robot away in a random direction. Camera input is ignored at this point. By generation 50, robots use vision to identify the position of the other robot which functions as a beacon then, but approaching the beaconing robot happens rather slow. Occupation of a target area has become somewhat sloppy, sometimes the beaconing robot leaves the target area too early. After 100 generations the use of camera information happens more smooth and beaconing robots stay on the target area all the time but not on the same place. After 200 generations the general switching strategy is performed more smoothly, and furthermore the approaching robot attempts to send the beaconing robot into the proper direction. This succeeds occasionally, but is obstructed sometimes because the beaconing robot does not make way for the approaching robot. From 200 to 1000 generations the same behavior does not change very much but is improved slightly. The sending into the proper direction through infrared manipulation is improved, robots move faster in more straight lines and do allow another robot to enter a target area better. This behavior remains stable for up to 1000 generations.

Replica 2

The strategy developed in replica 2 is different from the one in replica 1 because the switch between the two target areas is performed in synchronization by the two robots. They achieve this by manipulating each others' turning behavior at the right time. First both robots explore the environment in straight lines (Figure 5.17: a). Once one robot is located on a target area, it produces a signal which changes the other robot's trajectory into a curve (Figure 5.17: b). When both robots are located on a target area at the same time, they each produce a signal which causes the other robot to turn sharply on the target area (Figure 5.17: c, note that both robots have moved further with respect to their positions in the preceding figure).

The behavior of turning on the target area and leaving it in the direction of the other target area is based on a combination of two communication channels: sound signals that provide an indication of whether or not the other robot is located on the other target area and visual signals that (in combination with the sound signals) encode information about the relative position of the other target area. Hence, both robots are used as a beacon which signifies the location of the other target area (Figure 5.17: d). Thus, robots are able to engage in a highly effective changing behavior which provides a near optimal solution for the task.

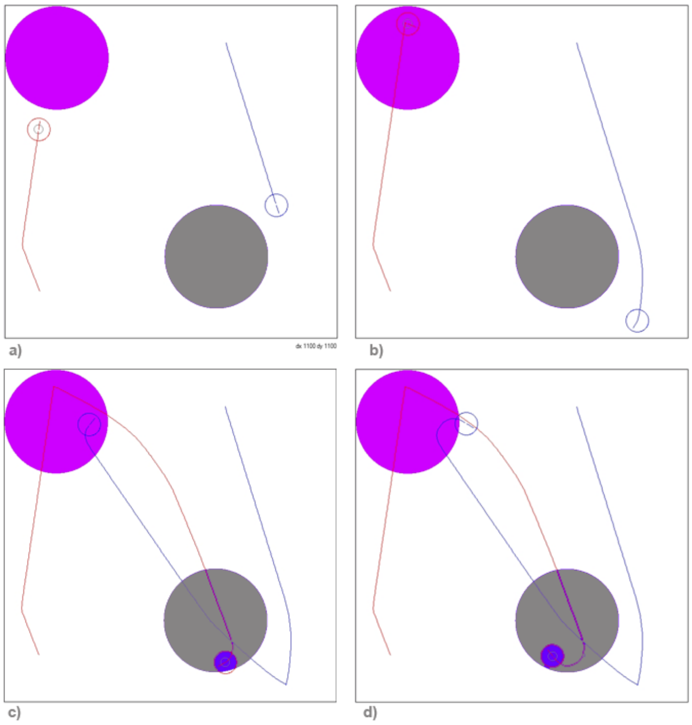


Figure 5.17: *Behavior evolved in replica 2*

Behaviors The following different behaviors are performed by the robots while solving the task:

1 Exploring the environment by producing straight trajectories and by avoiding walls and other robots by turning with an angle of about 90° when they are encountered.

2 Turning clockwise slightly while exploring the environment.

3 Turning clockwise more sharply when located on target areas.

4 Turning clockwise very sharp on target areas when the other robot is perceived through camera.

5 Focusing on the other robot through camera and moving toward it by keeping it in the right visual field.

6 Exiting a target area and moving straight.

Signals To effectively engage in a synchronized changing pattern the two robots produce a variety of signals. Most signals are used to manipulate the behavior of the other robot. Below the most important signals produced by the robots are summarized (see also Figure 5.18):

A A sound signal of 0.8 or 1.0. emitted by robots doing normal exploration. The signal is 0.8 when no target area has been encountered yet. When target area 1 has been encountered the robots emit 1.0, when target area 2 has been encountered the robots emit 0.8.

B A sound signal of 0.4 or 0.3, depending on the target area. This signal is produced by robots which are located on a target area while the other robot is located outside any target area. The signal is 0.4 when target area 1 is occupied and 0.3 for target area 2.

C A sound signal of 0.5 that is emitted by both robots when they are each located on a target area and do not perceive each other through camera.

D A sound signal of 0.5 declining to 0.0 when a robot on a target area perceives the other robot in its right field of view. The signal declines from 0.5 to 0.0 as the other robot progressively moves more into the right field of view.

E A sound signal of 1.0 declining to 0.7 when a robot is located on target area 1 and perceives the other robot in its left field of view. The signal declines from 1.0 to 0.7 as the other robot progressively moves out of the left field of view.

F A sound signal of 0.7 declining to 0.0 when a robot is located on target area 2 and perceives the other robot in its left field of view. The signal declines from 0.7 to 0.0 as the other robot progressively moves out of the left field of view.

Signal **D** differs somewhat in the frequency used with respect to the two different target areas. The signal drops more rapidly when the observing robot is on target area 1 and more gradually when the observing robot is on target area 2. Likewise, signals **E** and **F** appear to be in essence the same signal but are operated on different frequency scales with respect to the two different target areas.

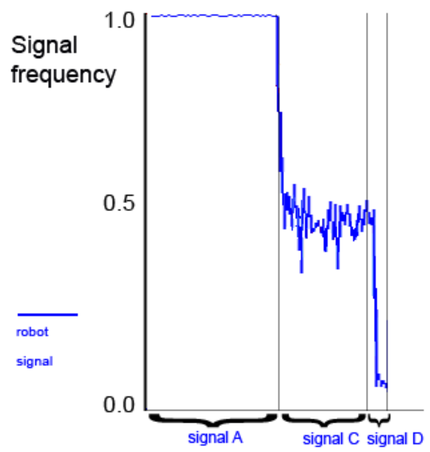


Figure 5.18: *Frequency of different sound signals observed in replica 2*

Effect of Signals on Behavior

A This signal causes the robots to explore in straight trajectories. When both robots are deprived of emitting signals, they will approach each other and keep turning in circles together on the spot. When one robot is deprived of hearing signal **A** from the other robot, the first robot will turn toward the other robot and approach it. So signal **A** causes the robots to ignore each other and explore normally.

B This signal causes the other exploring robot to make a wide clockwise curve instead of exploring in straight trajectories.

C This signal causes the other robot to turn clockwise on the target area but it keeps moving forward as well.

D This signal causes the other robot to almost stop moving forward, start emitting signal **A** and to turn very sharply toward the robot emitting this signal.

E This signal has the same function as signal **A**, it causes the other robot to explore in straight trajectories.

F This signal is a combination from signal **A** and **D**. It causes the other robot to explore normally when the frequency is high, and making it turn when the frequency drops. It has no clear function and could very well be a rudimentary leftover from the evolutionary process.

Functional Analysis Robots explore the environment in straight trajectories under the influence of signal **A**. Once they locate a target area they manipulate the other robot with signal **B** to alter its straight course and have it turn clockwise a bit. This causes the robot which is not located on the target area to alter its course slightly in the direction of the target area while at the same time the signaling robot moves away from this target area. This increases the chance that the robots will be divided equally in the environment and hence increases the chance that they will each occupy a different target area at the same time.

Once both robots occupy a different target area they synchronize their movement through signal **C**. This signal functions as a selective attention mechanism because it triggers the robots to pay attention to the other robot through camera as opposed to ignoring it first. Both robots cause the other robot to turn more sharply until the other robot is perceived through camera. When one robot perceives the other through its right visual field, it will emit signal **D** and move forward. The other robot which does not yet perceive the first robot visually changes from signal **C** to signal **A**. The **D** signal causes the other robot to turn toward the robot producing this signal and move toward this robot while at the same time it 'responds' with signal **A** causing the first robot to move straight. The effect is that when both robots perform this at the same time they turn toward each other and move forward. Once the robots are outside the target areas they presume normal exploration without any indication about the direction of the other target area, but because they are aligned based on the position of the other robot, they are able to reach the other target area.

Because this turning behavior is always performed clockwise, other robots will always show up in the right field of view first. If this happens for both robots at the same time, it will both cause them to stop turning and move forward. In this way it is possible for the robots to each use their own 'lane' to move to the other target area and by keeping the other robot in the right field of view they can avoid collision on the way.

With respect to signal **E** and **F**, when the strategy described is executed properly these two signals are not used because the other robot is never perceived in the left field of view. So although being emitted on different frequencies under different conditions, they do not have an explicit function, but are merely a rudimentary by-product of the evolved solution.

Evolutionary Overview In the initial generations (1-10) robots learn to avoid walls and each other and occupy a target area once they encounter it until the other robot also enters this target area. Then they use signals until one robot leaves the target area. From generation 10 to 25 robots lose the ability to wait on a target area until they receive a signal from the other robot. Instead, they keep moving over target areas and explore the environment in straight lines, but they do use a signal when they move over a target area to alter the course of the other robot into a curve. From generation 25 to 50 the behavior remains rather stable. Robots make sharper curves when they are influenced by the signal of the other robot on a target area, but camera is not used. In 100 generations robots have learned to use camera information as well in order to determine the direction of the other target area. This happens on one target area only, on the other target area the robots keep moving in straight lines. After 200 generations robots still use the camera information on one target area only, but the strategy has increased in smoothness. After 500 generations the robots have learned to use camera information to localize the other robot/target area for both target areas. The signals of the communication system are also more clearly defined. Up to 1000 generations this behavior is improved slightly but in general remains stable.

Differences and Similarities in the Two Replicas

Within the two different evolved strategies a number of differences and similarities exist.

Differences:

- The number of different information bearers. In replica 1 three signal channels can be distinguished: sound signals, infrared manipulation through movement and directional information through camera. In replica 2 only sound signal and directional information through camera is used. Infrared manipulation through movement is omitted, although robot do use infrared sensors to avoid collision. This can be explained by the fact that in replica 1 robots are located together on target areas, while in replica 2 they only perceive each other through infrared occasionally. Hence, this might have caused the use of infrared channels in replica 1 to convey directional information, while in replica 2 this feature is not exploited.

- It can be argued that replica 2 is more optimal than replica 1, because in the former the robots move in parallel and no time is 'waisted' on waiting for the other robot to move. This is acknowledged when average fitness scores are compared; replica 2 scores on average 16% better than replica 1.

- In replica 1 four different functional roles are performed by the robots; a 'beaconing' robot which guides the other robot toward it, a robot homing on this beacon, a 'pointing' robot which steers the other robot in the right direction and a following robot having itself steered in the right direction. Robots switch between these roles during the trial, depending on the situation. In replica 2 two both robots perform two roles simultaneously; they both act as a beacon and actively home in on the beaconing robot at the same time.

Similarities:

- In both replicas a mechanism of selective attention can be observed. The information about the location of the other robot is always present (red LEDs are always on), but this information is only used when the robot is triggered through sound signals to pay attention. When not triggered to pay attention, this information is used differently (avoidance) in replica 1 or ignored in replica 2.
- In both replicas other robots are used as a beacon to determine the direction in which the proper target area can be found. In replica 1 a robot will 'home' in on the other robot, while in replica 2 both robots try to keep the other robot in the right field of view only. The function is alike though: the angle at which the other robot is located is used to determine the proper direction of movement.
- Signals are used to manipulate the movement of the other robot. In replica 1 the occupying robot manipulates the other robot through signal **A** by causing it to explore in straight trajectories. In replica 2 both robots cause each other to make turns of various sharpness when they are located on target areas.

5.3.3 Concluding Remarks

Both replicas show that a team of autonomous robots is able to integrate different signal channels (sound, visual, infrared) in order to achieve an effective communication system which allows them to solve the task. Sound signals are used to mediate the overall behavior and the transfer of information over other signal channels. In order to do so four different sound signals can be distinguished. The sound signaling system is both rich in the number of different signals and their effect on the behavior of robots and it is tightly integrated with other signal modalities. This shows that robots are able to combine sensory inputs of different information bearers in a functional manner. Furthermore robots display the ability to selectively take into account or ignore the information coming from the other robot and the ability to switch to different functional roles. Overall the evolved behavior can be conceived as more complex indeed.

5.4 Summary

Two series of experiments were performed in EvoRobot. The first series shows the influences of the three different dimensions from the task space model on the use of communication signals. Some dimensions clearly affect the development of communication but the dimensions influence each other, making it hard to study them in isolation. The second series of experiments shows how an increased task complexity may lead to a communication system with increased complexity. Two different communication systems resulting from a more complex task setup were discussed.

Chapter 6

Application in e-Puck Robots

6.1 Outline

This chapter describes the process of transferring obtained results from the experiments described in Chapter 5 onto a robotic hardware platform, the e-Puck robot. First, the characteristics of the e-Puck robot are described. Then some practical issues concerning the transfer are highlighted, followed by a description of the resulting performance of two e-Puck robots executing behavior evolved in EvoRobot.

6.2 The e-Puck Robot

The e-Puck robot is a miniature mobile robot designed by the Ecole Polytechnique Fédérale at Lausanne "for educational purposes at university level" under an open source hardware license[17], allowing anyone to contribute to the development. The purpose is to provide a small autonomous robot which has a good structure, is flexible, user friendly, has good robustness, simple maintenance and is relatively cheap so it can be used in various research enterprises or serve as an educational tool[18]. The e-Puck robot resembles a Khepera robot in its dimensions and capacities but it is newer and hence incorporates equal performance and modern features (like Bluetooth) at a lower price.

An e-Puck robot consists of a spherical body of 75 mm in diameter, see Figure 6.1. It has several outputs which can execute behavior or 'broadcast' information into the environment:

- Two wheel motors, which can rotate independently.
- Eight LEDs placed around the body, which can be turned on and off independently.
- Two body LEDs, red and green.
- A speaker, which can play sound files.
- A Bluetooth module to communicate with a PC or other e-Pucks using designated Bluetooth protocols.

[17]The license can be found at `http://www.e-puck.org/index.php?option=com_content&task=view&id=18&Itemid=45`.

[18]Free from the spirit of the e-Puck project stated at `http://www.e-puck.org/index.php?option=com_content&task=view&id=13&Itemid=31`.

- A RS-232 connection for serial communication with a PC or debugger.

To perceive its environment the e-Puck is equipped with the following sensors:

- Eight infrared sensors which have dual purpose, they can be used as proximity sensors to perceive obstacles or as light sensors to perceive luminosity.
- Three microphones, which can detect sound.
- An accelerometer, which measures relative movement in three dimensions.
- A camera, which typically captures a subsampled 40x40 pixel image at 4 fps in color or 8 fps in grayscale mode. Different dimensions may be specified for other research needs.
- A group of three infrared sensors pointed down, which can be used to detect different surfaces or drops.

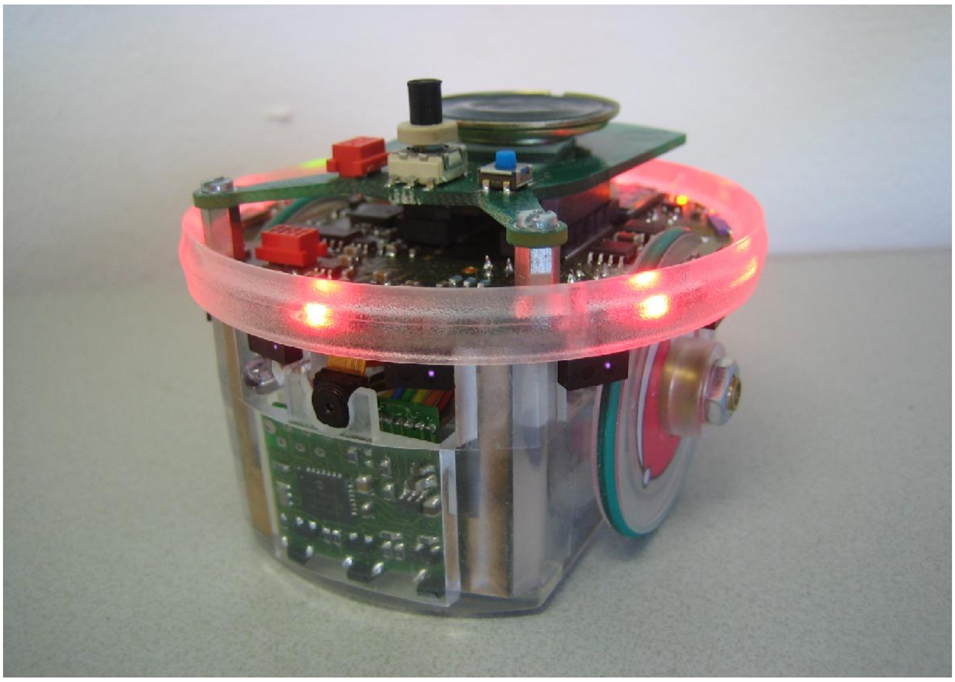


Figure 6.1: *Picture of the e-Puck robot*

6.3 Transfer from EvoRobot to e-Puck

6.3.1 Two Approaches

In order to transfer an evolved behavior from the EvoRobot simulator onto an e-Puck robot two approaches can be considered. The first method consists of uploading the full set of weights, biases and time constants encoding the evolved behavior of an individual (denoted as ϕ) onto the e-Puck. Having received this evolved 'brain' of an individual, the e-Puck should perform in interaction with the physical environment a behavior that is similar to that exhibited in simulation. However, differences between the real and simulated robot and environment can lead to differences in the observed behavior. This approach requires the e-Puck to be able to parse ϕ in order to display the proper behavior. Advantages of this approach are its 'natural' appeal in the sense that each e-Puck is truly autonomous and the fact that there exist no communication lag because once ϕ has been received no communication with a computer is necessary. Drawbacks are the lack of control, because an e-Puck robot operates as a black box since only its behavior is observable. Hence it is difficult to visualize and analyze the dynamics of the neural controller in interaction with the environment.

The other approach does not require the uploading of ϕ, but instead sensory data and motor commands are sent back and forth between the e-Puck and EvoRobot. In this way the control of the e-Puck is executed within EvoRobot, using the same routines as with a simulated robot. So, in essence only the virtual robots of EvoRobot are replaced with physical counterparts. Advantages of this approach are the fact that it allows for a high degree of control, because the sensory activation, scored fitness and number of time steps can be monitored within EvoRobot just like with a simulated robot, and the fact that this method is relatively straightforward to implement. Drawbacks are the fact that a communication lag exists, requiring the e-Puck to run at a lower speed in order to effectively send back and forth all sensory information and motor commands.

The second approach was adopted in this thesis because of its two advantages and the fact that the communication lag in general was under 200 ms. allowing for a rather adequate execution when the e-Pucks did not move at full speed (typically at 20% of the full speed). Also, this approach did not require the development of a control program able to parse ϕ to run on the e-Puck, but an existing control program called SerCom[19] was modified to be used in conjunction with EvoRobot. Regarding the communication between robots, with the uploading method a specific routine handling the communication signal would have been needed, where the medium would either be real sound or Bluethooth. By maintaining the communication signal within EvoRobot, no such routine was required because the communication signal could be handled in the same manner as with simulated robots.

6.3.2 Modifications in EvoRobot

Since the ϕ was handled in the same manner as with virtual robots, some modification in EvoRobot were needed in order to operate properly with e-Puck robots. First, the basic functioning of EvoRobot did not include the handing of camera data received from e-Pucks. Thus, a routine was added which parsed the incoming camera information into the three input neuron setup as described in Section 5.3.1. From the e-Puck a value v ranging from [0-60] is received, which encodes for the span of 60 horizontal pixels of the captured camera image. So, $v = 0$ encodes for no perception of another e-Puck robot, $0 < v < 30$ encodes for the perception of another e-Puck in the left field of view and $v >= 60$ encodes for the perception of another e-Puck in the right field of view. This was then translated to three input values corresponding to the three camera input values used in simulation.

Second, the ground sensors of an e-Puck robot consist of a group of three infrared sensors, which measure the luminosity l of a surface, with $l \approx 1000$ for a white surface and $l \approx 200$ for a black surface. Because in EvoRobot two different target areas are distinguished and hence

[19]Developed by the e-Puck development group, downloadable at `http://svn.gna.org/viewcvs/e-puck/trunk/program/advance_sercom/`.

a simulated robot is equipped with two virtual ground sensors for each target area, these two virtual sensors needed to be simulated using data from only one real ground sensor. This was done by using a gray and black surface for the two different target areas respectively. The real ground sensor then measured $l \approx 600$ on the gray target area and $l \approx 200$ on the black target area[20]. Prototype values of the gray and black target areas were acquired during an initialization phase at the beginning of a trial. These prototypes then were compared to actual ground sensor readings, where the smallest difference between prototype and ground sensor reading determined which virtual ground sensor was activated.

Incoming proximity values were boosted to allow for a better detection of walls and other robots.

Since the sound signals where all maintained within EvoRobot, no modifications were needed to handle these.

6.3.3 Modifications in SerCom

SerCom is the program which ran on the e-Pucks. Since it did not include a routine to process camera data for detection of other e-Pucks, such a routine was added. The implemented algorithm scans a captured camera image for the presence of red pixels which are typically present when red LEDs of another e-Puck are perceived. Red pixels are distinguished from other colored pixels by checking if their RGB values are within the range of the color red[21]. The relative angles at which red pixels are perceived are added up and divided by the total number of red pixel observations, yielding the average relative angle of the other e-Puck. Through averaging the relative angle, a fairly correct angle of the other e-Puck is obtained, even if multiple blobs of red pixels (resulting from different LEDs) are perceived. A minimum of two red pixels which needed to be at different horizontal positions is specified in order to counter faulty detection of another e-Puck. The camera is rather sensitive, so sometimes red pixels are perceived even though the other e-Pucks LEDs are not in view. Using a camera resolution of 60 x 20 pixels, the calculated relative angle is encoded in [1 to 60], where 1 stands for maximum left (-18°) and 60 for maximum right (+18°).

Within SerCom a loop was executed which listens for incoming commands and execute behavior accordingly. This loop would parse one character encoding a command at the time and execute the entailed behavior before parsing the next command. This turned out not to function when proximity, floor sensors and camera information were asked for in a subsequent string of command characters. To counter this problem, a routine was added to the main loop which captured proximity values, floor values and camera info using only one command character.

When the e-Pucks would have moved at full speed, there would not have been enough time to process proximity or camera data due to the communication lag between e-Pucks and the computer running EvoRobot, causing e-Pucks to collide with walls and with each other before they could make a turn. Thus, the motor output was adjusted to 20% of the speed the e-Puck robots are capable of, to slow down the e-Pucks so the communication lag did not cause problems.

[20] These values are slightly different for every individual e-Puck robot.

[21] RGB encodes a color by specifying values ranging from [0 to 255] for red, green and blue channels. A pixel typically is red(dish) when the value for the red channel is high while green and blue channels are relatively low.

6.4 Performance of the e-Pucks

6.4.1 Two Evolved Behaviors Tested

The two behaviors described in Section 5.3.2 were transferred onto e-Puck robots. After some testing, modification and tweaking both behaviors observed in simulation could also be distinguished in the behavior of real robots. In general robots would perform rather similar to their virtual counterparts, with the connotation that they 'lost' their way easily. This was mainly due to the fact that both the perception of the other e-Puck through camera and the perception of the two different target areas was much more noisy than in simulation.

The behavior evolved in Replica 1 was executed as follows. Robots explored the environment and once a target area was found robot 1 remained there by moving in circles. When robot 2 found the other target area, it would turn until it had located robot 1 through its camera. Then robot 2 moved toward the occupying robot 1 in a straight trajectory. When the other target area with robot 1 was reached, entering robot 2 attempted to steer robot 1 in the directory of the other target area. This failed in general, most presumably due to the rather delicate use of infrared sensors in combination with a 'sound signal'. Mostly the two robots then got 'confused' about their occupying and exploring roles, but usually one of the robots (not necessarily the robot that just entered) would leave the target area eventually and start searching for the other target area again.

The behavior evolved in Replica 2 was executed in a similar fashion. Robots explored the environment randomly in straight trajectories, sending out a 'sound signal' when they passed over a target area which caused the other robot to turn slightly. When both robots were located on a target area at the same time, they would both turn more sharply and attempt to perceive the other robot through camera. When they occasionally succeeded in perceiving the other robot, they would move in a straight trajectory toward the other robot and hence the direction of the other target area. This behavior was observed frequently, but robots had trouble to execute it repetitively as happening in simulation. Again this was most likely due to noisy sensors and hence a noisy 'sound signal', causing the robots to loose their synchronization and start exploring normally again.

Because the robots moved at a rather slow speed and the fact that they 'lost' their way from time to time, no attempts were made to measure and compare fitness score of the real robots with the simulated robots. Robots in simulation would no doubt score higher, but this would not be so informative as the objective of the transfer was merely to show the feasibility of real robot application.

6.4.2 Problematic Issues

Some problems were encountered while transferring the evolved results onto the e-Pucks. Most of these problems were technical or practical, rather then theoretical of nature. In general the idea of transferring is straightforward, if the simulator matches the real world close enough it is not problematic to run evolved behavior on real robots. However, issues like sensor noise, slipping wheels, incomplete camera images and communication lag did affect performance of real robots. This was especially stringent when evolved behavior relied on a delicate balance of multiple sensory input. With the smallest distortion or variation this balance can be lost. This was most clearly observable in the 'steering behavior' of Replica 1 which relied on integration of two forms of communication, sound signals and infrared manipulation to point the other robot to the proper direction. Because of variations in sensory input along with difference in motor behavior, robots were not able to perform this steering behavior properly.

Also the fact that a minimal detection of two red pixels was required to detect the other robot caused camera information to be less reliable than its simulated counterpart. This minimal requirement was necessary however, to counter faulty detection of the other robot[22]. Another problem which crippled the testing process was the fact that robot batteries ran out rather quickly. This was most likely due to a malfunctioning battery charger.

Most of these problems may be solved to a good extent by optimizing the pre-processing of camera information, the perception of target areas through ground sensors and by adding additional noise to the simulation. However, the transfer from simulation to reality might still produce a certain drop in performance. The amount of performance loss might vary for different evolved individuals, since the impact of variation between the simulated and real environment might have different impact on the robot's ability to solve the task, depending on the strategy adopted by the robot.

6.5 Summary

This chapter described the process of transferring evolved behaviors onto e-Puck robots. It turned out that it is possible to have e-Puck robots execute evolved behavior. Overall the obtained results indicate that the same qualitative behavior observed in simulation is produced by the real robots. However, the performance is not as adequate as it is in simulation, due to slow movement, sensory noise and less informative camera input. Observed behavior was similar to simulation from time to time, but was not executed in a smooth and continuous manner. Nevertheless, the possibility of a transfer from simulation to real environment was shown. It can be argued that this increases the plausibility of the EvoRobot simulator. Also, the analysis of the differences between evolved behavior in simulation and observed behavior in real robots may help to improve the simulator.

[22]The camera occasionally produced some erroneous values, which could also encode for red pixels. Because these erroneous values occurred always in a vertical line of pixels, a minimal requirement of two red pixels in different columns countered this.

Chapter 7

Conclusions

7.1 Summary

This thesis has discussed the development of communication systems in a team of artificial agents by using techniques from Evolutionary Robotics. To obtain effective controller programs for mobile robots the use of evolutionary algorithms on neural networks can be very fruitful. In order to further facilitate the development of well performing control programs communication can be of aid. However, to effectively apply evolutionary techniques there is a need to understand the nature of tasks which can drive the evolutionary process toward using communication. Thus, a research question was formulated as follows: *what is the nature of tasks which may trigger the development of communication?* To investigate this question a task space model was proposed which embodied three dimensions that are hypothesized to be of influence on the use of communication. From this task space model implementations of tasks were derived which through systematically switching on and off the three dimensions rendered eight different task setups. The performance of robots was tested on these eight task setups with signaling enabled and compared to the performance in the same task setup with signaling disabled. The results show that it is hard to view the different task dimensions independently, because the presence of one aspect influences the usefulness of communication in relation to another aspect. Nevertheless, in answer to the research question it can be concluded that the most influential aspect which boosts the use of communication is the possibility for robots to have access to information which is useful for other robots. Other factors like the ability to interact locally through infrared sensors and the need to cooperate are found to be less influential, although it should be noted that the latter indeed depends heavily on the particular implementation.

Inspiration drawn from the first research questions lead to the second research question: *what are the characteristics of evolved communication systems and what level of complexity can be achieved?* In order to investigate this question, various aspects of complexity on both the task setup and the resulting communication system were discussed and based on this a complex task was developed in which two robots had to engage in a switching behavior. This resulted in some rich communication systems which included selective attention, various functional roles and integration of different informational sources. Hence it was shown that robots are capable of developing a highly effective communication system through evolutionary search. Two communication systems of the most effective replicas were discussed. Based on the stipulated aspects of complexity, these communication systems can be conceived as more complex indeed.

The third research question, *can behaviors developed in simulation be transferred onto real robots*, was addressed by transferring evolved behavior from the EvoRobot simulator onto e-Puck robots. After some tweaking and modification, observed behavior in real robots was comparable to the behavior robots exhibited is simulation. However, problematic issues that are rather typical for application in real robots, caused the performing e-Pucks to be not as effective as their simulated counterparts. Nevertheless, it may be concluded that the transfer from simulated to real environment is possible. This strengthen the plausibility of the EvoRobot simulator as well as provide insights for further improvement.

7.2 Discussion and Future Research

The insights resulting from the first series of experiments can help to formulate new effective task setups which trigger the development of communication. However, they are still limited in the sense that they only apply on some aspects of a task with a specific implementation, most notably the fact that robots can have information which is useful for other robots. More extensive studying of different aspects may shed more light on the nature of effective tasks.

The second series of experiments may help to understand what type of communication systems can be evolved and shed light on the relation between communicative, behavioral and cognitive skills. For instance, the experiments in which robots were confronted with the switching task demonstrate how a relatively simple scenario might lead to the emerge of a rather rich communication system, a rich behavioral repertoire, and to complex cognitive skills such as selective attention.

A different important aspect which is still relatively unknown is to which extent the obtained communication systems are able to scale up effectively. Most studies performed so far, including the experiments in this thesis, limit the number of robots to a handful. Also, the repertoire of individual behaviors which need to be developed is limited[23]. It would be interesting to investigate if features of communication systems observed in these smaller models can also be obtained in much larger models involving more agents and more extensive tasks requiring a wide variety of individual behaviors. Related to this, it might also be interesting to study how the simple model proposed in this thesis can scale up with respect to the sensory-motor complexity of the robots, the complexity of the environment or the task to be performed.

Another interesting topic concerns the identification of the conditions that might lead to the emergence of language-like features like compositionality, repetitiveness, advanced conceptual generalization and so on. Communication systems developed so far have more resemblance to animal communication which is reactive in nature. Thus, communication utterances are a direct result of changes in sensory input and hence the meaning is directly coupled to changes in the environment observed by the agent. However, based on the promising results obtained so far it is not too far fetched to hypothesize that robots faced with more complex tasks and equipped with more complex neural controllers may indeed develop communication systems which embody the sophisticated features of human language. Future research may focus on investigating which of those features of human language can be obtained by evolving communication in Evolutionary Robotics.

[23]Although the list of individual behaviors observed in two replicas of the second series of experiments in this thesis is relatively large.

Bibliography

Beer, R. D. (1995). On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3(4):469–509.

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

Brooks, R. A. (1991). Intelligence without reason. In Myopoulos, J. and Reiter, R., editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 569–595, Sydney, Australia. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.

Clark, A. (1996). *Being There: Putting Brain, Body, and World Together Again*. MIT Press, Cambridge, MA, USA.

Cybenko, G. (1989). Approximation by superposition of a sigmoidal function. *Mathematics of Control, Signal and System*, 2:303–314.

Dawkins, R. and Krebs, J. (1978). Animal signals: information or manipulation. In Krebs, J. and Davies, N., editors, *Behavioural Ecology: an Evolutionary Approach*, pages 282–309. Blackwell Scientific Publications, Oxford.

Di Paolo, E. A. (1999). *On the Evolutionary and Behavioral Dynamics of Social Coordination: Models and Theoretical Aspects*. PhD thesis, School of Cognitive and Computing Sciences, University of Sussex.

Di Paolo, E. A. (2000). Behavioral coordination, structural congruence and entrainment in a simulation of acoustically coupled agents. *Adaptive Behavior*, 8(1):25–46.

Dorigo, M. and Schnepf, U. (1993). Genetics-based machine learning and behaviour based robotics: A new synthesis. *IEEE Transactions on Systems, Man and Cybernetics*, 23(1):141–154.

E-Puck robot development group (2007). `http://www.e-puck.org/`.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14(2):179–211.

Floreano, D., Mitri, S., Magnenat, S., and Keller, L. (2007). Evolutionary conditions for the emergence of communication in robots. *Current Biology*, 17.

Floreano, D. and Mondada, F. (1994). Automatic Creation of an Autonomous Agent: Genetic Evolution of a Neural Network Driven Robot. In Cliff, D., Husbands, P., Meyer, J.-A., and Wilson, S., editors, *3rd International Conference on Simulation of Adaptive Behavior (SAB'1994)*. MA: MIT Press. D. Cliff, P. Husbands, J.-A. Meyer, and S. Wilson (eds.).

Fodor, J. A. and Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: a critical analysis. In Pinker, S. and Mehler, J., editors, *Connections and Symbols*. MIT Press, Cambridge, MA, USA.

Harvey, I., Paolo, E. D., Wood, R., Quinn, M., and Tuci, E. (2005). Evolutionary robotics: A new scientific tool for studying cognition. *Artificial Life*, 11(1):79–98.

Haugeland, J. (1985). *Artificial intelligence: the very idea*. Massachusetts Institute of Technology, Cambridge, MA, USA.

Hauser, M. D. (1997). *The Evolution of Communication*. MIT Press/BradfordBooks.

Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor.

Holland, J. H. (1986). Escaping brittleness: The possibilities of general-purpose learning algorithms applied to parallel rule-based systems. In Michalski, R. S., Carbonell, J. G., and Mitchell, T. M., editors, *Machine Learning: An Artificial Intelligence Approach: Volume II*, pages 593–623. Kaufmann, Los Altos, CA.

Louis, S. and Rawlins, G. (1992). Predicting convergence time for genetic algorithms. Technical report, Department of Computer Science, Indiana University, Bloomington, IN, 47405.

Marocco, D. and Nolfi, S. (2006). Emergence of communication in teams of embodied and situated agents. In *Proceedings of the 6th International Conference on the Evolution of Language*, pages 198–205.

Marocco, D. and Nolfi, S. (2007). Communication in natural and artificial organisms: Experiments in evolutionary robotics. In Lyon, C., Nehaniv, C. L., and Cangelosi, A., editors, *Emergence of Communication and Language*, pages 789–803. Springer Verlag, London. In Press.

McCarthy, J. and Hayes, P. J. (1969). Some philosophical problems from the standpoint of artificial intelligence. In Meltzer, B. and Michie, D., editors, *Machine Intelligence 4*, pages 463–502. Edinburgh University Press.

Mondada, F., Franzi, E., and Guignard, A. (1999). The Development of Khepera. In *First International Khepera Workshop*, HNI-Verlagsschriftenreihe, Heinz Nixdorf Institut, pages 7–14.

Mondada, F., Franzi, E., and Ienne, P. (1994). Mobile Robot Miniaturization: A Tool for Investigation in Control Algorithms. In *Third International Symposium on Experimental Robotics*, pages 501–513. Springer Verlag.

Nilsson, N. J. (1984). Shakey the robot. Technical Report 323, AI Center, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025.

Nolfi, S. (2000). Evorobot 1.1 user manual. Technical report, Rome: Institute of Psychology, CNR.

Nolfi, S. (2002). Evolving robots able to self-localize in the environment: The importance of viewing cognition as the result of processes occurring at different time scales. *Connection Science*, 14(3):231–244.

Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: The Biology, Intelligence and Technology.* MIT Press, Cambridge, MA, USA.

Nolfi, S., Floreano, D., Halloy, J., Holmquist, L., Loreto, V., Szathmry, E., Sol, R. V., and Steels, L. (2006). Blue paper: A research roadmap for developing artificial embodied and communicating agents. `http://ecagents.org/imgs/blu_paper.pdf`.

Nolfi, S., Miglino, O., and Parisi, D. (1994). Phenotypic plasticity in evolving neural networks. In Gaussier, P. and Nicoud, J., editors, *From Perception to Action. Proceedings of the PerAc'94 Conference.*, pages 146–157. IEEE Computer Society Press, Los Alamitos, CA.

Parisi, D. and Mirolli, M. (2007). How producer biases can favour the evolution of communication: an analysis of evolutionary dynamics. `http:ecagents.istc.cnr.it/dllink.php?id=296&type=Document`.

Quinn, M. (2001). Evolving communication without dedicated communication channels. In Kelemen, J. and Sosk, P., editors, *ECAL01*, Lectures Notes in Computer Science, pages 357–366, Prague. Springer.

Sims, K. (1994). Evolving 3d morphology and behavior by competition. *Artificial Life*, 1(4):353–372.

Stanley, K. O. and Miikkulainen, R. (2002a). Continual coevolution through complexification. In *GECCO '02: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 113–120, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Stanley, K. O. and Miikkulainen, R. (2002b). Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127.

Steels, L. (2005). The emergence and evolution of linguistic structure: from lexical to grammatical communication systems. *Connection Science*, 17(3-4):213–230.

Tinbergen, N. (1952). Derived activities: Their causation, biological significance, origin and emancipation during evolution. *Quarterly Review of Biology*, 27:1–32.

Ujhelyi, M. (1996). Is there any intermediate stage between animal communication and language? *Journal of Theoretical Biology*, 180(1):71–76.

Wagner, K., Reggia, J. A., Uriagereka, J., and Wilkinson, G. S. (2003). Progress in the simulation of emergent communication and language. *Adaptive Behavior*, 11(1):37–69.

Watson, R. A., Ficici, S. G., and Pollack, J. B. (2002). Embodied evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1):1–18.

Weaver, W. and Shannon, C. E. (1949). *The Mathematical Theory of Communication.* University of Illinois Press, Urbana, Illinois. republished in paperback 1963.

Werbos, P. J. (1974). *Beyond regression: New Tools for Prediction and Analysis in the Behavioral Sciences.* PhD thesis, Committee on Applied Mathematics, Harvard University.