

DRAMA, a connectionist model for robot learning:  
Experiments on grounding communication through  
imitation in autonomous robots

Aude Billard



Ph.D.  
University of Edinburgh  
1998



GRAMA, a computer-based model for robot learning:  
Experiments on knowledge communication through  
imitation in autonomous robots

Andrés Ballester



18 12  
University of Valencia  
2008

## Abstract

The present dissertation addresses problems related to robot learning from demonstration. It presents the building of a connectionist architecture, which provides the robot with the necessary cognitive and behavioural mechanisms for learning a synthetic language taught by an external teacher agent. This thesis considers three main issues: 1) learning of spatio-temporal invariance in a dynamic noisy environment, 2) symbol grounding of a robot's actions and perceptions, 3) development of a common symbolic representation of the world by heterogeneous agents.

We build our approach on the assumption that grounding of symbolic communication creates constraints not only on the cognitive capabilities of the agent but also and especially on its behavioural capacities. Behavioural skills, such as imitation, which allow the agent to co-ordinate its action to that of the teacher agent, are required aside to general cognitive abilities of associativity, in order to constrain the agent's attention to making relevant perceptions, onto which it grounds the teacher agent's symbolic expression. In addition, the agent should be provided with the cognitive capacity for extracting spatial and temporal invariance in the continuous flow of its perceptions. Based on this requirement, we develop a connectionist architecture for learning time series. The model is a Dynamical Recurrent Associative Memory Architecture, called DRAMA. It is a fully connected recurrent neural network using Hebbian update rules. Learning is dynamic and unsupervised. The performance of the architecture is analysed theoretically, through numerical simulations and through physical and simulated robotic experiments. Training of the network is computationally fast and inexpensive, which allows its implementation for real time computation and on-line learning in a inexpensive hardware system. Robotic experiments are carried out with different learning tasks involving recognition of spatial and temporal invariance, namely landmark recognition and prediction of perception-action sequence in maze travelling.

The architecture is applied to experiments on robot learning by imitation. A learner robot is taught by a teacher agent, a human instructor and another robot, a vocabulary to describe its perceptions and actions. The experiments are based on an imitative strategy, whereby the learner robot reproduces the teacher's actions. While imitating the teacher's movements, the learner robot makes similar proprio and exteroceptions to those of the teacher. The learner robot grounds the teacher's words onto the set of common perceptions they share. We carry out experiments in simulated and physical environments, using different robotic set-ups, increasing gradually the complexity of the task. In a first set of experiments, we study transmission of a vocabulary to designate actions and perception of a robot. Further, we carry out simulation studies, in which we investigate transmission and use of the vocabulary among a group of robotic agents. In a third set of experiments, we investigate learning sequences of the robot's perceptions, while wandering in a physically constrained environment. Finally, we present the implementation of DRAMA in Robota, a doll-like robot, which can imitate the arms and head movements of a human instructor. Through this imitative game, Robota is taught to perform and label dance patterns. Further, Robota is taught a basic language, including a lexicon and syntactical rules for the combination of words of the lexicon, to describe its actions and perception of touch onto its body.



## Acknowledgements

First and before all, I wish to deeply thank Gillian Hayes and Kerstin Dautenhahn for their guidance and encouragements all along this project. I gratefully acknowledge their warm encouragements to submit my work to peer review through conference and journal publications. They proved to be perfect supervisors, giving me a large amount of freedom while always being present for critical and useful comments.

Many thanks to David Willshaw and Rolf Pfeifer, examiners of this thesis, for their very judicious comments. Many thanks also to John Demiris, Maja Mataric, Bruce Graham and anonymous reviewers for useful comments on drafts reports of this work.

Many thanks to Luc Steels for gracefully inviting me at the laboratory of Artificial Intelligence at the Vrije Universiteit Brussels. I benefited enormously from the month I spent there, both by getting the chance of using a different hardware and software set-up, and also from interesting discussions I had with him and other members of the lab.

An enormous thanks to all the technicians of the DAI, whose precious help made this work possible. I am particularly grateful to Alexander Colquhoun for his excellent work on developing several electrical circuits of the robot. I warmly thank Tim Colles, David Wyse, Hugh Cameron, Douglas Howie, Robert MacGregor, Neil Wood, Karen Konstroffer, Olga Frank and Janice Gailani for their constant and very kind help for all sorts of computing, hardware and documentary bits.

Warmest thanks to Auke Jan Ijspeert for these wonderful three years spent together in Edinburgh and for sharing all the joys, difficulties, excitations, ... of our doctoral works. He made the long hours spent in the lab very enjoyable.

I am forever grateful to my mother for her constant support, encouragement and faith in my capabilities, which gave me the strength for carrying out my studies. I dedicate my work to her and to all women who were not permitted to pursue scientific studies.

I gratefully acknowledge financial support from the Swiss National Research Fund, the Leon et Helene Mouttet foundation and the Sunburst fund.

## Acknowledgments

First and foremost, I wish to thank my dear friends and family members for their support and encouragement throughout this journey. I am particularly grateful to my parents, who have always been my biggest supporters and have encouraged me to pursue my dreams. I also want to thank my friends, who have been there for me through thick and thin, and have helped me stay motivated and focused on my goals.

I also want to thank my colleagues and mentors at work, who have provided me with valuable feedback and guidance. I am especially grateful to my supervisor, who has been a great role model and has helped me grow both personally and professionally.

Finally, I want to thank my partner, who has been my rock and my biggest supporter. I could not have done this without your love, support, and encouragement. You have been my constant source of strength and motivation, and I am forever grateful to you for everything you have done for me.

I also want to thank the many people who have helped me along the way, including my teachers, friends, and family members. I am particularly grateful to my friends, who have been there for me through thick and thin, and have helped me stay motivated and focused on my goals. I also want to thank my colleagues and mentors at work, who have provided me with valuable feedback and guidance.

I am also grateful to my family members, who have always been my biggest supporters and have encouraged me to pursue my dreams. I am particularly grateful to my parents, who have always been my biggest supporters and have encouraged me to pursue my dreams.

I am also grateful to my friends, who have been there for me through thick and thin, and have helped me stay motivated and focused on my goals. I am particularly grateful to my friends, who have been there for me through thick and thin, and have helped me stay motivated and focused on my goals.

# Declaration

I hereby declare that I composed this thesis entirely myself and that it describes my own research.

## Contents

Abstract	iii
Acknowledgements	v
Declaration	vi
List of Figures	viii
1 Introduction	1
1.1 Formal theory in robotics	1
1.1.1 Classical robotics	1
1.1.2 Adaptive learning robots	2
1.1.3 Neural networks	3
1.1.4 Connectionist models	4
1.2 Theory and control of machines	10
1.2.1 Introduction	10
1.2.2 Examples	16
2 Background work	17
2.1 Learning robotics	17
2.1.1 Reinforcement learning	18
2.1.2 Artificial Neural Networks	20
2.1.3 Supervised Learning	24
2.2 Robot Dynamics	25
2.2.1 Dynamics of rigid bodies	25

Aude Billard  
Edinburgh  
March 16, 1999

Introduction

I have written this book for those who are interested in the history of the world and who want to know more about the people who have lived on this planet. It is a story of the human race, from the first man to the present day.

John G. Bennett  
1963

# Contents

<b>Abstract</b>	<b>iii</b>
<b>Acknowledgements</b>	<b>v</b>
<b>Declaration</b>	<b>vii</b>
<b>List of Figures</b>	<b>xviii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Recent trends in robotics . . . . .	2
1.1.1 Situated embodied robots . . . . .	4
1.1.2 Adaptive learning robots . . . . .	7
1.1.3 Social robotics . . . . .	9
1.1.4 Communicative robots . . . . .	11
1.2 Scope and content of this thesis . . . . .	12
1.2.1 Contribution . . . . .	14
1.3 Thesis Plan . . . . .	15
<b>2 Background work</b>	<b>17</b>
2.1 Learning robotics . . . . .	17
2.1.1 Reinforcement learning . . . . .	18
2.1.2 Artificial Neural Networks . . . . .	20
2.1.3 Evolutionary Algorithms . . . . .	21
2.2 Social Robotics . . . . .	23
2.2.1 Cooperative robotics . . . . .	24

2.2.2	Socially intelligent robotics . . . . .	26
2.3	Linguistics . . . . .	28
2.3.1	Evolutionary linguistics . . . . .	29
2.3.2	Natural language processing . . . . .	32
2.4	Summary . . . . .	33
<b>3</b>	<b>Work hypotheses</b>	<b>35</b>
3.1	Terminology . . . . .	35
3.2	The symbol grounding problem . . . . .	41
3.3	Social interaction and the development of language . . . . .	43
3.4	Our approach compared to others . . . . .	45
3.4.1	Situatedness and embodiment . . . . .	46
3.4.2	Social interaction . . . . .	47
3.5	Synthesis . . . . .	48
<b>4</b>	<b>The DRAMA Architecture</b>	<b>51</b>
4.1	Associative Learning and Time Perception . . . . .	51
4.2	Review of models for spatio-temporal associations . . . . .	53
4.2.1	Terminology and classification of neural networks . . . . .	53
4.2.2	Associative memory . . . . .	56
4.2.3	Recurrent Neural Networks . . . . .	57
4.3	Structure of the DRAMA architecture . . . . .	59
4.3.1	The Willshaw net . . . . .	59
4.3.2	A control architecture for autonomous robot . . . . .	62
4.3.3	Data encoding . . . . .	64
4.3.4	Event recognition preprocessing module . . . . .	65
4.3.5	Associative module (DRAMA) . . . . .	66
4.4	Summary . . . . .	76
4.4.1	Discussion . . . . .	76
<b>5</b>	<b>Evaluation of DRAMA performance</b>	<b>79</b>
5.1	Model functionality . . . . .	79

5.1.1	The role of the spatial filter . . . . .	80
5.1.2	Considerations on factors affecting the ability of the network to store binary patterns without time . . . . .	85
5.1.3	Storage of competitive rules . . . . .	88
5.1.4	The importance of choosing the threshold correctly . . . . .	89
5.1.5	The role of the temporal filter . . . . .	91
5.1.6	Space and time efficiency . . . . .	95
5.2	Sequence learning . . . . .	96
5.2.1	Learning algorithm for the short-term memory parameter . . . . .	97
5.2.2	The dynamics of the network . . . . .	100
5.3	Summary . . . . .	103
5.3.1	Discussion . . . . .	104
<b>6</b>	<b>Overview of the robotic experiments</b>	<b>109</b>
6.1	Hardware and Software description . . . . .	110
6.1.1	The robots . . . . .	110
6.1.2	The simulator . . . . .	114
6.2	DRAMA processing of the robot's sensor and actuator information . . . . .	115
6.3	Summary of the experiments . . . . .	119
<b>7</b>	<b>Grounding of a robot's perceptions and actions</b>	<b>125</b>
7.1	Transmitting a vocabulary from a teacher robot to a learner robot . . . . .	126
7.1.1	Imitative strategy . . . . .	126
7.1.2	Symbol grounding using a simultaneous association process . . . . .	128
7.1.3	Symbol grounding using sequence processing of sensory variation . . . . .	133
7.2	Transmitting a vocabulary from a human instructor to a learner robot . . . . .	140
7.2.1	Teaching a vocabulary for the orientation . . . . .	141
7.2.2	Teaching a vocabulary for describing objects . . . . .	142
7.3	Discussion and synthesis . . . . .	151
<b>8</b>	<b>Grounding of perceptions of a group of robots</b>	<b>155</b>
8.1	Learning the vocabulary . . . . .	155

8.1.1	Experimental set-up . . . . .	156
8.1.2	Results and discussion . . . . .	158
8.2	Measure of the influence of the imitative strategy on the learning performance . . . . .	161
8.2.1	Experimental set-up and procedure . . . . .	162
8.2.2	Results and discussion . . . . .	162
8.3	Synthesis . . . . .	164
<b>9</b>	<b>Using the robot's communicative ability</b>	<b>167</b>
9.1	Two-robot collaboration . . . . .	168
9.1.1	Experimental procedure . . . . .	169
9.1.2	The behavioural architecture . . . . .	169
9.1.3	Experimental set-up . . . . .	174
9.1.4	Results . . . . .	175
9.2	Collaboration among a group of robots . . . . .	176
9.2.1	Retrieval mechanism . . . . .	177
9.2.2	Results . . . . .	178
9.3	Human-robot interaction . . . . .	180
9.3.1	Retrieval mechanism . . . . .	181
9.3.2	Results . . . . .	182
9.4	Synthesis . . . . .	183
<b>10</b>	<b>Learning sequences of perceptions</b>	<b>185</b>
10.1	Experimental set-up . . . . .	185
10.2	The experimental procedure . . . . .	187
10.3	Results . . . . .	188
10.4	Synthesis . . . . .	194
<b>11</b>	<b>Experiments with Robota, the doll robot</b>	<b>195</b>
11.1	The doll robot . . . . .	196
11.2	The robot's controller . . . . .	199
11.3	Experiments . . . . .	201

11.3.1	Learning dance patterns . . . . .	201
11.3.2	Experiments on word combinations . . . . .	204
11.4	Tests with children . . . . .	209
11.5	Discussion . . . . .	213
11.6	Synthesis . . . . .	215
<b>12</b>	<b>Discussion</b>	<b>217</b>
12.1	DRAMA's theoretical performance . . . . .	217
12.1.1	The model as an associative memory . . . . .	218
12.1.2	The model as a recurrent neural network . . . . .	220
12.1.3	Limitations of the model . . . . .	221
12.2	DRAMA's experimental performance . . . . .	223
12.2.1	Grounding of a robot's perceptions and actions . . . . .	223
12.2.2	Grounding of perceptions of a group of robots . . . . .	229
12.2.3	Using the robot's communicative ability . . . . .	231
12.2.4	Grounding of a robot's sequences of actions and perceptions . . . . .	233
12.2.5	Imitative behaviour . . . . .	235
12.3	Overall synthesis and outlook . . . . .	236
12.3.1	Future work and perspectives . . . . .	236
12.3.2	Cognitivist versus behaviour-based approach . . . . .	240
<b>13</b>	<b>Conclusion</b>	<b>245</b>
13.1	Contributions . . . . .	246
	<b>Bibliography</b>	<b>249</b>
<b>A</b>	<b>Results of chapter 8: Grounding in a group of robots</b>	<b>273</b>
<b>B</b>	<b>Results of chapter 10: Learning sequences of perceptions</b>	<b>277</b>
<b>C</b>	<b>Results of chapter 11: Experiments with Robota, the doll robot</b>	<b>279</b>
C.1	Results of section 11.3.1: learning dance pattern . . . . .	279
C.2	Results of section 11.3.2: experiment on word combinations . . . . .	280



# List of Figures

4.1	Schema of the Willshaw network . . . . .	60
4.2	Schema of the robot's control system, using the DRAMA architecture . . . . .	63
4.3	Event detector module . . . . .	65
4.4	The DRAMA architecture. . . . .	67
4.5	<b>Left:</b> Bidirectional connectivity of two network units. <b>Right:</b> One unit connectivity. . . . .	67
4.6	Threshold-based retrieval mechanism. . . . .	70
4.7	Propagation of the unit activity along the network connections. . . . .	71
4.8	Transcription of sensor input into event detector output and DRAMA units output. . . . .	72
4.9	Example of association between unit 0 and unit 1. . . . .	75
5.1	Examples of training pattern configurations which lead to unsuccessful learning when retrieval is based only on a threshold rule on the confidence factor. Bottom example becomes successful 1) when retrieval is based on both a threshold rule on the confidence factor and a threshold rule on the time parameter, or 2) when retrieval is based only on a threshold rule on the confidence factor and on a threshold rule on the number of winning units (see text for explanations). . . . .	81
5.2	Values of the connection parameters (cf) for determining basic behaviours of obstacle avoidance, light phototaxis and wandering. The predominance of one behaviour over the other is determined by the ratio of their respective connection strength. . . . .	89
5.3	Overlap between desired and actual output during recall of a 20 units network (10 In, 10 Out) for different values of noise (spurious unit activity) and three different threshold strategies: (1) $T = 2$ , no threshold on time; (2) $T = T_a$ , no threshold on time; (3) $T = T_a$ , $e = e(tp)$ . . . . .	90

5.4	<b>Top:</b> Variation of short-term memory, error and time threshold during the training of the sequence <i>ABCDEFCDG</i> . <b>Bottom:</b> Superposed plots of the retrieved (straight line) and training (dotted line) pattern activation along the training; the figure shows snapshots of (left) the three first cycles of the training and (right) the 37th training cycle, that is after convergence. . . . .	99
5.5	Possible orbits of the network's dynamic. . . . .	102
6.1	<b>Top:</b> schema of the LEGO robots' structure. <b>Bottom:</b> picture of the learner robot inside the Dodgem cage. . . . .	112
6.2	DRAMA connections state in the experiments. . . . .	115
6.3	Activity of the units corresponding to the left and right motor, the compass, light and inclination sensors, during 1000 processing cycles. . . . .	117
6.4	Experiments of chapter 7: transmission of a vocabulary from a teacher robot (left) or human instructor (right) to a learner robot. . . . .	121
6.5	Experiment of chapter 8: Transmission of a vocabulary in a group of robot. . . . .	122
6.6	<b>Top:</b> Experiment of chapter 10: teacher and learner robots travel in a series of three corridors. <b>Bottom:</b> Experiment of chapter 11: a human instructor teaches Robot <sub>a</sub> , the doll robot, a vocabulary to label sequences of arm and head movements. . . . .	124
7.1	<b>Teaching Scenario:</b> A teacher robot (1) teaches a learner robot (2) a vocabulary to describe (left) its actions and orientation and (right) its inclination and light intensity measurements. While following the teacher, the learner robot grounds the teacher's 'words' onto its own observations. . . . .	127
7.2	<b>Left:</b> Picture of the two LEGO robots. <b>Right:</b> Picture of the two Fischer-Technik robots, beginning to climb the hill . . . . .	129
7.3	Results of experiment on hill climbing. . . . .	130
7.4	Simulated and physical environments in experiments on teaching the robot a vocabulary for its actions and orientation. . . . .	134
7.5	Confidence factor values during a run in simulation studies (left) and physical experiments (right), for the second phase of teaching. . . . .	139
7.6	<b>Left:</b> Mean value of confidence factor parameters for each compass-radio signal pair. <b>Right:</b> Learning curve along each cycling around the cage for each of the four associations signal-compass orientation. . . . .	141
7.7	<b>Left:</b> The LEGO robot standing in front of an aluminium covered area. Behind it are the box and cardboard wall. <b>Right:</b> The simulated teacher robot (No 2) followed by the student robot (1), moving between hill 2 and 3. . . . .	144
7.8	Comparison of three simulator experiments using different object dispersion. . . . .	146

7.9	<b>Top:</b> Confidence factor values for connections between radio units (1 unit = 1 'word': 5 words for the three hills and the two boxes) and colours, shape and inclination sensors units (hills' and boxes' features). <b>Bottom:</b> Success of signal-object correlation given 4 different values of threshold $T$ . Y/N stands for Yes/No (correct/incorrect correlation); R: radio sensor. 'R→' and '→R' columns show results of learning for each direction of association (from radio sensor to object's description and vice-versa). . . . .	151
8.1	Graphical representation of the simulated environment for the experiments on learning of a group of robots. . . . .	157
8.2	Learning curve (mean value for all agents) for four different thresholds (10,20,30 and 50) on the radio unit activation. . . . .	159
8.3	Learning curves for follower and non-follower learner agents in three different learning tasks: learning a vocabulary for objects, for scaling values of polar coordinates and for orientation relative to a compass. . . . .	166
9.1	The behavioural architecture of the mother-child scenario. . . . .	170
9.2	Schematic view of the 'Mother-Child' experiment. . . . .	174
9.3	Speed of learning colour patches locations in three cases (mean value over all nine robots and all runs; error bar are standard deviations): as a result of each robot's individual exploration (-), by listening to another robot's speech which talks as soon as it finds a new location (-.-), through dialogue conversation when meeting another robot (...). . . . .	179
9.4	Schema of the robot's path inside the laboratory. . . . .	181
10.1	<b>From left to right:</b> a graphical representation of the simulated environment and a picture of the physical one. . . . .	186
10.2	Variation of the time parameter values for each correct connection signal-object. Results of simulations and physical experiments (left and right columns respectively). . . . .	190
10.3	Retrieval of the sequence of sensor stimuli when presenting the radio signal for 'South' (signal 1) after learning of the time series. . . . .	192
11.1	<i>Top:</i> The LEGO structure of the robot doll with the mini keyboard; <i>Bottom:</i> The doll dressed up, with, on the right side, the feeding bottle, the glasses and the infra-red hand sensors. . . . .	197
11.2	Schematic representation of the sensor positions on Robota's body. Correspondences between movements of the demonstrator's arm and head and Robota's recognition of these movements are indicated. . . . .	198
11.3	Examples of a teaching of eight dance patterns . . . . .	203

11.4	Examples of teaching of two word combinations for lifting up the robot's right arm and pressing the switch on the robot's left arm. . . . .	206
11.5	Kyo Takiguchi 'feeding' Robota . . . . .	211
C.1	Left: G. Hayes; time≈10-15 min. Right: A-J Ijspeert; time≈10 min. . .	279
C.2	Name: A-J Ijspeert; time≈10 and 15 min. resp. . . . .	279
C.3	Name: A. Billard; time≈15 and 10 min. resp . . . . .	280
C.4	Name: A. Billard; time≈10 min. for all . . . . .	280

# Chapter 1

## Introduction

Research in *Artificial Intelligence* (AI) aims at building artificial agents which show similar cognitive and behavioural capabilities to those of natural ‘intelligent’ agents. This thesis contributes to research in AI-based robotics, which studies the building of autonomous mobile agents, commonly called robots. The present work attempts to integrate three directions of research taken recently by this field, namely the building of adaptive learning agents, the study of social collaborative agents, the development of agents capable of understanding human language. The aim of this thesis is three-fold:

- To design a general connectionist architecture for learning spatio-temporal regularities and time series in an autonomous robot.
- To investigate a method, based on imitation, for the transmission of a language and of motor skills, action sequences, between heterogeneous agents.
- To build robotic agents capable of learning a synthetic language and action sequences taught by another agent, either a human or another robot.

We follow a synthetic approach and develop a connectionist model, the DRAMA architecture, which provides the robot with the necessary cognitive and behavioural skills for learning a basic synthetic language. We distinguish between behavioural and cognitive capabilities. The agent’s behaviour relates to the dynamic interaction of the agent with its environment, as observed by an external agent, while the agent’s cognition is an internal process, which results in the agent’s actions. The term *language*, used above, refers to any communication system which shows similar characteristics

to that of *natural language*, that is, the use of symbols to convey meaning and the use of syntactic rules, which define the semantics behind the order of combinations of symbols. In section 3.1, we define formally the terminology common to Linguistics and Psychology, which we use in this thesis.

We implement and verify the model in different robotic experiments, in which a robot is taught a synthetic proto-language. The language consists of a lexicon and of combinations of words of the lexicon, which form English proto-sentences<sup>1</sup>. The teaching scenario is based on an imitative strategy, whereby the robot replicates the teacher agent's movements. Through this imitative teaching approach, teacher and learner agents are lead to share similar proprio- and exteroceptions<sup>2</sup>. The learner agent grounds its understanding of the teacher's words, which describe the teacher's current observations, upon its own perceptions, which are similar to those of the teacher. Learning of the robot is unsupervised and results from the self-organisation of the robot's connectionist architecture.

The first section of this chapter gives a brief introduction to the research areas behind this work. The second section points out the main issues tackled by this work and briefly summarises our approach to solving these. The third section of this chapter gives a short summary of the contents of each chapter of the thesis.

## 1.1 Recent trends in robotics

Recent trends in AI robotics follow an *anti-objectivist*<sup>3</sup> approach to the design of the robot's controller, which emphasizes the robot's unique<sup>4</sup> and subjective perception of the world, as an effect of its individual physical structure. This position contrasts with the *objectivist*<sup>5</sup> approach, followed by so called *Good old fashioned AI* (GOFAI)

---

<sup>1</sup> Proto means that the sentences do not fulfil all the requirements of the English grammar, see section 11.3.2.

<sup>2</sup> The terms proprio- and exteroception relate to internal and external perceptions of the agent respectively.

<sup>3</sup> My labelling.

<sup>4</sup> Uniqueness refers here to the idea that each physical agent differs from another agent, no matter how similar the two agents are, and that, thus, its interaction with and perception of the world is different from that of any other agent.

<sup>5</sup> My labelling.

[Brooks & Stein 94] or *classical* AI [Pfeifer 96], which views the robot as a computational machine, independent from its physical support, that relies on an abstract and objective<sup>6</sup> model of the world. In the objectivist approach, the robot's behaviour is entirely determined by its perceptions and how these are interpreted relative to its world model. The robot's computation consists of the manipulation of abstract entities, *symbols*, which represent intrinsic properties of the world.

This approach's assumption of the existence of a unique and global representation of the world received three main criticisms. First, it was pointed out that there is, in fact, more than one possible representation of a same phenomena, i.e the same phenomena can be described in different ways, and thus, some representations might be more appropriate than others for solving the particular task. Second, "the world is only partially knowable" [Verschure & Pfeifer 92], i.e. each agent can only build a partial and thus subjective representation of the world. Third, the world is dynamic and thus failure of making the correct representation, which remains consistent with changes in the agent's perception of the world, can lead to a significant degradation of the robot's behaviour. This is referred to as the *frame problem* [Hayes 84]. The classical approach has also been criticised for not considering two main issues. The first one is the *symbol grounding problem* [Harnad 90], which refers to the question of how the symbols or the representations of the world acquire their *meaning* for the computational agent (see section 3.2 for more explanations). The second one is how the agent's actions affect its environment and reflect back to its perceptions. This refers to the notions of *situatedness* and *embodiment* introduced by [Brooks 91b] (see section 1.1.1).

In reaction to the classical approach, behaviour-based systems were developed as a paradigm for the design of the robot's control system. This started with Brooks's subsumption architecture [Brooks 89]. The control of behaviour-based systems is distributed over several modules which directly connect sensors to actuators and the overall behaviour emerges from the interaction of the different modules. In contrast to knowledge-based architectures of classical AI, behaviour-based systems do not rely on a global model of the environment [Brooks 91a], but they rather use "the world as its own model" [Brooks 91b]. This proved to make the system more robust to variation in

---

<sup>6</sup> Objective means here that the model would reflect general and intrinsic properties of the world, which are true and observable for all agents, independently of the type of agent.

the environmental dynamics [Brooks & Stein 94, Brooks 97].

The anti-objectivist approach views the ‘perception-to-action’ process as a continuum. It stresses the bidirectionality of the process, as opposed to an objectivist approach which views it as an uni-directional process, from perception to action. [Pfeifer & Scheier 94] point out the importance and possible precedence of the action to perception link on the opposite direction and speak of a *sensory-motor coordination* [Pfeifer & Scheier 98, Pfeifer 95]. This position is illustrated by a number of robotic experiments (carried out by the authors ([Pfeifer & Scheier 98], [Pfeifer 95]) and by other people, see [Pfeifer 95]), in which the robot’s behaviour is entirely controlled through the combination of sensor-actuator correlations (most of these works used a connectionist model for the robot’s controller). Such an approach considers then not only the influence of the environment on the agent, but also the influence the agent has on its environment. The robot’s behaviour is no longer intrinsically defined in the controller but rather *emerges* from the coupling between the robot’s internal dynamic and that of the environment [Dautenhahn 97b, Verschure & Pfeifer 92].

Such an account of the robot’s behaviour has similarities with the *enactivist* position, first introduced by Maturana [Maturana & Varela 80], which describes a living system in terms of the structural coupling of its cellular and molecular components (*autopoiesis*) and of its interaction with its environmental medium: “the behaviour of a living system is not something that the living system does, not something that the medium specifies of its own, the behaviour arises and takes place in the relation living system/medium” [Maturana *et al.* 95]. In an anti-objectivist approach, there is no longer a single objective world description, based on universal conceptual representations. Rather, the world is the result of a continuum of interactions between different agents and each agent can only build a subjective representation of the world, based on its own history of interactions. The implication that this philosophical position has on the building of the robot’s controller is described in the following.

### 1.1.1 Situated embodied robots

Robots have a dynamical and interactive relationship with their environment and with other agents in this environment. To encapsulate this idea, R. Brooks introduced the

notions of *embodied and situated* robots [Brooks 91b]. Embodiment relates to the fact that robots have a body, that is they occupy a physical space and consequently affect the environmental dynamics, and that they have sensors, which provide them with a perception of the state of their environment and, to some extent, of other agents in this environment. "The robots have bodies and experience the world directly - their actions are part of a dynamics with the world and have immediate feedback on their own sensations" ([Brooks 91b], p.571, original emphasis). Situatedness accounts for the fact that the robots' behaviour is spatially and temporally constrained by the dynamic interaction the robots have with their environment. Robots are provided with actuators, which allow them to act onto their environment. Their actions are directed by internal processing of their sensor perceptions, which are acted upon by the environment. "The robots are situated in the world - they do not deal with abstract description, but with the here and now of the world directly influencing the behaviour of the system" ([Brooks 91b], p.571, original emphasis). In this perspective, the robot does not rely on an abstract predefined model of the world, but rather on a subjective model of the world, whose conceptual representation is grounded onto the robot's physical perceptions of the world (the embodiment notion, emphasized in *the robot grounds regress* [Brooks 91b]) and represents the particular mode of interaction the robot has with its environment (the situatedness notion, summarised in *the world is its own best model* [Brooks 91b]).

Note that the analogy between the enactivist position and the notion of embodiment of artificial or natural agents, which we mentioned in the beginning of this chapter, has been discussed first by Varela [Varela *et al.* 93]. In this text, Varela stresses the physical aspects of the term embodiment relative to the agent's sensor-motor coordination: "By using the term *embodied*, we mean to highlight two points: first, that cognition depends upon the kinds of experience that come from having a body with various sensorimotor capacities, and second, that these individual sensorimotor capacities are themselves embedded in a more encompassing biological, psychological and cultural context" [Varela *et al.* 93]. While this point of view was primarily meant to natural agents, it could as well apply to artificial ones, such as robots.

The emphasis given to the physics of the robots leads to the idea that the beha-

viour of embodied situated agents is determined by both their ‘brain’ (controller) and their body, and by the interaction of these with the environmental dynamics. Therefore, building robotic agents requires careful design of the robot’s body and controller [Brooks & Stein 94], such as to take advantage of the dynamical interaction the robot makes with its environment [Hallam & Malcolm 94]. Following this line of thought, a number of works showed how complex behaviour, such as collecting, recognising and tracking objects ([Mataric 94a], [Parker 94b], [Pfeifer & Scheier 98], [Steels 94a], [Webb 95]), could be reduced to a set of simple sensor-motor coordination behaviours. In a knowledge-based approach, solving these tasks might have required building a map of the environment and some sort of planning system in the first case and complex sensor processing module in the second. While these works were still limited in the cognitive complexity achieved, people ([Brooks 97], [Brooks *et al.* 98b] [Dautenhahn 96], [Mataric 97d], [Pfeifer 98]) discuss how the approach could scale up to the design of high-level cognitive processes, as observed in high primates and humans (see sections 1.1.3 and 2.2.2). Current work in this direction studies the building of a humanoid robot, showing a range of behaviours similar to that of humans ([Breazeal & Scassellati 98, Brooks *et al.* 98a, Scassellati 98], [Kuniyoshi 94, Berthouze & Kuniyoshi 98]).

The work of this thesis is in line with the anti-objectivist approach. We address the symbol grounding problem in a robot. We consider both the cognitive and behavioural skills behind a robot’s learning of a language. In particular, we point out the importance of movement co-ordination between the communicative agents, for the agents to share a common perceptual context, onto which they can ground a common understanding of the language. Movement imitation is, in our experiments, the means by which we create a coupling between the two agents. It results from mutual phototaxis behaviours, whose implementation required a specific design of the robots’ bodies, i.e., positioning of light emitters and detectors, and controllers, i.e., recognition of specific light intensity and frequency. Situatedness is achieved by using a connectionist model for the robot’s controller, in which the agent’s behaviour results from multiple correlations across the robot’s sensors and actuators.

### 1.1.2 Adaptive learning robots

Closely related to the idea that the robot's behaviour is partly determined by the environmental dynamics, is the idea that the robot *learns* from its interaction with the environment. If the environmental dynamics is to change, then the robot's internal processing should also change such as to conserve the right coupling between the robot and its environment. The robot *adapts* its internal dynamics by updating its internal computational structure, such as to produce the 'correct' response for the particular task, that is, to perform the correct action given a particular perception. The process leading to the update of the robot's internal structure is called a *learning* process, in analogy with the biological term which relates to the human brain's internal transformation (synaptic growth, strengthening and pruning), simultaneous to human development of behavioural and cognitive skills. Through the update of its internal processing structure, the robot *learns* the novel regularities which characterise the dynamics of its environment, i.e. it learns to correctly predict its perceptions. It also learns the correlation between its actions and their result on the environment and updates its structure in order to produce the actions with the most desirable perceptual consequences.

The robot learning process contributes to the dynamical coupling the robot has with its environment. These characteristics make robot learning fundamentally different from machine<sup>7</sup> learning. Learning results in an internal transformation of the robot's controller, which further changes the robot's behaviour. Consequently the robot engages in a different mode of interaction with its environment, from which it can make new experiences leading to further learning. From learning regularities in its perceptual experience, the robot can construct a model of the world, which is a model of the dynamics of its perceptions in correlation to its actions, and a model of its body, which is of its internal dynamics and structure.

The construction of a model of its own body is fundamental to the idea of embodiment [Dautenhahn 95, Dautenhahn 97b, Berthouze & Itakura 98]. As noted by Dautenhahn [Dautenhahn 97b], it is not sufficient for a control program to be running on a hard-

---

<sup>7</sup> The term machine, here, relates to asensorial standing computational devices, by opposition to robot, which are mobile and have sensors.

ware platform to say that the robot is embodied. For a robot to be embodied, it has to have a concept of body, which it dynamically constructs upon its perceptual experience during its running time. Moreover, the agent's concept of its body is fundamental to its development of a model of the world, as it constructs its conceptual representation by reference to its own dynamics (subjective perception). Dautenhahn points out the similarity of this approach to Rosenfield's [Rosenfield 93] notion of *autobiographical memory*, which refers to the self-referential aspect of human memory and to the dynamical process of remembering in humans: "It is not that my memories exist as stored images in my brain, conscious or unconscious; the act of memory is one of my relating to myself, or to others, or to past experiences, or to previously perceived stimuli. This is the very essence of memory: its self-referential base, its self-consciousness, ever evolving and ever changing, intrinsically dynamic and subjective." ([Rosenfield 93], p.8, taken from [Dautenhahn 95], p.10). By analogy to the notion of autobiographical memory, Dautenhahn develops the concept of an *autobiographical agent* to encapsulate the subjective and dynamical aspects of learning and remembering of an artificial (robotic) agent: "An autobiographic agent ..[is].. an embodied agent which dynamically reconstructs its individual 'history' (autobiography) during its life-time" [Dautenhahn 96]. The agent's history is the agent's subjective account of its behaviour. Constructing the agent's history is to give *meaning* to the agent's perceptual information by interpreting it in the frame of a temporal relationship with the agent's actions. Therefore, an agent builds a meaningful (to the agent) conceptual representation of the world, by extracting temporal regularities in its perception-action loop.

Forming conceptual representations of the agent's perceptions underlies learning of a language, as the words of the language are a symbolic representation of concepts. The work of this thesis is based on the assumption that the processes behind the formation of concepts are the same as those allowing the symbol-concept mapping. Forming concepts is for the agent to extract invariances in the continuous flow of its perceptions, whereby the repetitive observation of a specific combination of sensor stimuli is interpreted as the representation of a given physical phenomena. We say that the so perceived combination of sensor stimuli represents the agent's concept of the phenomena. On the other hand, learning the symbol-concept mapping is for the agent

to associate its perception of another agent's physical signal (the symbol) with other perceptions, which are a representation of the concept meant by the signal. Therefore, the processes leading to the formation of concepts and of the symbol-concept mapping result both from an associative mechanism across multiple sensor-actuator modalities.

The above assumption is in agreement with a description of a robot's learning mechanism as a dynamical process, through which the robot builds a subjective model of the world, as discussed in previous sections of this chapter. In our approach, the robot builds a model of the world, by forming concepts grounded onto its proprio- and exteroception of varying sensor stimuli. In addition, through external teaching, it learns a symbolic representation of the concepts, which is similar to that of another homogeneous (similar robot) or heterogeneous agent (human).

### 1.1.3 Social robotics

While research in learning robotics is concerned with the robot interaction with its environment in general, research in *social robotics* focuses on the robot's interaction with other agents, e.g. humans or other robots. We here define briefly which research directions we include under the social robotics label. We define the expression in more detail in section 2.2.

Research in social robotics studies the design of social skills for robots. One research direction in this area is concerned with the study of organised groups of robots. This relates to research in *collective robotics*. Works in collective robotics are divided into two main groups, depending on whether they use explicit (i.e. defined by design) communication and collaboration or not [Mataric 94a]. The first group is generally referred to as research in *collaborative or cooperative robotics*. A second research direction taken by research in social robotics is, what we call, *socially intelligent robotics*. We define briefly this expression in the following paragraphs and refer to section 2.2.2 for a more complete definition. The work described in this thesis contributes to research in *socially intelligent robotics*. A short review of research in collective robotics is given in section 2.2.1. There, we point out aspects of these works which are relevant to this thesis, such as the study of the importance of explicit communication for the performance of a group of robots.

*Socially intelligent robotics* is concerned with the building of robots capable of engaging in complex social interactions, similar to that observed in high primates, with humans or with other robots. While research on collaborative robotics takes inspiration from ethological studies of insect societies, research in socially intelligent robotics takes inspiration from psychological and ethological studies of *social intelligence* and *social learning* in primates. *Social intelligence* is an ill-defined concept [Hemelrijk 95], which relates generally to cognitive and behavioural skills which animals (usually high primates) acquire and use through their social interactions with conspecific agents. *Social learning* or social learning theory aims to define the cognitive and behavioural competences by which an agent learns from its interactions with conspecific agents (the notions of social learning and social intelligence are further defined in section 3.1). Examples of social learning can be found in the animal kingdom, and especially in primate societies, whereby, for instance, one agent learns new skills by the observation and imitation of another agent's behaviour. Imitation capabilities have been studied, e.g., in macaques (see the well known example of the Japanese macaques of Koshima Island whose habit of washing potatoes developed from the observation and imitation of the individual finding of one macaque [Kawamura 63]), in parrots and mynah birds for reproducing songs [Nottebohm 76] and of course in human babies (e.g [Bandura 61, Meltzoff 90]) and in primates in general [Galef 88, Heyes 96, Whiten & Ham 92].

Psychologists, ethologists and roboticists find a common interest in the study of high-level social skills for robots, which can be used for gaining a better understanding of the mechanisms behind similar skills in animals ([Brooks *et al.* 98b], [Dautenhahn 97a], [Mataric 97d]) or for improving the robot's performance in interaction with humans or robots ([Berthouze & Itakura 98], [Dautenhahn 95], [Klingspor *et al.* 97], [Kuniyoshi 94], [Mataric 97a], [Pfeifer 98]). Examples of social skills which are currently investigated through human-robot or robot-robot experiments are, e.g., imitative behaviour ([Bakker & Kuniyoshi 96], [Cooke *et al.* 97], [Demiris & Mataric 98], [Demiris *et al.* 97], [Kuniyoshi 95], [Scassellati 98]), and, in particular, imitation used for learning in experiments on robot learning by demonstration ([Hayes & Demiris 94], [Gaussier *et al.* 98b], [Kuniyoshi & Inoue 94], [Mataric 94b], [Scassellati 98], [Schaal 97]). People study also the development and use of symbolic communication through robotic experiments on learning of a lexicon

([Steels & Vogt 97], [Yanco & Stein 93]) and on learning of locations in an office environment ([Tatsuno 96], [Torrance 92b]) and the use of body communication through facial expressions of emotion [Breazeal & Scassellati 98].

Of the above mentioned studies, most relevant to us are the works on robot learning by imitation and on robot learning of a language. We refer to the latter studies in the next section. In the works on robot learning by imitation, people developed complex algorithms, often based on neurological studies ([Hayes & Demiris 94], [Mataric 94b], [Scassellati 98]), to produce the sensor-motor mapping which results in the robot reproducing the movements demonstrated by a human instructor. In these studies, the robot's imitative behaviour is used to teach the robot complex motor skills, by the robot first observing and then repeating the movements of the arms ([Kuniyoshi & Inoue 94],[Mataric 94b], [Schaal 97]) and of the head ([Demiris *et al.* 97], [Scassellati 98]) of a human demonstrator. In the work of this thesis, we study how imitation skills can be used for teaching a robot a synthetic proto-language. Using the learner agent's imitative behaviour, namely following the other agent<sup>8</sup> or mirroring the other agent's movement, the teacher agent leads the robot through situations in which the robot can perceive the stimuli, to which the teacher refers symbolically by emitting specific signals (in the experiments, the signals are transmitted either by radio or by pressing keys on a keyboard). The robot can then attach the teacher's signals to its perception of other distinctive sensor stimuli, therefore giving meaning to the signals in terms of its own sensor perceptions.

#### 1.1.4 Communicative robots

Communication skills are fundamental to human society. They play an essential role for the transmission of knowledge across generations and are at the basis of social relationships between individuals [Lock 78]. Humans do not communicate only with each other. They also address verbally or through gesture their pets and other familiar animals, which they expect to show some degree of understanding of their talking, such as obedience to orders, responding to their name. Similarly, a minimum of

---

<sup>8</sup> Following behaviour is an implicit means of imitation, as when the first robot follows the second robot, it replicates the second robot's movements (moving, stopping, turning) in the plane.

communicative capabilities must be expected from artificial agents, such as robots, which are expected to share several aspects of human daily life (e.g. vacuum cleaner, food/money distributors, autonomous toys) [Dautenhahn 97a, Dautenhahn 95]. For instance, human-robot interaction might be facilitated if the robot might be able to learn and remember its different users' preferences. Such ideas are currently implemented in the Internet domain, by defining user preference sensitive search engines (e.g. 'softbots' or *situated* computer interfaces [Lueg & Pfeifer 97]), and are discussed in research on robot entertainment [Blumberg *et al.* 96, Fujita & Kitano 98, Maes 95].

Communicating with a robot has at least two advantages. The robot could explain its behaviour, which would make the task of its programmer easier in case of failure and would make the robot more trustable for its user. The robot could be commanded verbally, which makes it a more natural way (requiring no training) of interacting with the machine than by using a remote control button. It could also be taught by its user or programmer, e.g. teaching new vocabulary to be attached to new observations. While the robot could be provided with a built-in communication protocol, it would however show more adaptability towards the task and environment in which it is to be used, if it was able to learn the language. If the robot could learn the meaning of each word of the language and the language's specific syntax, the robot would not be restricted to be used only in specific domains or a specific country. Driven by these considerations, we develop, in this thesis, a cognitive model which allows a robot to learn a syntactical proto-language. The language is composed of a lexicon, whose items/words can be combined to form ordered sequences of words, which satisfy basic rules of the English grammar.

## 1.2 Scope and content of this thesis

This thesis addresses issues related to robot teaching by demonstration. We study how to teach a robot a language to describe concepts related to its perception of the world. The robot is taught either by a human or by another robot. The studied problem amounts to tackling the two following issues: 1) how to solve the *symbol grounding problem*, that is grounding arbitrary signals onto the agent's perceptions; 2) how to develop a similar symbolic representation of the world by two functionally different

agents.

Inspired by psychological and linguistic studies of the development of symbolic communication in children, we determine a number of key features for the cognitive and behavioural mechanisms of our agent. We require the agent to possess associative capabilities for spatio-temporal association across multiple sensor-actuator modalities and behavioural skills for coordinating its actions with that of a second agent.

We develop a learning architecture, DRAMA (Dynamical Recurrent Associative Memory Architecture), made of a connectionist model, which allows learning of spatial regularities and time series. We design a control system which is composed of event detector modules for segmenting the sensor-actuator information and the DRAMA associative module. Important characteristics of DRAMA are its robustness when faced with highly variable, noisy, data and its fast time of computation (for processing and retrieval of the data), which make the architecture very suitable for robotic implementation. We implement the architecture in several robotic experiments, to allow learning of spatio-temporal regularities in a robot's perceptions. In the experiments, DRAMA provides direct, on-line, control of the robot's learning (signal-perception association) and behaviour (retrieval of perception-action association).

We carry out experiments in simulated and physical environments, using different robotic set-ups, increasing gradually the complexity of the task. In a first set of experiments, we study transmission of a vocabulary to designate actions and perception of a robot. The robot is taught either by a second robot or by a human instructor. The experiments consist of the robot learning to attach arbitrary signals, symbols, onto different sets of its proprio- and exteroceptions. Through this teaching process, the robot builds a symbolic representation of concepts of the world which is similar to that of the teacher agent. Further, we carry out simulation studies, in which we investigate transmission and use of the vocabulary among a group of robotic agents. In a third set of experiments, we investigate learning sequences of the robot's perceptions while wandering in a physically constrained environment (series of corridors). We then present the implementation of DRAMA in *Robota*, a doll-like robot, which can imitate the arm and head movements of a human instructor. Through this imitative scenario, *Robota* is taught to perform and label dance patterns. Further, *Robota* is taught a synthetic

proto-language, composed of a lexicon and combinations of words of the lexicon, to describe its interaction with the teacher.

### 1.2.1 Contribution

The work of this thesis does not follow directly from any previous studies of robotics. Indeed, little work has yet been done in teaching a physical robot a synthetic form of communication. Closest works are those of Yanco and Stein [Yanco & Stein 93] and Vogt and Steels [Steels & Vogt 97], in which a robot learns a lexicon to describe sets of actions and perceptions respectively (these works are described in section 2.3.1). The present study differs from those works in several aspects:

- The learning and behavioural capacities of the robots result from a single cognitive architecture; it is a connectionist model which has general ability for extracting spatio-temporal regularities in a dynamic environment. It is used, in the experiments, for other learning tasks in addition to that of learning the language, which involve learning time series of perception-action. Thus, our model is more general than the compared studies, which used a learning mechanism, designed specifically for the particular language task.
- We carry out a wide range of experiments, using different robotic set-ups, which allows us to investigate the generality of the method relative to different experimental contexts. In particular, we show that it can be applied to both human-robot and robot-robot communication, without restriction on the robot's morphology (shape and sensors). In the experiments, we use vehicle-like robots and a doll-shaped robot. In contrast, the above mentioned studies implemented their model in a single robotic set-up, using vehicle based robots.
- The language which the robot is taught is not restricted only to a lexicon, where each word of the lexicon relates to a single specific perception, as in the two compared studies. Because of the general property of the robot's architecture for learning time series, the robot can be taught combinations of words of the lexicon to label combinations and sequences of perceptions. Therefore, our model suggests its application to learning a complete language, composed of a lexicon

and syntactical rules for combining the terms of the lexicon.

- Finally, based on psychological studies of the development of communication in children, we stress the importance of a behavioural co-ordination between the communicative agents, in addition to the above mentioned learning skills, for the good transmission of the language. This behavioural co-ordination is achieved, in our experiments, through movement imitation.

The role of behavioural skills in the development of a language has been neglected by most studies of computational linguistics (and in particular by the two above mentioned works). In this aspect, our work brings a novel contribution to current research in those areas, addressing the *symbol grounding problem* (see section 3.2) from a behaviourist by opposition to a pure cognitivist point of view, for which the notions of embodiment and situatedness are key issues for symbolic cognition.

Finally, note that the work of this thesis differs from research in computational linguistics, which investigates different hypotheses for the development of a language. We do not address the question of why a language might develop, but how it can develop, that is, how it can be learned and transmitted. In other words, we address the question of how different agents can develop a common understanding of a language, and, in particular, how one agent can achieve an interpretation of a set of symbols similar to that of a second heterogeneous (functionally and physically different) agent. The heterogeneity of the communicative agents is an aspect which is often neglected in linguistic studies.

### 1.3 Thesis Plan

This thesis is divided into fourteen chapters. The second chapter of this thesis gives an overview of the literature background to this work. It is a brief introduction to the psychological and linguistic theories behind our approach to learning a symbolic communication system. Chapter 3 presents the hypothesis at the basis of this work and compare these to approaches of related works in robotics and computational linguistics. Chapters 4 and 5 give a formal mathematical description and theoretical analysis of the performance of the connectionist architecture, DRAMA, which we develop to

control the learning process and behaviour of the robot. A brief review of related connectionist models is given in the second section of chapter 4. However, reading of these chapters requires one to be familiar with mathematical notations and to have some background knowledge of connectionist architectures. Chapters 6 to 11 report on the implementation of the DRAMA architecture in different robotic set-ups for different experimental studies. Chapter 6 gives a brief overview of the scenario of each experiment reported in chapters seven to eleven. Chapter 7 reports on four experiments in which a mobile robot is taught by a teacher robot and a human instructor a vocabulary to describe its perceptions and actions. Chapter 8 describes experiments, in which we study scaling up of the teaching method to transmit a vocabulary among a group of robots. Chapter 9 presents three case studies, in which the robot or its user benefits from the robot's understanding of a vocabulary, as it facilitates the interaction between the two agents. Chapter 10 presents an experiment on learning time series of a robot's perceptions, when travelling in a series of corridors. Chapter 11 describes experiments with a doll-like robot, called *Robota*. Through an imitation game, whereby the robot mirrors arm and head movements of a human, the robot is taught to perform and label sequences of actions. Further, *Robota* is taught a set of English proto-sentences to describe its interaction with the teacher. Finally, we present a set of tests, carried out with children of 5 and 6 years old, in which the children taught the robot simple words and combinations of movements.

Chapter 12 summarises and discusses the achievements of the work presented in the previous chapters. The reader can find there an overview of the work reported in this thesis and a discussion of its contribution to current research on related topics. It also presents some possible directions of research following from this work. Chapter 13 gives a brief summary of the main results of the thesis. In appendix D, we give a list of the publications, which reported on different parts of this work.

## Chapter 2

# Background work

This chapter makes a short review of related work in research areas most closely related to the work of this thesis, namely research in *Learning Robotics*, *Social Robotics* and *Linguistics*. In the first section, we describe different learning algorithms which have been implemented in robotic experiments, mentioning in particular connectionist models, as we use one in this work. In the second section, we present different approaches to studying social behaviour in robots, either as part of a group of multiple robots (collaborative robotics) or as part of human-robot interaction. In the third section, we present linguistic studies, which model the development and processing of natural (human) language. We describe first work in *Computational Linguistics*, which study, through computer simulations, scenarios for the evolution of language, and then work in *Natural Language Processing*, which develop algorithms for producing and understanding natural language by a computer. The last section of this chapter briefly summarises the main aspects of each of these works which are directly relevant to our research. In chapter 3, we present our approach and compare it to the works presented here.

### 2.1 Learning robotics

Research on learning robotics is concerned with the development and implementation of learning algorithms for the control of robots. A learning algorithm is a program which has the ability to learn from a set of examples. Learning algorithms can be distinguished by the type of learning they can perform (e.g. pattern memorisation,

pattern classification and generalisation), and the type of algorithm at the basis of their computation. A learning algorithm consists of optimising sets of parameters internal to the program (the robot's controller or other type of program), such that the program computes the desired function (memory, classifier, etc.) onto the set of training patterns. Mobile Robotics uses three main types of learning algorithm which are Reinforcement Learning (RL), Artificial Neural Networks (ANN), also called connectionist models, and Evolutionary Algorithms (EA). Note that artificial neural networks are more than just learning algorithms. They are computational models whose computational capabilities are exploited in a large domain of application (see section 2.1.2). In robotics, there are used as control system for directing the robot's behaviour and the robot's learning.

### 2.1.1 Reinforcement learning

Reinforcement learning is based on the idea that the robot receives rewards (*feedback signal* [Wyatt 97]), which are negative or positive scalar values, based on its performance. Depending on the reward, the agent reinforces (positive reward) or decreases (negative reward) its confidence on the correctness of its current behaviour. The rewards are given by an external coach or by an evaluation function (*reinforcement function* [Wyatt 97]) internal to the robot's controller. Most methods relies on a pre-defined model of the world (Markov model), which is described as a fixed set of discrete possible states for the agent's sensor-actuator. A Markov model represents the robot's behaviour as a series of transitions across a finite set of sensor-actuator states; the possible transitions from a state to another are predetermined. RL relies on the Markov assumption, that the next state is entirely determined by the current state of the model. This turns out to be an important limitation in robotics, as the robot has only access to a limited number of state variables (given its limited sensor perceptions), which are often not sufficient to distinguish between two sensory similar situations. Similar situations could however be distinguished if one would have access to the history of states preceding the current one. Hidden Markov Models were developed to cope with this problem, where hidden parameters are used to encapsulate all the information of the previous steps. These algorithms have been used very successfully in Natural

Language Processing research (see section 2.3.2), but have not been applied frequently to robotics. Another limitation of RL lies in its relying on a predefined representation of the world (Markov model), which can lead to important degradation of the robot's behaviour, if the representation is not consistent with change in the environmental dynamics (see discussion of section 1.1).

The robot adapts its behaviour by varying the probability function (*its policy*), which determine the transition between each of the sensor-actuator states. The learning algorithm converges to the behaviour which maximises the reinforcement, i.e. the sum of rewards over time. Important is the fact that the robot receives usually a reward after several time steps and has thus to extrapolate which of its choices of actions during the previous time steps was correct or incorrect (the extrapolation over time is known as *the credit assignment problem*). An important characteristic of RL methods compared to EA and most ANN is that it is based on dynamic computation, which is sufficiently inexpensive to allow computation to be performed on-line (*on-line learning*) and on-board of the robot. Learning is however quite slow [Wyatt 97], which makes the model not reliable in the face of fast and important change in the environmental dynamics. On-line learning relates to the fact that the robot adapts its behaviour dynamically while running and collecting data. This is a fundamental characteristic for making the robot really adaptive and robust in its daily functioning. RL Algorithms have been applied to several robotic learning tasks, mostly based on action selection, that is, devising the correct action to be performed for a given perceptual context, e.g. for navigation [Gaussier *et al.* 98c, Mataric 94a], object manipulation (foraging, box/ball pushing) [Asada *et al.* 97], [Mataric 97c], [Wyatt *et al.* 98], robots communication [Mataric 97b, Yanco & Stein 93].

Finally, note that there exist several variations of the reinforcement learning algorithm, of which we presented here the general ideas. A review of these is beyond the scope of this thesis and the knowledge of the present author. The reader can refer, e.g., to Sutton [Sutton & Barto 98] and Wyatt [Wyatt 97, Wyatt 98] for a good review of the field.

### 2.1.2 Artificial Neural Networks

Artificial Neural Networks (ANN), also called connectionist models, are simplified models of the human brain processing, whose functioning is based on parallel computation of networks of neurons. An ANN consists of a network of *neurons* or units, which implement mathematical functions. The way the units are combined in the network determines the overall mathematical function which is computed by the network. A network is usually composed of a set of input and output units linked by uni- or bi-directional connections between each unit. Each connection is associated with a *weight* which determines the strength of connectivity between the units linked by this connection. The *activity* or value of an output unit is equal to the result of its *activation function* on the sum of activity of the input units multiplied by the corresponding connection weight.

There exist several types of ANN which differ both in their internal functioning and their computational properties, e.g. associative memory, topographical map, pattern recognisers, etc. For an extended list of these, the reader can refer to [Arbib 95]. In section 4.2, we describe in more detail some of these models, whose properties are relevant to our problem, namely the associative memory models, and in particular Hebbian networks and the Willshaw Net, and the recurrent neural network models.

Connectionist models have been shown theoretically to have general computational properties; for instance, feed forward networks (which have only one direction of connectivity from input to output units) with one hidden layer (this corresponds to a layer of units between the input and output units) and recurrent neural networks (feed forward networks with backwards connections from output units to some or all input and hidden units) are capable of computing any continuous function [Pineda 87]. For this reason, they have been applied to diverse domains (e.g. particle physics, financial market, medical diagnosis) for data classification .

Connectionist models are used in robotics, e.g., to determine the robot's behaviour, by defining a mapping between the network inputs, which correspond to the robot's sensor measurements, and the network outputs, which determine the robot's actuator activity. The mapping between input and output units in the network determines the

correlation between perceptions and actions of the robot, which further determines the robot's behaviour. Learning corresponds to updating the network connectivity, such as to represent the desired sensor-actuator correlations.

Connectionist models have been used for many different learning tasks in mobile robotics, for example, to develop basic navigation behaviours such as obstacle avoidance [Hallam *et al.* 92, Salomon 98] or phototaxis behaviour [Lee *et al.* 96], for guided navigation based on landmark recognition [Gaussier & Zrehen 95, Harris & Recce 97, Zrehen 95] and on prediction of sequences of the robot's action-perception [Nehmzow 98, Tani & Fukumura 95], and for object manipulation [Pfeifer & Scheier 98]. These methods were shown to generalise well, given a relatively small set of data, and to be able to separate relevant data from significant amounts of noisy (randomly variable) data. These characteristics make connectionist models particularly relevant to robot learning, since a robot can access only a small, and, thus, often biased, amount of data (if one wants to keep the time needed to collect data reasonable) and has to cope with an important amount of irrelevant data (noise). However, ANNs rely often on heavy and time consuming computations, exceptions being Hebbian networks as used by [Pfeifer & Scheier 98, Hallam *et al.* 92, Walker *et al.* 93], which prevents their implementation for on-line (and on-board) learning of a robot. In this case, training of the network is done off-line, using data which the robot has collected in a first travel through the environment. In this thesis, we develop a connectionist model which has more computational power than usual Hebbian networks, as it allows learning of time series (similarly to recurrent neural networks), while being sufficiently computationally inexpensive to allow on-line learning of a robot (by contrast to other recurrent neural networks which have to be trained off-line). See chapters 4, 5 and 12 for a detailed description of this model and a discussion of the model property relative to other connectionist models.

### 2.1.3 Evolutionary Algorithms

Evolutionary algorithms (EA) are another method used to design the structure and parameters of a robot's controller. They differ from RL and ANN models basically in the fact that they are not used as part of the robot's controller. The learning process

occurs within a population of agents and on the time scale of several generations of agents rather than on one agent's life time. The algorithm updates the agent's controller structure as a function of the agent's performance during a 'life-time' period. The principle of EA is based on an analogy to the Darwinian model of evolution by natural selection. The controllers of the robots are described by *chromosomes*, one for each robot. The evolutionary algorithm generates new solutions for the robots' controllers, by *breeding* the chromosomes, using biologically equivalent processes of *cross-over* and *mutation*. The elements of the chromosomes can either encode the parameters of the controller, in which case the technique is called a Genetic Algorithm (GA) [Holland 75], or rules which determine the controller structure, a technique called Genetic Programming (GP) [Koza 92]. Chromosomes are ranked depending on their performance, or rather on the performance of the function or system they represent on the assigned task, which is evaluated through a *fitness function*. The chromosomes with maximal performance are bred and the resulting chromosomes replace the less well performing ones in the set. Because the principle at the basis of EAs is not domain specific, EAs are very general optimisation techniques, which have been successfully applied to diverse domains such as Economics, Engineering and Artificial Intelligence.

In robotics, EAs have been used to develop robot controllers to produce, e.g., basic behaviours of obstacle avoidance and phototaxis as part of navigation tasks [Miglino *et al.* 95, Lee *et al.* 96, Perkins & Hayes 97, Wilson *et al.* 97], as part of collective tasks [Nolfi 97b, Nolfi 97a], or as part of competitive tasks, prey-predator game, [Floreano *et al.* 98]. Particularly interesting is their implementation for designing complex ANN structures, which might prove too difficult to be developed by hand. For instance, GA and GP have been used to develop complex recurrent neural network structures, which result in a hierarchical behaviour controller [Angeline *et al.* 94, Wilson *et al.* 97], and for locomotion controllers, using structure of non linear time dependent neurons [Beer & Gallagher 92, Yamauchi & Beer 94, Ijspeert *et al.* 98, Kodjabachian & Meyer 98].

Because of the wide range of successful applications of EAs and the lack of mathematical analysis of their performance, EAs appear somewhat as "wonder boxes", which have none or few limitations on the complexity of the problem they could solve. How-

ever, in practice, it appears that the chance that a GA or GP produces a relevant solution relies on a good design of the fitness function, that is the evaluation function of the robot's behaviour, and a good choice of the encoding of the solution (chromosome representation). Choosing these parameters correctly depends on having a particularly good idea of the form and characteristic of the desired controller, which would sometimes be enough to design a first, non optimal version by hand.

## 2.2 Social Robotics

The term 'social' has appeared recently in several robotics papers, see, e.g., [Asoh *et al.* 97], [Breazeal & Scassellati 98], [Brooks *et al.* 98a], [Canamero & Van deVelde 97], [Dautenhahn 98], [Dautenhahn 95], [Kuniyoshi 94], [Mataric 97a], [Pfeifer 96], [Scassellati 98]. However, its use has been quite diverse and has seldom been explicitly defined. For the purpose of this thesis, we define the expression *social robotics* to refer generally to research in robotics, which studies systems showing some form of social behaviour. We follow the definition of the term social, given by the 1995 Oxford dictionary [Oxford 95]: "1) relating to society or its organisation; 2) cooperative; practising the division of labour; 3) concerned with the mutual relations of human beings or of classes of human beings". We define social robotics as referring to the following lines of research:

- *Collective robotics*, which is the study of multi-agent systems, characterised by self-organisation and emergent functionality [Steels 94b], insect-like agents (their behaviour and level of cognition) and stigmergic communication [Beckers *et al.* 94, Mataric 94a].
- *Cooperative or collaborative robotics*<sup>1</sup>, which studies multi-agent systems engaged in collaborative or cooperative behaviour. Cooperative robotics is a sub-field of collective robotics [Uny Cao *et al.* 95].
- *Socially intelligent robotics*, which is concerned with the building of robots capable of engaging in complex social interactions, similar to that observed in high

---

<sup>1</sup> We did not find in the literature an explicit distinction made between collaborative robotics and cooperative robotics.

primates, with humans or with other robots.

In the rest of this section, we describe briefly some research directions followed by studies in cooperative robotics and socially intelligent robotics, pointing out their relevance to this thesis.

### 2.2.1 Cooperative robotics

Cooperative or collaborative robotics studies systems of multiple robots, which interact cooperatively. Cooperative behaviour refers to “a subclass of collective behaviour that is characterised by cooperation” [Uny Cao *et al.* 95]. It is “a form of interaction generally based on communication” [Mataric 95], “a joint collaborative behaviour, which is directed toward some goal in which there is a common interest or reward” [Barnes & Gray 91]. Multi-agents systems can be cooperative by design, in which case “the two or more robots are aware of each other’s existence, and can sense and recognise each other directly or through communication”, or it can entail “implicit cooperation, in which the robots usually do not recognise each other but merely coexist and indirectly cooperate by having identical or at least compatible goals” ([Mataric 94a] p.37). Surveys of work in this field can be found in ([Uny Cao *et al.* 95], [Mataric 95, Mataric 94a], [Parker 94b]).

Research on collaborative robotics takes inspiration from studies of insect societies, e.g. ants, bees and other eusocial insects, which are composed of agents capable of simple behaviour but which in a group can accomplish quite sophisticated tasks, e.g. nest building and foraging [Wilson 71]. People study the *emergence* of complex group behaviour from multiple robot systems, in which each robot is only capable of simple behaviours, e.g., obstacle avoidance, wall following, phototaxis. All these systems implement group behaviour, following a particular control architecture, e.g., the SWARM architecture [Beny 88], which consists of a distributed system with a large number of autonomous agents, the CEBOT [Fukuda & Kawauchi 93] system, which refers to a decentralised hierarchical architecture inspired by cellular organisation of biological entities, the ALLIANCE architecture [Parker 98], which describes a system made of a group of heterogeneous agents, all controlled by a behaviour-based architecture.

Group behaviours which have been studied through robotic experiments are, for instance, flocking behaviour (collective locomotion) ([Arnaud 97], [Mataric 94a], [Vaughan *et al.* 98]), collaborative transportation [Bay 95, Johnson & Bay 95] or manipulation of objects [Asada *et al.* 97, Martinoli *et al.* 97], collective manipulation [Steels 94a] or grouping of objects, involving sorting the objects [Melhuish *et al.* 98, Pfeifer & Scheier 98] and then grouping the objects by either pushing ([Wyatt *et al.* 98], [Parker 94b]) or holding them ([Martinoli & Mondada 95], [Mataric 94a]). Also, interesting to mention is the recent application of multiple robot systems for simulating popular games or entertainment activities, such as the soccer game [Kitano 98] and dance/theater performances [Werger 98].

There is also an engineering motivation behind research on collaborative robotics, where people investigate whether it is more efficient to distribute the area of expertise needed for performing a complicated task between several robots rather than designing a unique multiple expert robot. There are several issues underlying the success of the collaborative task, when performed by a group of robots. To achieve robust and efficient coordination of the group relies on a good choice for the basic behaviours which the robots are provided with, such as to get a good tradeoff between competing and collaborative behaviours [Mataric 94a, Parker 94b]. The number of robots used for the task is crucial; too many robots can lead to more destructive than constructive interactions, as was shown for box-pushing tasks and foraging tasks [Martinoli *et al.* 97, Parker 94b]. This last issue can be compensated, by introducing heterogeneity (for the behaviours or bodies) among the robots [Parker 94b]. People also investigate the use of learning techniques to optimise the agents' behavioural parameters, such as to obtain maximal efficiency for the performance of the group of agent, using, e.g., reinforcement learning techniques [Mataric 97c], task oriented action selection mechanism [Parker 98] or genetic algorithms [Floreano *et al.* 98, Steels 94b].

Of particular interest to us are studies which showed that using explicit symbolic communication could improve the performance of a group of robots in a collaborative task. For instance, in Steels's experiment [Steels 94a], the life duration of a group of robots is improved when the robots can communicate with each other. Using explicit predefined radio signals, they can inform each other of the current energy level of their

battery. As a result, one robot can leave the recharging station, while its battery has not been completely recharged in order to allow another robot to recharge its battery before it dies out. L. Parker [Parker 94a] compares the performance of a group of robots in a puck-collecting task when the robots can or cannot use explicit communication to report on their current achievement. She shows that communication is a factor which improves the performance of the group, when the cost of redundancy of actions is important. Redundancy results from having homogeneous robots, that is, robots capable of the same tasks. A tradeoff has to be found between diminishing redundancy (using heterogeneous agents) and keeping the system reliable (by duplicating the agents performing the same tasks), which can be balanced by using explicit communication. Mataric et al. [Mataric 97b] shows how the *hidden state* problem (namely the problem of not having access to all the information necessary to perform the task) and the *Credit assignment problem* of the reinforcement learning paradigm can be lowered by using direct communication between the collaborative agents to transmit sensing and reinforcement information.

### 2.2.2 Socially intelligent robotics

We introduce the expression *Socially intelligent robotics*<sup>2</sup> to refer to recent trends in robotics, which investigate the design of robots capable of engaging in high-level social interactions with humans (or with other robots). In contrast to collective robotics, which studies insect like behaviours, socially intelligent robotics is concerned with complex social interactions, similar to those observed in primate societies. The work in this area often bases its design principle on studies of social intelligence and social learning in animals (see section 3.1 for definition), which are concerned with the understanding of the cognitive and behavioural skills behind these high-level social skills. Issues investigated are the means of communicating symbolically, the ability for learning by observation or by imitation, the need for social rules.

The use of explicit, symbolic communication is an important means of interaction

---

<sup>2</sup> The expression *socially intelligent robotics* was inspired by the title of the workshop on 'Socially Situated Intelligence', organised by B. Edmonds and K. Dautenhahn, held as part of the Fifth International Conference of The Society for Adaptive Behaviour, SAB'98, in Zurich (CH), August 17-21.

between a human and a robot. Communication can, e.g., be used for teaching or commanding the robot [Crangle & Suppes 94, Klingspor *et al.* 97]. The recently developed speech processors, which are algorithms (and hardware) capable of understanding and producing natural language, have been used to communicate with a robot for different applications. A number of works report on teaching a robot names for locations in an office environment, while the communication protocol is either built-in [Asoh *et al.* 97, Tatsuno 96], or is learned by the robot as part of the landmark learning task [Henis & Levinson 95, Torrance 92b]. Particularly relevant to the work of this thesis are studies on building learning capabilities for the robot which could allow it to develop its understanding of the language. We describe these further in section 2.3.1 of this chapter.

Inspired by studies on learning by imitation in primates (see section 1.1.3), people have developed methods based on imitative skills to allow robot teaching by demonstration. Imitation is an interesting means for guiding the robot's attention. It can thus be used more generally to guide the robot through situations about which it could be taught. For instance, the imitation interaction leads imitators and imitated agents to make similar proprioceptions (e.g. of movement, inclination, energy consumption) and similar exteroceptions, as they share simultaneously the same physical space. Thus, the robot's ability of imitating the actions of another agent can be used to teach the robot new motor skills by having it observing and then imitating the actions of that of a demonstrator [Demiris *et al.* 97, Cooke *et al.* 97, Kuniyoshi & Inoue 94]. From a programming point of view, this amounts to solving at least three problems: 1) how to get the agent to observe the example, that is how to constrain the agent's attention to observing only the relevant stimuli [Scassellati 98], 2) how to get the agent to reproduce the observed action pattern, which amounts to defining a mapping process from the agent's sensor (visual) stimuli (observation of the other actions) to the agent's actuators [Cooke *et al.* 97, Demiris & Mataric 98, Mataric 94b], 3) what to learn from the reproduction/imitation process, e.g. learning action sequence patterns ([Hayes & Demiris 94], [Mataric 94b], [Kuniyoshi 95], [Schaal 97]) or patterns of perception-action sequence [Demiris & Hayes 96, Dautenhahn 95, Gaussier *et al.* 98b].

An important part of this research is directly inspired by physiological models

of primates' cognitive processes related to similar skills. For instance, Demiris [Demiris 98, Demiris & Hayes 96] develops his model of imitation from studies of imitation games in human babies [Meltzoff 90] and in primates [Whiten & Ham 92]. Scassellati [Scassellati 98] inspires his model of visual attentional mechanisms from psychological and neurophysiological studies of humans and primates. Kuniyoshi et al. [Bakker & Kuniyoshi 96] conduct a vast project which attempts to reproduce Piaget's model of infant development. Brooks et al. [Brooks *et al.* 98a] develop a humanoid robot which would show cognitive and behavioural skills similar to that of a human. They input a lot of effort to building each cognitive and behavioural module, following biological and psychological models of the corresponding skills in primates and humans. Finally, interesting to mention are also the works which study the design of emotions, as built-in behaviours or drives underlying the robot's control processes. The aim behind these studies is two-fold: to understand the dynamic behind human emotions better and to use emotions as a schema for guiding the robot's autonomy of decision ([Canamero & Van deVelde 97], [Pfeifer 94]). Particularly interesting to us are the works which develop robots capable of showing facial [Breazeal & Scassellati 98] or behavioural ([Fujita & Kitano 98], [Zrehen *et al.* 98]) expressions of emotions, which they use as a means to engage in body communication with humans.

## 2.3 Linguistics

Linguistics is the study of human language. While there exist and have existed numerous languages spoken by humans, interestingly, they all seem to share common features. These are the use of symbols and of syntactic rules. Symbols are arbitrary signals, which are used to transmit concepts, that is, abstract representations of the world. A syntax consists of a set of rules which assign a functionality (nouns, verbs, adjectives, etc.) to the symbols depending on their meaning and which determine the order in which to combine the symbols, depending on the symbols functionality and the meaning one wants to transmit. These features seem to be unique to human language and to have no equivalent in other animals' communication systems<sup>3</sup>. For these

---

<sup>3</sup> This is a very controversial subject. For instance, people argue that some type of apes might be capable of simple forms of language. Their argumentation is based, e.g., on the successful experiment

reasons, the study of human language seeks to bring insight into the particularity of human cognition in contrast to that of other animals. Note that the term language refers generally only to human language (see section 3.1 for a definition of the Linguistic terminology, which we use in this chapter and in the rest of the thesis).

This section presents two main trends of research in this field which are related to the topic of this thesis. These are researches on *Evolutionary linguistics*, which study the means of development and transmission of language across a population of agents, and research on *Natural Language Processing* which develops tools for the processing of human speech.

### 2.3.1 Evolutionary linguistics

Evolutionary linguistics is the study of language viewed as a complex adaptive system. Among the issues addressed by this field are the questions of how languages arise, how they change over time, and how they can be learned. Language is seen as the product of local interactions between language users from which the global characteristics of the language emerge by selectionist mechanisms. Language is not created by some central agent; it is an emergent phenomenon that arises spontaneously [Steels 96b].

Computational linguistics is the study, through computer simulations, of the development of synthetic languages. These works study different aspects of a language's evolution. For instance, they formulate and investigate different hypotheses for the evolution of symbolic communication and then of language. Several scenarios for the evolution of language have been devised by theoretical linguists, which are based on physiological studies of the central nervous system (e.g. [Deacon 97, Pinker 94]) or on sociological studies of primates and human societies (e.g. [Aitchison 96, Dunbar 93]). Some of these have been implemented through computational studies (e.g. [Arita & Koyama 98, Kirby & Hurford 97b, MacLennan 91, Di Paulo 98, Steels 96b]). In this aspect, particularly interesting is the work of Bart de Boer [deBoer 97], who investigates through simulations the hypothesis that the development of language might be constrained

---

of Savage-Rumbaugh [Savage-Rumbaugh 93] at teaching chimpanzees a basic lexicon and the studies of whales complicated songs [AAS 83] and birds duets [McFarland & Steels 97]. For some insight into this debate, see [Bickerton 90, Deacon 97, Wilson 80].

by the physics of the vocal tract. Using a model of the human articulatory system, he shows how a number of consonants and basic vowels can be produced which are common to all languages.

A second important aspect of a language dynamic which has been studied at length by computational linguistics is the language transformation (formation of dialects, transfer of word meaning) over time. This can be the result either of an erroneous transmission of the word meanings across generations of agents [Kirby & Hurford 97b] or of a composition of meaning resulting from mixing the language definitions of different populations of agents [Steels 97b].

Directly related to the evolution of a language is the study of its formation, as the effect of a social agreement of two or more agents on using a common definition for similar concepts. There are several issues underlying the understanding and use of a language. A language is a means of transmission of concepts. For this, it uses symbols, which are arbitrary signals which different agents have agreed to use in order to represent similar concepts they have individually formed. From the perception and understanding of the language to the reproduction of language, there are several key steps, which can be enumerated as such: 1) segmentation of sound input into sound strings (words recognition); 2) *meaning* extraction of the sound strings through the creation of a *lexicon*<sup>4</sup> (a lexicon is a set of meaningful symbols) 3) learning of a *syntax* and a grammar, which means to recognise regularities behind the combination of terms of the lexicon and to attach a meaning and a grammatical functionality to each term; 4) forming new combination of terms of the lexicon, using the syntactic rules, to convey new meanings.

Numerous simulation studies have been done to model these diverse aspects of the formation of language. For instance, people investigated how two agents can agree on using a common lexicon [Arita & Koyama 98, Kirby & Hurford 97b, MacLennan 91, Steels 96a], and further how they can categorise their perceptions (formation of concepts) in order to achieve coherence in their usage of the lexicon [Steels 97b, Steels 97a].

---

<sup>4</sup> Being no linguists myself, I apologise if linguist people get confused by my using of their terminology and my very simplified definition of it. My aim was not to redefine the linguistic terminology, but to present a brief account of the main ideas behind research in this area, which might be biased by my own view of the problem.

All of these models are based on combinatorial analysis. That is, categorisation of sensor perceptions into concepts and mapping between concepts and symbols results from a process of statistical elimination among all possible meaning-object pairs, where the most likely pairs, i.e. the most frequently observed, are chosen. The associative process was simulated, e.g. as the input/output of an artificial neural network [Kirby & Hurford 97a], as a matching process [Steels 96b] or as a probability function [Oliphant & Batali 97]. Further, people investigated the development of syntax from the learning of a lexicon [Kirby 98, Steels 98].

### **Grounding a vocabulary onto perceptions of physical agents**

Of particular interest to us are the works of Yanco [Yanco & Stein 93, Yanco 94] and Steels & Vogt [Steels & Vogt 97, Vogt 97] who studied the development of a shared lexicon by autonomous mobile robots.

In H. Yanco's experiments, two (and sometimes three) robots develop a common vocabulary to designate their actions. The scenario of the experiment consists of achieving coordination of movements between different agents. There is a leader agent and one or two follower agents. The leader agent has access to external cues, radio signals emitted by a human coach, which none of the follower agents can perceive. The task of the leader agent is two-fold: it has 1) to learn which action to perform (spinning, moving straight, turning left/right) for each signal and 2) to learn what (radio) signal to send to the follower agent which designates the action to perform. The follower agent has to learn to perform the correct actions for each of the leader's signals. It is a supervised learning strategy based on reinforcement learning. The robots receive reinforcement after each trial; they receive positive or negative reinforcement depending on whether they have chosen to perform the correct action. The experiments are carried out in simulation with two or three robots and in a physical set-up using two mobile robots (Ernie and Bert), while varying the size of the vocabulary from two to twenty words. Qualitative and quantitative evaluations of the experiments show that the agents converge successfully to the same usage of the vocabulary in all cases and that the number of steps to convergence increases exponentially (by a factor 10) with the size of the vocabulary.

In Steels & Vogt [Vogt 97]'s experiment, two mobile robots develop a common vocabulary to designate different objects in their environment, namely three boxes emitting light at different frequency, and to name themselves (infra-red detection). The scenario of the experiment consists of running a set of *language games*, as developed previously by L. Steels in simulation [Steels 96b]. One language game consists of having two agents, a speaker and a listener agent. The speaker agent chooses a topic of conversation. It selects a feature out of a set which it can perceive, and a word out of a predefined set to label the feature. The hearer agent interprets the speaker's word by associating it with one of the features it has perceived. If the hearer has chosen the same feature as that chosen by the teacher, the game ends up as a success; otherwise it is a failure. The hearer agent keeps a record of the number of times a pair word-feature has been selected, and, when it is its turn to speak, it chooses the word which has been paired most frequently with the feature it wants to describe. The experiment is said to be successful if the ratio of success over failure increases with successive playing of the game and ends up with only successful games.

In the physical implementation of the language game, the speaker robot first 'points verbally' (via radio communication) to one of the four visual quadrants, before speaking. This allows it to reduce the set of possible features among which the hearer should discriminate, therefore speeding up the convergence process [Vogt 98]. The authors showed that they could obtain a maximal convergence of 90% success (ratio of the number of successful games over total number of games) within 5000 games.

### 2.3.2 Natural language processing

Research in natural language processing (NLP) studies the development of systems capable of interpreting and producing speech. It is a cross study between research in Artificial Intelligence and research in Linguistics, as it uses mathematical tools of the first in order to implement models of human processing of speech of the second. Its aim is strongly linked to industrial applications in the telecommunication market, such as the development of devices capable of translating verbal speech into written text (and vice-versa) and of programs capable of dialoguing with a human user.

Constructing a program capable of interpreting and producing human verbal speech

amounts to solving two main issues: 1) how to parse continuous sound input, such as formed by human speech, in order to extract words and sentences and 2) how to interpret the sentence, that is to understand the meaning behind the particular combination of words. There are, of course, numerous issues related to these two points, e.g. solving ambiguous meaning, integrating exceptions to grammatical rules, etc., which we do not discuss here, as they are not directly relevant to our problem (the interested reader can refer to [Mellish *et al.* 94, Ritchie 94]). Relevant to us is the type of mathematical tools which are used for parsing and producing speech.

As mentioned in the introduction of this section, the principal characteristic of human language is its syntax, that is, its particular way of combining symbols to form sentences, whose meaning is determined by the particular ordering of the symbols. Speech processing requires the ability to discriminate between significant and noisy signals (word parsing) and the ability for processing recursion, that is, a short-term memory for recording sequences and combinations of words. The tools used for this purpose are Hidden Markov Models (HMMs) (see [Bengio 96b, Rabiner 89] for a review) and Neural Networks (NN), and in particular Recurrent Neural Networks (RNN) (see [Giles 92] for a review). Examples of implementations of HMMs and NNs (or a combination of the two), for learning regular syntactical and natural languages are, e.g., [Bengio 95, Morgan & Bourlard 90, Kolen 94, Pollack 87] respectively. In this thesis, we develop a RNN model, which has similar properties of sequence processing as those of other RNNs and of HMMs (see discussion in section 5.2.2) and implement it in a robotic experiment, in which a robot is taught English proto-sentences to describe its actions and perceptions (see chapter 11).

## 2.4 Summary

This chapter presented a brief review of related works in the robotics research domains, which are concerned with the implementation of learning and social skills, and works in the linguistic domain, which study the development of an agent's understanding of a language. We presented three learning techniques used most often in robotics, namely Reinforcement Learning (RL), Artificial Neural Networks (ANN), and Evolutionary Algorithms (EA). Of particular interest to us are ANNs, as they perform well in the

face of noisy data and are thus very relevant for our robotic implementation. In this thesis, we develop a connectionist model which allows learning of time series.

We presented two trends in Social Robotics: research on collaborative robotics, which studies multiple robots systems, and research on socially intelligent robotics, which investigates the design of robots capable of engaging in high-level social interactions with humans or with other robots. Interesting to us is work on collaborative robotics, which showed that the performance of a group of robots, engaged in a collective task, could be improved by using explicit communication between the robots. Relevant to the topic of the present study are researches in socially intelligent robotics, which investigate the design of communicative and imitative skills for experiments on robot teaching by demonstration. The work of this thesis contributes to research in socially intelligent robotics.

We then mentioned two research areas in Linguistics, namely Evolutionary Linguistics, which simulates the dynamics of language formation and evolution, and Natural Language Processing, which develops algorithms for processing and producing human speech. Particularly relevant to us are the works in applied computational linguistics of Yanco and Stein [Yanco & Stein 93] and Steels and Vogt [Steels & Vogt 97], who implemented learning of a lexicon by autonomous robots. We take insight from Natural Language Processing studies on the capacity of mathematical algorithms, such as recurrent neural networks, for sequence processing, which is a basic ability for producing and understanding a language.

The work of this thesis is inspired by research in these different areas. In chapter 3 we describe our approach and compare it to related works in Robotics and Linguistics which we presented in this chapter.

## Chapter 3

# Work hypotheses

This chapter presents the hypotheses underlying the work of this thesis. We study how to teach a robot a synthetic language to describe its perception of the world. Key issues which we address are 1) how to solve the *symbol grounding problem*, which is grounding arbitrary signals (symbols) onto the agent's perceptions; 2) how to develop a similar symbolic representation of the world by two functionally different agents.

Inspired by psychological and linguistic studies of the development of symbolic communication in human infants and primates societies, we determine here a number of key features for the cognitive and behavioural mechanisms of our agent. We introduce the ideas which brings us to making these assumptions in the second and third sections of this chapter. In the fourth section, we contrast our approach to that of other works in related areas. In the fifth and last section, we summarise the issues presented in the previous sections and give a list of hypotheses on which we base our work. We begin this chapter by a section in which we define the terminology we use throughout this thesis.

### 3.1 Terminology

In order to avoid any terminological misunderstandings, we begin with a short definition of key terms used in this chapter and in the rest of this manuscript.

**Communication and symbols:** The term *communication* refers generally to any process of interactive exchange between at least two agents. One can distinguish

between indirect communication, based on the observation of one agent's behaviour, and direct communication, based on intentional information transmission by one agent to another. Indirect communication is, e.g., the bird's interpretation of its conspecific's flying behaviour, as meaning that a predator is approaching. Direct communication is, for instance, the honey bees' waggle dance, which forager bees use to transmit to each other information about flowers' locations (see e.g. [McFarland 93, Wilson 80] for a description of diverse communication systems of animals). *Symbolic communication* refers to a form of communication which transfers information from one agent to a second agent by means of symbolic signals. Symbolic signals or *symbols* are arbitrary physical signals which have a *meaning*, that is, an interpretation, for an agent. A signal is said to be symbolic when its being associated to a specific meaning is the result of a social convention between two or more agents. Symbolic signals are to be distinguished from iconic signals, which refer to signals whose meaning is derived from analogy between the signal features and that of the object it designates (iconic signals are, e.g., pictures, drawings, etc.), and indexical signals, which are linked to objects through a causal relationship, such as a temporal precedence between the signal and the observation of its referent (an indexical signal is, e.g., the smoke before fire, the bending of trees by a windy day, etc.). We say that two agents are *communicating*, once they have developed a similar interpretation of a set of arbitrary signals in terms of their own sensor perceptions, that is, once they have achieved a similar *categorisation* [Harnad 90] of sensor perceptions and have successfully attached to them the same set of arbitrary signals.

**Meaning:** A symbolic signal is meaningful or has acquired meaning in our robotic experiments, when it has been assigned an interpretation in terms of the robot's perceptions, that is, when it has been associated with specific states of the robot's sensors and actuators.

**Language:** We use the term *language* to refer to any communication system which shows similar characteristics to that of human language, that is, the use of symbols to convey meaning, syntactic rules for the combination of the symbols into sentences and other grammatical rules for the semantics of the sentences.

**Proprio- and exteroceptive perception:** We distinguish between proprioceptive and exteroceptive perceptions of the robot. Following the definition of the Oxford dictionary [Oxford 95], proprioceptive relates to "stimuli produced and perceived within an organism, especially relating to the position and movement of the body", while exteroceptive relates to "stimuli produced outside an organism". In the experiments, the proprioceptive perceptions of the robot are its perception of movement (actuator state), of energy consumption (energy level sensor), of inclination (tilt sensors) and of orientation (relative to a compass). The exteroceptive perceptions are the robot's perception of light (produced by the second robot), of infra-red (detection of objects), of touch through the bumpers on the vehicle robots and the switches on the doll robot's body (object detection) and of radio waves (means of communication with the robot).

**Representation and concept:** We use the term *representation* in two occasions: 1) We speak of *conceptual representation* or *concepts* to refer to the robot's extraction of spatio-temporal invariance in its sensor-actuator state space. This definition is similar to Harnad's notion of *categorical representation*: "categorical representations [...] are learned and innate feature-detectors that pick out the invariant features of object and event categories from their sensory projections" [Harnad 90]. Concepts (to us) relate to categories built onto spatio-temporal features of the robot's sensor measurements and actuator states. 2) We use the term *representation* to refer to the functionality of the symbolic signals in our experiments, where these signals label some of the robot's conceptual representations. This definition is similar to Harnad's notion of *symbolic representation*: "Elementary symbols are the names of these object and event categories, assigned on the basis of their (non-symbolic) categorical representations. Symbolic representations [are] grounded in these elementary symbols, [which] consist of symbol strings describing category membership relations" [Harnad 90].

**Sociality:** We use the term *social* on several occasions, to describe the type of interactions the robots have with each other and with the human instructor. We base our definition of *social* on that of the 1995 Oxford dictionary [Oxford 95] and define the term *social robotics* to refer to research in robotics which studies

systems capable of any kind of social skills (see section 2.2.2 for both definitions).

By social interaction, we refer to “1) an interaction involving at least two agents exchanging information 2) an interaction which shows some degree of reciprocity and bidirectionality between the participants and 3) an active involvement of both participants in the interchange, bringing it different experiences and knowledge, both qualitative and quantitative” (definition taken from [Garton 92]).

In section 1.1.3, we refer to the notions of *social intelligence* and *social learning*, used in Psychology. The definition of *social intelligence* varies considerably in the literature [Hemelrijk 95]. This is not very surprising, as the expression is made of two terms, namely ‘social’ and ‘intelligence’, which are themselves very difficult to define. The term was defined in the 1950-1970’s, in order to encapsulate the idea that social interactions might play an important role in the evolution and development of primates’ intelligence. Social intelligence or *Machiavellian intelligence* [Byrne & Whiten 88] relates to cognitive and behavioural skills which an agent develops and uses to interact (in complex ways) with other agents. The list of these skills varies from one author to the other, whether they are described in terms of cognitive processes or behaviours. We retain the following three aspects: 1) The ability for distinguishing between individual agents (recognition of the self and of others) [Dautenhahn 97a, Dautenhahn 95] and of addressing these individually [Edmonds 97]. 2) The ability for modelling other agents’ behaviours [Goody 95]<sup>1</sup>, used for predicting their actions [Edmonds 97] and for learning from their observation ([Byrne & Whiten 88], [Dautenhahn 97a]). 3) The recognition of rules, which structure the interaction of the agents ([Edmonds 97], [Mugny & Carugati 89]<sup>2</sup>). Point one above is strongly linked to the ability of communicating symbolically, i.e. through a language [Goody 97]. Point two

---

<sup>1</sup> “For social intelligence seeks to reach goals and solve problems by modelling and managing others’ reactions to our own actions. Social intelligence means that problem-solving schemata have a slot for modelling the responses of a social Other” [Goody 95] p.207

<sup>2</sup> For [Mugny & Carugati 89], social intelligence develops through social interaction. They report on an attempt to measure the influence of social factors on the development of humans’ intelligence is made by measuring correlations between the results of adults to an IQ (Intelligent Quotient) test and social and cultural differences between these people. For these studies, social intelligence relates to possessing skills of “awareness of and respect for rules and social norms” ([Mugny & Carugati 89], p.85), which the child develops through its diverse social interactions at school and elsewhere.

leads to the ability for imitation ([Byrne & Whiten 88], [Humphrey 88]<sup>3</sup>), and for social learning (see below). *Social learning* refers to the agent's ability for learning from its interaction with other agents. It includes learning by observation and learning by imitation (see below).

In section 2.2.2, we introduced the expression *socially intelligent robotics* to refer to a new trends in robotics, which investigates the design of high-level social skills for robots. These works often base their design principle on studies of social intelligence and social learning in animals. Each of the three above mentioned aspects of social intelligence have been partially and independently addressed by works in AI. The ability for sophisticated means of communication among agents, point one, has been addressed by studies of computational linguistics (see section 2.3.1). Imitation and learning by imitation, point 2, have been used as part of experiments on robot teaching by demonstration (section 2.2.2). The use of social rules for the organisation of a group of agents, point 3, has been studied through computational studies [Edmonds 97, Shoham & Tennenholtz 96] and physical experiments with groups of robots (see the works in collaborative robotics presented in section 2.2.1).

**Imitation:** The term *imitation* refers to a behaviour skill, which leads to "some sort of similarity in behaviour among two or more individuals"[Davis 73]. In our experiments, this skill consists of being able to replicate or mirror the actions of another agent. We do not assume any form of intentionality behind the imitating act. In our experiments, the agent is said to imitate another agent, as a result of it following the other and thus implicitly replicating the other agent's movement in the plane (chapters 7, 8 and 10), or, as result of it mirroring the agent's arm and head movements (chapter 11). Note that the definition and usage of the term 'imitation' is controversial in the literature, see [Demiris 98, Hayes & Demiris 94] for a discussion and review of position in this area.

We distinguish between *learning to imitate* and *learning by imitation*, where the former relates to the learning process leading to an agent's development

---

<sup>3</sup> "The social system serves the purpose [of teaching] in two ways: (i) by allowing a period of prolonged dependence during which young animals, spared the need to fend for themselves, are free to experiment and explore; and (ii) by bringing the young into contact with older, more experienced members of the community from whom they can learn by imitation" [Humphrey 88] p.20.

of imitative skills. In this thesis, we consider only situations in which the agent learns new skills by imitation. In our experiments, the agent's imitative capability is built-in, as a predefined wiring between specific sensors and actuators of the robot.

**Learning:** The agent is said to *learn* when its internal computational structure is shown to change under the effect of its integrating new information.

### Cognitivism versus behaviour-based approach

In AI, cognitivism relates to the classical AI approach (see sections 1.1 and 3.2), which views an intelligent system as a pure cognitive entity, independent of its physical support. The system's behaviour is uniquely and entirely determined by its computational capacities for manipulating symbols, which are abstract representations of the world. Following [Pfeifer 95]: "The cognitivism paradigm can be very briefly characterised by the terms computation, representation and functionalism [...] Cognition is viewed as symbol manipulation [where] computational processes operate on representations, the symbol structures. [...] Functionalism means [...] that thinking and other intelligent functions need not be carried out by means of the same specified machinery in order to reflect the same kinds of processes. In other words, the idea is that intelligence or cognition can be studied at the level of algorithms or computational processes without having to consider the underlying structure of the device on which the algorithm is performed. Briefly we can view cognition as computation" (the above citations are all taken from [Pfeifer 95]). Applied to the design of a robot, this approach "reduces AI to the problem of constructing a brain-in-a-box symbolic manipulator which would act intelligently if given appropriate connection to a robot (or other perceptuo-motor system)" [Brooks & Stein 94].

The *Behaviour-based* approach relates to the line of research introduced by Brooks's subsumption architecture [Brooks 89]. It proposes a framework of control algorithm for a robot, which is based on decentralised control, without reliance on a model of the world (see section 1.1). The Behaviour-based approach should be distinguished from *behaviourism*, which refers to a school of psychology. For general reference, behaviourism was introduced in the 1920's in rejection of intro-

spectionism. It considers that only behaviour which can be observed and measured by more than one person are relevant for psychological study [Gross 96].

In this thesis, we contrast cognitivist and behaviour-based approaches, for the study of the symbol grounding problem as viewed by a robot (section 12.3.2). During this discussion, we view the cognitivist approach as being concerned only with the robot's computation capabilities, which it considers to be independent of the physical support which carries out the computation (the robot's body, sensors, actuators, etc) and of the physical world in which the computation is carried out. By opposition, the behaviour-based approach gives a particular emphasis to the robot's physics (embodiment) and the robot-environment coupling (situatedness) and designs the robot's controller, such as to take advantage of these aspects.

## 3.2 The symbol grounding problem

The first issue which is tackled by the work of this thesis is that of addressing the so called *symbol grounding problem*. This notion was first introduced by Harnad [Harnad 90] to rephrase the issue raised first by Searle in his famous 'Chinese room argument'<sup>4</sup> [Searle 80], into the question of "how is symbol meaning to be grounded in something other than just more meaningless symbols?" [Harnad 90]. This position is a criticism of the 'classical' AI approach, based on the *Physical Symbol Systems Hypothesis* [Newell & Simon 76], which views the computation of an intelligent agent as the manipulation of symbols, but does not give an explicit account of how the symbols were to take meaning in the first place. By contrast, Harnad addresses specifically this problem. He argues that "cognition cannot be just symbol manipulation" and that "symbols have to be grounded into meaningful representations of the world" [Harnad 90]. That is, the symbols become *meaningful* to the agent, only once they have been grounded into the agent's physical and subjective perception of the environment.

---

<sup>4</sup> In the 'Chinese room argument', Searle argues that correct manipulation of symbols, and, thus, of concepts represented by the symbols, as demonstrated by a computer capable of speech processing, was not sufficient to prove that the computer had grasped the meaning behind the symbols. Thus, the computer could not be said to have a human equivalent understanding of its actions, namely of the sentences it produced.

In an attempt to describe the symbol grounding problem from the point of view of a robot, Harnad writes [Harnad 93]: "grounding means sensorimotor grounding: Symbols must be grounded in the capacity to discriminate and identify the objects, events and states of affairs that they stand for, from their sensory projections. A robot could perform discrimination (same-different judgements and similarity judgements, which are relative judgements on pairs of objects) using analogy processing and comparators alone (superimposing analogy projections of objects), but identification requires a mechanism for recognising categories of objects'. Further, Harnad stresses the idea that "connectionism, with its general pattern learning capability, seems to be one natural candidate [...] to find the invariant features of the sensory projection [and further] to categorise and identify objects correctly'.

Our approach to the symbol grounding problem follows directly from Harnad's. A symbol is, in our experiments, a specific sensor input, either a radio input (see chapters 7,8,9 and 10) or a keyboard switch input (see chapter 11). Each symbol is attached a meaning, by being correlated to a combination of other sensor inputs of the robot (see chapters 7 and 8) or to sequences of sensor inputs (see chapters 10 and 11). We use a connectionist architecture, which provides the robot with the necessary computation ability for spatio-temporal associations of the symbol-meaning pairs. In agreement with Deacon's suggestion [Deacon 97] that "the acquisition and use of symbols requires considerable facility for conditional associative learning, including an efficient short-term memory for sequences and combinations, and an ability to easily and rapidly produce new combinations", we develop a Dynamical Recurrent Associative Memory Architecture (DRAMA) which provides short-term memory of events, learning of temporal sequences and fast retrieval of the learned combinations by means of Hebbian mutual associativity. Recent neurobiological studies support such a Hebbian approach to account for the brain processing involved in discrimination and categorisation of words (see [Pulvermuller 98] for a good review of these studies).

The power of symbolic communication over other forms of communication lies in the fact that it allows transmission of abstract notions, i.e. of concepts. We say that an agent has a conceptual representation of an object or a situation, once it has formed an internal representation of it. In order for two agents to communicate, they must

first be able to form the same concepts and then achieve the same mapping between the symbols and the concepts to which they refer. Note that the question of whether the formation of concepts precedes that of language, which relates to the question of whether cognition, that is, the manipulation of concepts, can be separated from that of language, is the subject of an old and on-going debate among philosophers and linguists. A review of this is beyond the scope of the present manuscript, but the interested reader can refer to [Bickerton 90, Chomsky 68, Jackendoff 93, Pinker 94] for a first flavour of both sides of the argument.

Following R. Pfeifer's assumption that (taking the point of view of a robot) "all concepts must be grounded in sensory-motor coordination" [Pfeifer 98], grounding of concepts is to us the result of spatio-temporal associations across multiple sensor actuator modalities of the robot. Grounding of concepts and grounding of symbols follow thus from the same general associative process. This, again, is in agreement with Deacon's suggestion that "symbolic processes and sensory-motor attentional process may utilise the same neurological computations for different purposes" [Deacon 92]. We implement this idea, in our experiments, by using a single architecture (DRAMA) to provide learning of both symbolic and conceptual representation, based on a general spatio-temporal association. In addition, the manipulation of symbols and concepts for remembering and prediction result from the same process, namely retrieval of the associativity between sensor actuator inputs, as recorded in the DRAMA architecture. Training and retrieval algorithms are run continuously during the experiments in order to generate the symbolic associations (e.g. radio input to other sensor-actuator inputs) and other sensor-actuator associations and then to use these associations to control the robot's movements (motor activity) and communication (transmission of radio signals).

### 3.3 Social interaction and the development of language

Our discussion so far has been restricted to defining only prerequisites on our agent's cognitive skills for it to develop its understanding of a language. In this section, we define criteria for our agent's behavioural skills, which will allow a correct transmission of the language when taught by a second heterogeneous (that is functionally and

physically different) agent. Our assumptions are based on psychological studies of the development of communication skills in children. We now summarise the key ideas resulting from these studies and describe their influence on our work.

Social interactions are thought to be essential for the development of social intelligence, since it "is a fundamental vehicle for the dynamic transmission of cultural and historical knowledge" [Garton 92]. In particular, social interactions, which take the form of a coordination between the child and its mother, are thought to be fundamental precursors to the development of the child's linguistic competence: "Social communication is fundamental to the development of language and cognition, permitting the establishment of a partnership within which communication takes place" [Garton 92]. This partnership consists of being an active participant in a turn taking game. The joint activity of the mother (or other child caretaker) and the child participates in the *scaffolding* process of learning a language. Scaffolding relates to a step by step teaching process which directs the child's learning process. "This process is local, task-directed and focuses the child's attention on relevant aspect of the task" [Lock 78]. The child must first learn to communicate, that is, to correctly respond in a turn taking game. This implies understanding the specificity of speech patterns (interrogative, imperative, declarative speech) and learning to produce the correct response. The child's prelinguistic understanding of these communication patterns is described by [Bremner & Gavin 97, Bremner & Gavin 88], who observed that children develop an ability for "synchrony of movements with adult speech patterns, imitation of facial expressions and selective attention to aspects of speech" [Bremner & Gavin 88]. Further, the child develops the ability to synchronise its visual attention to that of the mother, by following the mother's direction of gaze, and to produce vocalisations in synchrony to the mother's speech.

The synchrony of activity between the child and mother results in their shared attention to specific visual and auditive patterns. The child develops his understanding of the mother's speech by associating a meaning to the mother's utterance in terms of his visual perceptions. The importance of a "co-ordination of joint activities involving mutual direction of attention" between the mother and child for the grounding of the child's understanding of language was first pointed out by Bruner

[Bruner & Watson 83] and then further expanded by diverse authors (see [Lock 78, Harley 94]). This is summarised by Harley's writing that [Harley 94] "to be effective early language learning must take place in a social setting .. [where] .. turn taking, mutual gaze and pointing are social devices .. [used for] .. establishing a joint attention [between speaker and listener] that creates a meaningful social setting necessary for the development of language'.

Following these studies, we design behavioural skills for the robot, which allows it to coordinate its behaviour with that of the teacher. Imitation, that is the action of replicating another agent's actions, results in coordinated (although delayed) activity with that of the imitated agent. Imitation is an interesting means for guiding the robot's attention, as it is an implicit means for guiding the robot's perceptions. While replicating another agent's movements, the imitator agent makes similar proprioceptions (e.g. of movement, inclination, energy consumption) and similar exteroceptions to that of the imitated agent, as both agents share simultaneously the same physical space. In the experiments reported in chapter 7 and 8, we implemented imitation skills as a following behaviour, provided by simple mutual phototaxis between the teacher/leader robot and the learner/follower robot. In the experiments of chapter 11, we built a mechanism, also based on phototaxis, for mirror imitation by a doll robot of arms and head movements of a human demonstrator. In the experiments of chapter 10, we designed environmental constraints, namely walls to form tight corridors, to constrain the perception of the robot to that relevant to the teaching (constrain to measuring only certain compass directions).

### 3.4 Our approach compared to others

This section compares our approach to related works in Computational Linguistics and Robotics. The works we refer to have already been described in chapter 2 of this thesis. In the text, we point to specific sections to which the reader can refer to have more details about these works.

### 3.4.1 Situatedness and embodiment

Recent studies of the development of communication have addressed the problem essentially from an evolutionary perspective, (1) either through theoretical models based on physiological studies of the central nervous system (e.g. [Deacon 97, Pinker 94]) or on sociological studies of primates and human societies (e.g. [Aitchison 96, Dunbar 93]), (2) or through computational studies (e.g. [Arita & Koyama 98, Kirby & Hurford 97b, MacLennan 91, Di Paulo 98, Steels 96b]), see section 2.3.1 for a more detailed description of these works. For these studies, the *symbol grounding problem* is solved once the necessary cognitive abilities have evolved. However, few of these studies considered the influence of behavioural and social factors on the development of communication, exceptions are [Dunbar 93, Di Paulo 98]. A common trend among the above mentioned simulation studies is to give a very simplified physical description of the communicative agents and their environment. The communicative agents are described only in terms of their cognitive (in contrast to behavioral) abilities that enable production and reception of the communicative signals. These cognitive functions can be simulated e.g. as the input/output of an artificial neural network [Kirby & Hurford 97b], as a matching process [Steels 96b] or as a probability function [Oliphant & Batali 97]. The agents in these simulations are disembodied, they do not occupy a physical space (they have no body, no sensors or actuators, and generally occupy not more than a single point in the space) and the result of their actions is atemporal (an action and its result occur in one time step). But most importantly, each agent has a perfect and identical perception of the environment features, based on an abstract model of the world.

By contrast, we take an approach, in which grounding of communication is a process inherently *situated* and dependent on the agent's *embodiment*, that is, it is based on the agent's individual world description which it constructs through its interactions with the environment (see section 1.1.1 for an explanation of the words in italic). The agent build its model of the world through the continuous update of the structure of its connectionist architecture, its learning process. Such a view is in direct agreement with Harnad's approach of the symbol grounding problem (section 3.2), who writes [Harnad 93]: "even at such abstract cognitive heights, [referring to] the highest level of abstraction of natural language when our interactions with objects are based only on

the interactions between names and descriptions, [...] embodiment is never escaped, for the power of names and propositions is completely parasitic on the meanings of those names, and those must all eventually be grounded in the sensorimotor interactions with the kinds of objects they designate, and the sensorimotor invariants on the basis of which the names are assigned”.

Embodiment and situatedness relate also to the fact that the robot behaviour is determined by the interaction between the robot’s own dynamic and that of the environment. Optimal behaviour results from a coupling between the agent and its environment, which requires careful design of the robot’s body and controller, in order to take advantage of the environmental dynamics (see discussion in section 1.1.1). Following this line of thought, we build our robots, their hardware and software structure, such as to take advantage of the environmental configuration. The robots are made of modular components (LEGO and FisherTechnik); it is therefore easy to transform their shape (change the height, width and form of the body), such as to make their travelling easier in the environment. We can also easily add sensors and change their disposition to provide the robot with more useful information. In particular, in several experiments we use two robots, a teacher and learner robot, which have different shapes and sensor capabilities according to their task. The hardware and software of the two robots was designed such as to provide them with phototaxis capabilities. This resulted in the imitative behaviour of the robots, which consisted of following another robot or mirroring the arm and head movements of a human demonstrator, and in their co-ordination of movements with that of the teacher, second robot or human.

### 3.4.2 Social interaction

In the computational linguistics studies we mentioned earlier [Arita & Koyama 98, Kirby & Hurford 97b, MacLennan 91, Di Paulo 98, Steels 96b], grounding of communication is regarded as a computational problem that can be solved solely by means of combinatorial analysis. For these authors, categorisation of sensor perceptions into concepts results from a process of statistical elimination among all possible meaning-object pairs, where the most likely pairs, i.e. the most frequently observed, are chosen. However, combinatorial analysis alone is not always sufficient to discard all irrelevant

information, as it is often difficult to present a sufficiently high number of relevant pair-meaning examples compared to irrelevant ones [Deacon 97]. There are also numerous situations in which one feature does not appear (naturally) without another one, e.g. the eyes, mouth, nose and other human facial features are bound to appear together with the whole face. In this case, combinatorial analysis would fail to attach two different concepts to the eyes and the face respectively, as there could be no example in which each of these features appears alone. Humans overcome this problem by using attentional mechanisms provided either by the speaker/teacher (pointing, increasing the tone of voice, linguistic deixis) and by the listener/learner (focus of gaze in the direction of the speaker's gaze or the direction pointed by the speaker's finger). Attentional mechanisms act as a cognitive process which restrict the number of observations before combinatorial analysis. However, there is more to this than just a single cognitive process. There is an interactive process between the two communicative agents, which requires a behavioural coordination between the two agents (see discussion of section 3.3). Other works have implemented such attentional mechanisms as processes distinct from the learning mechanisms, e.g. Steels and Vogt's pointing strategy [Steels & Vogt 97] and Yanco and Stein's action-selection mechanism [Yanco & Stein 93] (see section 2.3.1 for a description of these works). By contrast, we develop a single cognitive architecture which enables both associative learning, selective attention from parsing of continuous sensory information (see section 4.3.5 for a description), and the creation of a mutual binding between the two agents by means of mutual phototaxis (see sections 7.1.1 for a description of the procedure).

### 3.5 Synthesis

This chapter presented the key ideas behind our design of a control system for a robot, which could provide the robot with the necessary cognitive and behavioural skills to learn a synthetic language taught by an external agent, a human or another robot. We made a number of assumptions, which were based on studies in Linguistics and Psychology of the development of language in children.

A language is composed of symbols, that are arbitrary signals, which designate concepts. The concepts are grounded onto the agent's perceptions, and, as such, are a

subjective representation of the world, as perceived by the agent. An agent builds conceptual representations, by extracting spatio-temporal regularities in its perceptions. In order for two agents to communicate through the exchange of symbols, they must be able 1) to form the concepts carried by the symbols and 2) to learn the same mapping between the symbols and the concepts. Based on these considerations, we define a set of requirements for the learning capabilities which an agent should be provided with, in order to build conceptual representation and then map these onto a set of symbols. The agent should possess:

1. the ability for making spatio-temporal associations across multiple sensor and actuator channels,
2. a short-term memory for processing temporal associations and a long-term memory for storing the associations,
3. a segmentation preprocessing of the sensor information, which acts as an information novelty detector.

This last requirement relates to the need to reduce the amount of information on which to process the association to a relevant subset. As the task of our agent is to extract correlations between an action and a consecutive or precedent perception or between different perceptions, relevant information for our agent are changes in its sensor measurement and in its actuator states.

Cognitive functions of associativity are, however, not sufficient for learning a language. A language results from a social agreement between two or more agents to using a common set of symbols to designate a common set of concepts. Social interactions, which take the form of a behavioural coordination between the communicative agents, are necessary for the transmission of a language. The meaning of the words of the language have to be grounded onto the different conceptual representations of the communicative agents. Coordinated activity results in shared attention to similar proprio- and exteroceptions, onto which the learner agent builds its understanding of the second agent's symbolic expression.

We propose to use movement imitation to provide the necessary behavioural coordina-

tion between the teacher agent (human or robot) and the learner robot. We implement this in our experiment, as a mutual following of teacher and learner robots (see chapters 7, 8 and 10) and as mirror imitation of arm and head movements of a human demonstrator by a doll robot (see chapter 11).

## Chapter 4

# The DRAMA Architecture

We begin this chapter by an outline of the functionalities required for an autonomous mobile agent to learn dynamically. We then review different types of learning architectures which have been used in robotic experiments, pointing out the aspects of these models which prevented their implementation in our experiments and which led to the development of the DRAMA (Dynamical Recurrent Associative Memory Architecture) architecture. We then give a mathematical description of the DRAMA architecture. The last section of this chapter summarises the main properties of the architecture and discusses shortly its functional characteristics in comparison to Hebbian associative memory models and recurrent neural networks, with which it shares several common properties.

### 4.1 Associative Learning and Time Perception

In this section, we reformulate the notions presented in chapter 2 of this thesis, taking a connectionist perspective, and define criteria for the functionality of the robot's learning controller.

Robot learning is a dynamic continuous mechanism. From collecting past and present sensor information, an **ideal** robot would learn spatial and temporal regularities in its experience of varying stimuli, which it would then use to predict future perceptions and to chose its current actions. Such an ideal robot would then adapt its behaviour to act adequately in a given environment, by learning to perform adequate actions given

a particular set of perceptions. However, the appropriate response at a given point in time depends not only on the robot's current perceptions, but potentially on all its previous perceptions. Thus, learning to act adequately in a given environment means for the robot to be able to predict the consequences of its actions several time steps ahead, in order to then choose the best set of actions. To achieve this, our ideal robot should be able to learn temporal sequences of perception-action. Learning sequences of perception-action consists of 1) measuring variation of sensor-actuator information 2) correlating consecutive variations measured in different sensor and actuator systems 3) recording the temporal ordering of and time lag between each correlated variation.

In a connectionist perspective (see section 2.1.2), which we follow for the design of our model, the sensor and actuator information is represented as vectors. Learning sequences of action-perception amounts then to making spatio-temporal associations within the robot's sensor-actuator vector space. Note that, in a binary representation as used in our model, the dimension of the sensor-actuator vector space is determined by the sensitivity of the scaling chosen to segment the sensor information and the number of states for the actuators.

These general considerations lead us to require the following computational capabilities for the learning architecture of our ideal robot:

- a segmentation preprocessing of sensor-actuator information, which triggers when a change occurs in this information,
- a short-term memory for recording the new information,
- an associative mechanism which can make spatial and temporal correlations between multi-dimensional information,
- a retrieval mechanism which can retrieve sequentially the associations.

## 4.2 Review of models for spatio-temporal associations

Since the perceptron model<sup>1</sup> [McCulloch & Pitts 43], numerous connectionist models have been developed. The properties and functioning of these models differ according to the different training and retrieval algorithms they use, their internal neuronal structure and the type of inputs/outputs (real numbers or integers) they accept. We make a brief review of the main terms used in the field and attempt a classification of current models, following that of [Haykin 94] and [Hertz *et al.* 91].

### 4.2.1 Terminology and classification of neural networks

The neural network (NN) structure can be fully or partly connected. Fully connected means that all nodes of the network are connected to all other nodes. If connections are uni-directional between two units or group of units, say from input to output units and not reversely, then it is called a *feed forward neural network*. If some units are connected in both directions, as in the fully connected case (e.g. outputs units are connected back to input units [Jordan 86] or hidden units back to input units [Elman 90]), then it is called a *recurrent neural network* (RNN).

Learning in connectionist models can either be *supervised* or *unsupervised*. Supervised learning relates to learning algorithms which aim to minimise the error between the current network output and a desired output function. The desired function gives feedback on the correctness of the output. In contrast, unsupervised learning algorithms (ULAs) does not rely on matching a desired function and must discover for themselves patterns, features regularities, correlations or categories in the input data. In ULAs, the network *self-organises*, i.e. updates the parameters of its connections (weights), in order to represent regularities in the input patterns. Examples of supervised learning algorithms are the *backpropagation algorithm* [Rumelhart *et al.* 86, Werbos 74] and its derivatives (see [Chauvin & Rumelhart 95, Pearlmutter 95] for a review). The backpropagation algorithm is based on an optimisation technique, called gradient descent, which minimises a *cost function*, which gives a measure of the error between the network's current and desired outputs. There exist a variety of such algorithms, which

---

<sup>1</sup> This model developed in 1943 is often presented in the literature as the starting point of the connectionist research field.

differ by the cost function and the gradient method they use (see the above mentioned survey for a review).

Example of learning algorithms are *Hebbian learning* and *Competitive learning*. Hebbian learning updates the network connection weights following the Hebbian update rule. The Hebbian rule originated from Hebb's proposal for the reinforcement of biological neural connections: "When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic changes take place in one or both cells such that A's efficiency as one of the cells firing B, is increased." [Hebb 88]. In some connectionist approaches, this statement is rephrased in a two-part rule [Haykin 94]: 1) if two neurons on either side of a synapse (connection) are activated simultaneously, then the strength of that synapse is selectively increased; 2) if two neurons on either side of a synapse are activated asynchronously, then that synapse is selectively weakened or eliminated. Hebbian learning can be used as part of unsupervised learning strategy as in Hebbian networks [Fyfe 95]. It can also be used for supervised learning in which the network is given explicitly the correct output, as it is the case in the Hopfield network [Hertz *et al.* 91].

Competitive learning is used in the learning stages of the ART and Kohonen networks. The principle is based on competition for activation among the output units; only the unit(s) which receive the biggest vote (above some threshold) for activation from other units is(are) activated. It is a winner(s)-take-all algorithm<sup>2</sup>. Competition for activation of output units is based on the connection weights from the input units to that output unit. In several networks, the winning units are those for which the sum (over all input units) of the connection weight multiplied by the input unit value is greater than a fixed threshold. In the ART network only one unit can be active at a time. Once the winning unit is determined, the connection weights between this unit and its input units are incremented or decremented, depending on whether the input unit is active or not. In the Kohonen net [Kohonen 89] and its derivatives [Kaski *et al.* 98], the winning unit is the unit for which the distance between the input vector and its weight vector is the smallest. The weights of the winning unit and other

---

<sup>2</sup> The winners-take-all algorithm (note the plural on winner) is the case when more than one output unit can be activated. The singular, winner-take-all algorithm, is the case when only one unit is activated.

output units in a close neighbourhood are updated, so as to minimise this distance.

Recurrent neural models use either the backpropagation algorithm (using its generalisation to RNNs [Pineda 87, Williams & Zipser 89] and its derivatives [Pearlmutter 95]) or Hebbian rules, e.g. the Hopfield net and the Boltzmann machine [Hinton & Sejnowski 86] and the DRAMA network which we present in this thesis, or else Evolutionary algorithms. Feed forward NNs use mostly either the backpropagation algorithm and derivatives or Hebbian rules, e.g. Willshaw, Kohonen and ART networks.

ANNs can be divided in at least five categories, which correspond to the type of computation they can perform: 1) Associative memory models, which store pairs of input-output patterns (e.g. Hebbian networks, such as the Willshaw network [Willshaw *et al.* 69] and other Hebbian type of network [Fyfe 95, Palm *et al.* 97, Sommer & Palm 98], some feed forward networks [Kolen & Pollack 91]). 2) Autoassociative memory models (e.g. Hopfield network and Boltzmann machine), which update their internal structure so as to represent an energy function of which the input pattern are minima<sup>3</sup>. 3) Models for pattern classification; these models can be made of feed forward NN with backpropagation or Hebbian NNs, e.g. ART network, which perform a linear segmentation of the input pattern space and classify patterns depending on their feature relative to the space decomposition; some models cluster the input space in a manner similar to Principal Component Analysis (PCA) (e.g. [Oja 82, Fyfe & Baddeley 95]) and Independent Component Analysis (ICA) (e.g. [Oja *et al.* 95, Hyvarinen & Oja 98, Girolami & Fyfe 97]). 4) Topographical maps, which classify the input patterns by drawing a one or two dimensional discrete map where each region of the map represents inputs with similar features (e.g. Kohonen nets [Kohonen 89] and other variations, see [Kaski *et al.* 98] for a review), 5) Models for time series prediction based on spatio-temporal association of input-output patterns (e.g. feed forward NN with time delay [Day & Davenport 93, Lin *et al.* 92], recurrent neural networks with backpropagation algorithm [Chauvin & Rumelhart 95, Pearlmutter 95]).

A complete review of all these models is beyond the scope of the present thesis (a good overview can be found, e.g., in [Arbib 95]). In the rest of this section, we give a brief review of models which show properties relevant to the problem tackled by

---

<sup>3</sup> Note that autoassociative networks are sometimes considered as associative memory in the literature.

this thesis, while pointing out these models' characteristics which prevented us from applying them in our robotic experiments.

The requirements of section 4.1, asking that the learning architecture shows associative properties for learning spatial and temporal invariance, lead us to consider mainly two types of connectionist models for solving our problem: the Hebbian associative memory models, for their properties of associativity, and the recurrent neural networks, for their ability to learn time series. In the following, we review the characteristics of the existing models with regard to these categories.

### 4.2.2 Associative memory

Associative memory models such as Hebbian networks present several relevant properties for our problem such as fast computation, because they use a one-time-step training algorithm (Hebbian rule), associativity of input patterns with different dimensionality (when using a binary encoding, the information coming from each sensor and actuator of the robot is represented by a different number of input units, depending on the sensor sensitivity and the number of actuator modes), bidirectionality of the association, which allows retrieval of the association from sensor input to actuator output (control of the robot's actions) and vice-versa (prediction of the actuator-sensor sequence).

A disadvantage of current associative models, however, lies in the fact that they do not record the temporal aspect of the correlated occurrence of a pair of input pattern, that is the time delay between successive occurrences of the two input patterns. In our problem, namely learning temporal sequences of the robot's perception-action, recording the time delay between each pattern occurrence in the sequence is important as it provides information on the robot's internal dynamic (delay of reaction given a perception) as well as on the environmental dynamic (delay between a change of perception after an action and delay between perception of different stimuli). Associative memory models which can learn sequences of patterns do exist (e.g. [Hattori & Hagiwara 96], [Kolen & Pollack 91], [Rinkus 95], [Schwenker *et al.* 96]). However, in these models, the time delay between each pattern occurrence is fixed and is equal to one processing cycle (it has therefore no relationship with the real time of pattern occurrence). That

is, the patterns of the series are presented sequentially to the net, without delay.

In section 4.3.1, we describe in details the functioning of one particular Hebbian network, namely the Willshaw network, whose functioning is based on Hebbian learning and a winners-take-all retrieval mechanism<sup>4</sup>. The DRAMA functioning is based on similar training and retrieval algorithms. For a more detailed review of other type of associative memory mentioned above, the reader can refer to, e.g., [Anderson 95, Hinton & Anderson 89, Mozer 93, Palm *et al.* 97].

### 4.2.3 Recurrent Neural Networks

In this section, we use the term recurrent neural network (RNN) by extension to refer only to RNNs trained with the backpropagation algorithm or its derivatives and which can learn time series. This leaves out the Hopfield network, which has a recurrent structure, but functions as an autoassociative memory.

Recurrent neural network (RNN) models encapsulate implicitly the temporal ordering of pattern occurrence in their recurrent structure. For this reason, they are good predictors of time series (see [Chauvin & Rumelhart 95, Pearlmutter 95] for a review of these models). RNNs are distinguished following two main criteria: 1) the RNN can use either a discrete or a continuous time scale; the later corresponds to RNN in which the neuronal activation function is described by differential equations; this type of network is called *real time recurrent neural network*. 2) the RNN can use a fixed or variable time delay between two time steps (this concerns only discrete RNNs); in the later case, the network is called an (*adaptive*)<sup>5</sup> *time delay recurrent neural network*. In a time-delay RNN, the activity of one neuron is propagated to other neurons of the network only after a time delay equal to the value of the neuron's *time parameter*. By updating the value of this time parameter together with the update of the connection weights, one can train the network to respond correctly to a sequence with variable time-delay of input occurrence. The DRAMA network uses a

---

<sup>4</sup> Note that the original version of the Willshaw network [Willshaw *et al.* 69] was not based on a winner-take-all mechanism. We describe the latest version as investigated by [Graham & Willshaw 96].

<sup>5</sup> The term *adaptive* is sometimes omitted in the literature; in this thesis, we refer simply to time-delay RNN for variable time-delay RNNs.

time parameter, which is similar in its function as that of time delay neural networks [Day & Davenport 93, Lin *et al.* 92, Lin *et al.* 93, Beer 95], see page 68 and equation 4.2 for a description of this parameter's uses in the neural activation function. However, in contrast to the above mentioned models, the time parameter in DRAMA is updated following a pseudo-Hebbian rule instead of the backpropagation algorithm. That is, the time parameters are updated whenever the two units linked by the connection are co-activated. However, the update does not consist of increasing the parameters' values as in the original Hebb's rule (see page 54), but rather the update leads to the calculation of the mean value of time delay between two units co-activation.

Learning algorithms for RNN models have been developed only in the framework of supervised learning. The exception is the Hopfield network, which uses Hebbian learning<sup>6</sup>. Currently, one uses either the generalised backpropagation algorithm ([Chauvin & Rumelhart 95], [Pineda 87], [Pearlmutter 95], [Williams & Zipser 89]), or Genetic Algorithms ([Beer & Gallagher 92], [Angeline *et al.* 94], [Ijspeert *et al.* 98], [Kodjabachian & Meyer 98]) to calculate the connection parameters. These algorithms were not suitable for our problem, principally because they require heavy computation (several steps of computation for training and retrieval algorithms), which can not be carried out on-line, especially not with a computationally limited hardware system such as the one we used, see section 6.1.1. Another reason for not using these algorithms is that they rely on an predefined evaluation function for the network output (supervised learning), which would have been very difficult to design as we did not know in advance all the regularities the robot might perceive during its autonomous random wandering. However, if we had been provided with a system with sufficient computational power (note that most current "affordable" robotic system do not have such power), we could imagine training the network on-line, taking the current state of the robot's sensors and actuators as the desired output, i.e. learning to predict sequences of sensor-actuator states. However, it is yet to be demonstrated that RNNs would be sufficiently robust (as real data are very noisy) and would generalise correctly from a small amount of data (real data are slow to be collected and it is difficult to get a significant amount

---

<sup>6</sup> The Hopfield net was not suitable for our problem as it does not perform learning of time series and because the training and rehearsal algorithms are very slow, based on several time steps of computation.

of them in a reasonable time), while it is known that the performance of RNNs can easily degrade, for example by forgetting previous learning, when presented with new training samples [Pearlmutter 95, Chauvin & Rumelhart 95].

Given that none of the current artificial neural networks (ANN) models could entirely satisfy our requirements for the learning architecture of our robot, we developed a connectionist model which is a combination of a Hebbian associative memory model and a recurrent neural network. It presents the same advantageous properties relative to the particular implementation of these models while discarding the disadvantageous ones which we mentioned in the previous sections. In the following section, we give the mathematical description of the model.

### 4.3 Structure of the DRAMA architecture

The DRAMA architecture functions as a recurrent associative memory. It was derived from the model of associative memory proposed by Willshaw [Willshaw *et al.* 69]. Its development was driven by our wish to build a control architecture to enable *real time* control and learning in a physical autonomous agent. In particular, the choice of using a neural network and especially a Hebbian associative memory was driven by considerations pertaining to its implementation on a real robot with limited computational power. We require 1) fast computation for the system to react in real-time 2) robustness and adaptability in the face of varying environmental constraints<sup>7</sup>, 3) as little built-in knowledge as possible to keep the system unspecific to a particular type of implementation (task, agent or environment).

#### 4.3.1 The Willshaw net

The original version of the Willshaw net [Willshaw *et al.* 69] was developed as a model of biological associative memory. It can be thought of as a fully-connected network with symmetrical connections, whose weights are updated following a basic (or '*clipped*') [Graham & Willshaw 97]) Hebbian rule, i.e. only the weights of connections with co-

---

<sup>7</sup> The variation of the environmental constraints are, in our experiments, e.g., changes in spatial distribution of objects, variation of lighting and electro magnetic field, and changes in the timing of sequence teaching when the human instructor gets tired!

active nodes are reinforced. The patterns consist of pairs of input-output bit-strings. The lengths of input and output strings do not need to be equal, thus patterns of different dimensions can be associated.

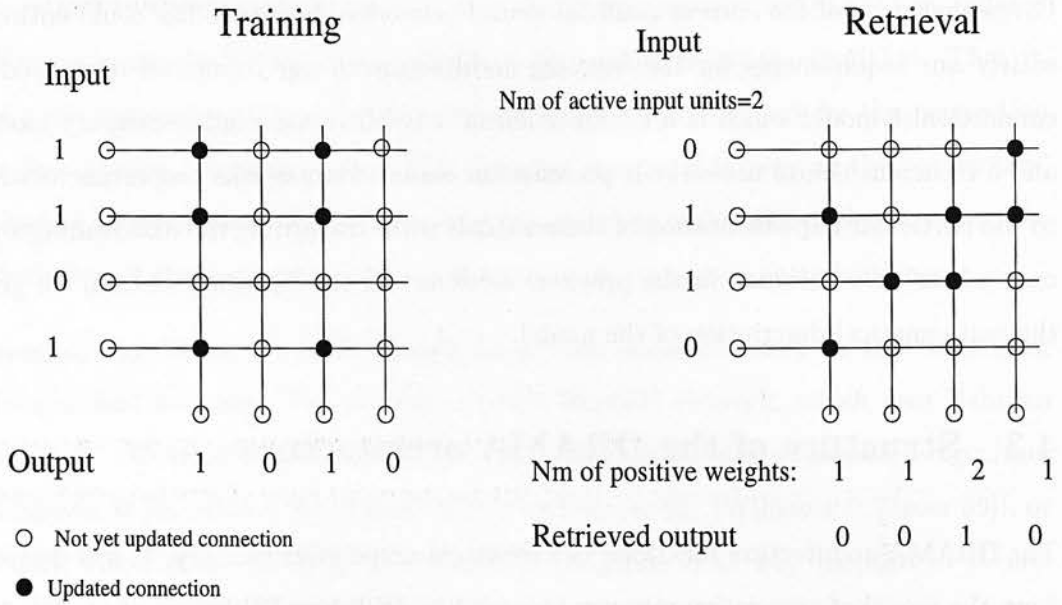


Figure 4.1: Schema of the Willshaw network

The patterns are presented as arrays of binary (0/1) inputs. The learning stage begins with all the weights equal to zero. When an input-output pair is presented, for a binary-encoded input, the connection weight or intersection node between two activated units, i.e. one input and one output node which are both 1, is updated to 1. Whenever a weight has been updated to 1, it will never return to zero. The recall of a memorised pattern is done by summing the positive connection weights between each active unit and each output unit. An input pattern is presented to the net. For each output column, the number of positive connection weights for each active input line are summed. The output nodes which have a number of positive connection weights greater than or equal to the number of active inputs are activated. Figure 4.1 shows two examples of training and retrieval of the Willshaw net when presented with 4 bits input-output patterns.

The Willshaw net was developed as a neurobiological model of memory [Willshaw *et al.* 69, Graham & Willshaw 95]. In contrast, our concern is to define an

artificial architecture of associative memory for robotic applications without necessarily being constrained by biological plausibility. Robustness is an important criterion and we therefore defined, in [Billard 96] and in this thesis, an update rule for the connection parameter which, similarly to the concept of unit usage [Graham & Willshaw 95], keeps an exact record of the connection usage; that is, the frequency of correlated activation of any two units is kept. We call this parameter a *confidence factor*. The network functioning when using such a confidence factor as the network's weight parameter bears similarities to other correlation-based associative memory, such as the Kohonen associative memory [Kohonen 89], Hopfield network [Hopfield & Tank 86] and some Hebbian networks [Fyfe 95]. The model in [Billard 96] was similar to Willshaw's because it kept the basic principle of training and the retrieving algorithm of the original model. This resulted in a statistical type of network whose functioning was a mixture of the classical Hebbian network and the Willshaw network. For a full discussion of its functioning and implementation, the reader may refer to [Billard 96].

The extension we describe in the following adds recurrent connections to each of the nodes of the network, in order to make correlations between delayed and simultaneous occurrences of different input patterns. The uniform structure of the original network is changed for a fully recurrent, non symmetrical network, whose connections are associated with two weight parameters, recording separately the spatial and temporal features of the training patterns. A one-time-step algorithm is used for updating the parameters, using a Hebbian rule, and a threshold-based algorithm is used for retrieval of the unit activity. The resulting model is a simple version of a recurrent NN (as compared with a RNN using back-propagation and with hidden layers) that satisfies our basic requirements, namely fast computation for real time functioning and temporal associative learning capabilities. Note that the model we present here has few similarities with the Willshaw network. Only the retrieval rule which is based on a threshold strategy where the thresholds act both on the connection weight strength and on the number of active input units is to some extent similar to that used in the Willshaw net (they are however different as the connection parameters have not the same function or use in the two models). The structure (in DRAMA there are asymmetrical connection parameters and self-connections on the units, while the Willshaw net has symmetrical connection parameters and no self-connections) and the training rules of

the two networks are however different. We presented the Willshaw network's functioning because the development of the DRAMA architecture was influenced by our study of the Willshaw model. In chapter 12, we discuss in more detail the differences between our present model and other Hebbian and RNN models.

The rest of this section presents the complete extended version we have developed from the original Willshaw model. We first give a brief overview of the model functioning for controlling learning and behaviour of robotic agents (in section 6.2, we give a detailed description of the implementation of the architecture for the experiments we report in the thesis). We then give the mathematical equations for the functioning of the network.

### 4.3.2 A control architecture for autonomous robot

The DRAMA architecture provides a general control architecture for autonomous agents. Because we implement the DRAMA architecture in robotics experiments, we will use the terms "actuators" or "sensor systems" to describe the different input patterns to the network. In the experiments, the robots' actuators are, e.g., the motors and a radio emitter. The robots' sensor systems are both proprioceptive (e.g. inclination sensor, an energy level checker, a compass) and exteroceptive (light and infra-red detectors, whiskers and bumpers). Note that, by extension, we use the term "sensor system" for both sensor and actuator systems when differentiating between them is unnecessary, i.e. when the processing of their information is independent of the type of system that provided it.

Figure 4.2 shows a schematic representation of the model with three sensor systems as inputs. The structure of the system is composed of two parts: a data preprocessing module for *event recognition* and the DRAMA architecture. At each processing cycle, the sensor-actuator vector state is measured and its information passed through the event detector modules associated with each sensory system. Sensor-actuator inputs are presented as arrays of binary data 0/1 (bit-strings or vectors) of different length for each system. Information from each sensor system is treated separately by each event detector module and, thus, an event is determined differently for each system. Each sensor is represented as a box with  $n$  input units, where the number of units associated

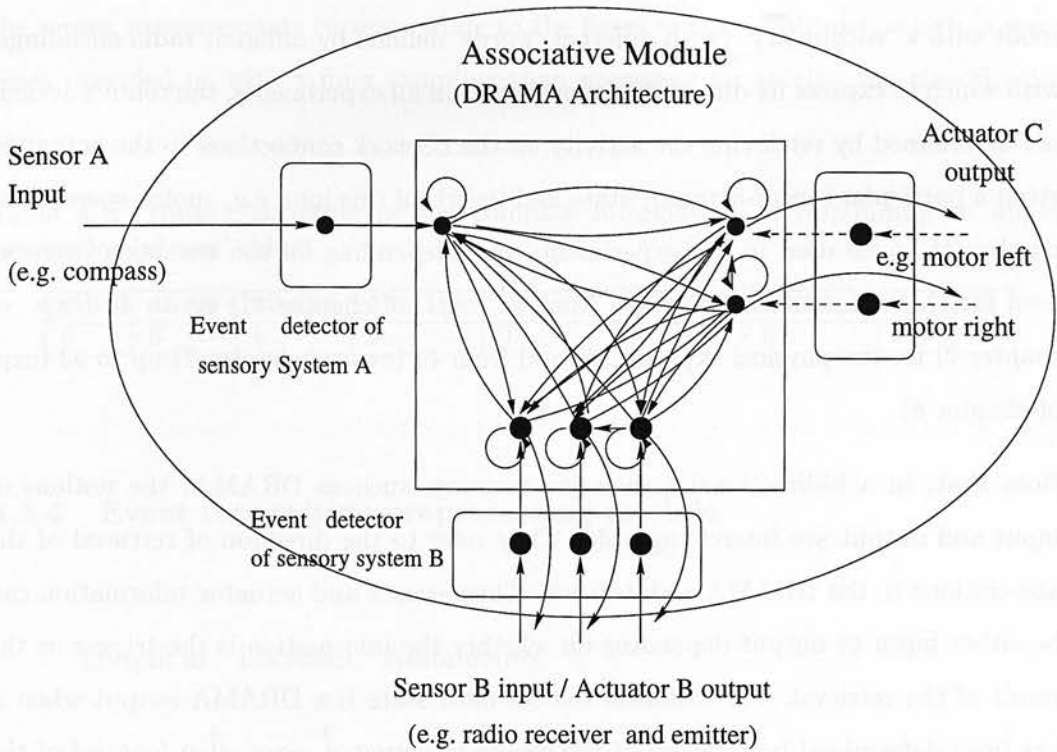


Figure 4.2: Schema of the robot's control system, using the DRAMA architecture

with a sensor can vary from one sensor to another. When a variation in one sensor or actuator input has been measured (see section 4.3.4), the new information is forwarded to the associative architecture (DRAMA) to be correlated with all simultaneous and previously recorded events in other sensor-actuator systems. DRAMA is a fully connected recurrent neural network, without hidden units. The self-connections on the units provide a short term memory of the units' activation, and consequently of a sensory event. Long term memory of two events consecutive activation is obtained by updating the internal connections following Hebbian rules (see page 73). Sensory systems A, B and C in figure 4.2 could be interpreted, e.g., as the motor, compass and radio systems of the robots, as in experiments of chapter 7. Note that in the experiments there was no input to the motors as there was no external mechanism for changing their activity, hence the dotted arrows in the figure. Only in experiments of chapter 11, the robot's motor activity could be changed, a result of the robot's moving one arm or the head, while mirroring the movements of the human demonstrator. In experiments of chapter 7, the radio signals are associated with different compass states, thus providing the

robot with a ‘vocabulary’ (with different ‘words’ defined by different radio encodings) with which to express its direction of movement. In all experiments, the robot’s actions are determined by retrieving the activity on the network connections to the actuators given a particular sensor-actuator state and inverting this into, e.g. motor speed. The number of nodes used in the experiments vary depending on the number of sensors used for the computation: it varies from 17 (exp. of chapter 11) up to 43 (exp. of chapter 7) for the physical experiments and from 42 (exp. of chapter 7) up to 93 (exp. of chapter 8).

Note that, in a bidirectional associative memory, such as DRAMA, the notions of input and output are interchangeable. They refer to the direction of retrieval of the associations in the DRAMA architecture. Thus, sensor and actuator information can be either input or output depending on whether the information is the trigger or the result of the retrieval. For instance, the actuator state is a DRAMA output when it has been determined by retrieval of the sensor to actuator association (control of the robot’s actions) and a DRAMA input when its information is used to calculate the sensor state (prediction of the robot’s perceptions).

### 4.3.3 Data encoding

We use a binary encoding, the state of each sensor and actuator being encoded in one bit-string of different length for each sensory system, depending on the sensor sensitivity and the number of actuator modes. As will be discussed in chapter 5, the model capacity decreases strongly when the pattern encodings overlap. Thus, in the experiments, we tried to encode all sensory information as orthogonal patterns, when this was possible. For example, information provided by the compass was encoded in a bit-string of length 8, where each bit would correspond to one of the 8 quadrants (see figure 4.1). Bit 1 corresponds to an angle of 0 and 45 degrees, bit 2 to angle between 45 and 90, etc. Thus, each compass measurement would be represented by a pattern with one single bit activated. Such a representation of the data serves as a first classification of the sensory information into subclasses. The fineness of the data segmentation is task dependent. In our experiments (see chapters 7 to 11), we chose a segmentation of

the sensor measurements corresponding to the finest sensor sensitivity, which in some cases provided us with a finer sampling than necessary for solving the classification task.

Table 4.1: Binary encoding of the compass information corresponding to angles between ‘min-max’ values.

0-45	45-90	90-135	135-180	180-225	225-270	270-315	315-360
0	0	1	0	0	0	0	0

#### 4.3.4 Event recognition preprocessing module

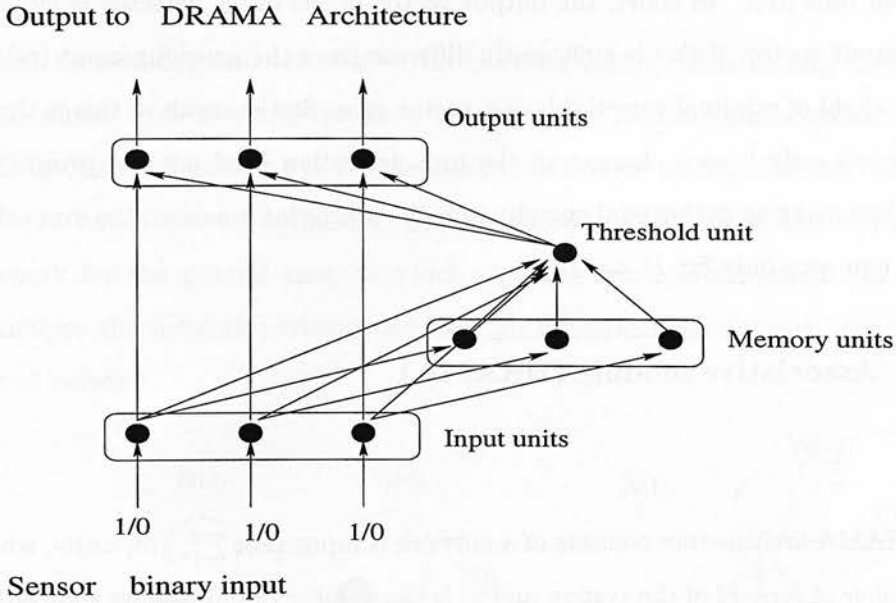


Figure 4.3: Event detector module

There is one *event detector* module per sensor. Each module receives  $n$  input units and outputs to  $n$  associated units in the DRAMA architecture, where  $n$  is the number of units of the particular sensor (see figure 4.3). The neuronal representation of the internal structure of the module is given in figure 4.3. Each input unit is connected to one *memory* unit, one *output* unit and to a *threshold* unit. Output  $y_i^n(t)$  of the memory unit  $i$  at time  $t$  is simply the value  $x_i$  of the input unit  $i$  at time  $t - 1$ ,

$y_i^m(t) = x_i(t - 1)$ . Output  $y_i^{th}$  of the threshold unit is the result of the function  $\theta(x, H)$  applied onto the difference between the input units and memory units outputs:  $y_i^{th}(t) = \theta(\sum_{i=1}^n |x_i(t) - y_i^m(t)|, H)$ , where the function  $\theta(x, H)$  is a threshold function that outputs 1 when  $x \geq H$ . Finally, the state of the output unit  $y_i(t)$  is calculated as follows:

$$y_i = \theta(x_i(t) + y_i^{th}(t), 2) = \theta(x_i(t) + \theta(\sum_{i=1}^n |x_i(t) - x_i(t - 1)|, H), 2) \quad (4.1)$$

The threshold  $H$  fixes the minimal number of unit inversions in the input before activation. For example, if  $H = 1$ , the threshold unit fires as soon as one input unit has changed from 0 to 1 and the output unit outputs 1 if it receives an input value  $x_i(t)$  equal to 1, otherwise 0. The output vector is then equal to the input vector, once the threshold unit fires. In short, the output vector of the event detector is either equal to the input vector, if this is sufficiently different from the previous input (relative to the threshold of minimal variation), or a vector zero. So the result of this is that event units detect only  $0 \rightarrow 1$  changes in the unit activation (and not the reverse). Note that, when using an orthogonal encoding for the sensor information, the event detector will be non zero only for  $H \leq 1$ .

#### 4.3.5 Associative module (DRAMA)

The DRAMA architecture consists of a network composed of  $\sum_{i=1}^n m_i$  units, where  $n$  is the number of sensors of the system and  $m$  is the number of input units associated with each sensor, different for each sensor. It is a fully connected recurrent network with non symmetrical connections, i.e. each unit is connected with all other units in the network and with itself (self-recurrent connections). Each unit also receives input from one output connection of the event recognition module. There are no hidden units. Note that, in the robotic experiments, the network is fully connected at the level of the sensory systems; that is, all units  $i$  in sensor  $k$  are connected with all units  $j$  in sensor  $l$  ( $l \neq k$ ); however, units inside the same sensory system are not interconnected (see figure 4.4). This was done in order to save the computation cost when running the system on-line, by reducing the size of the weight matrices and the number of

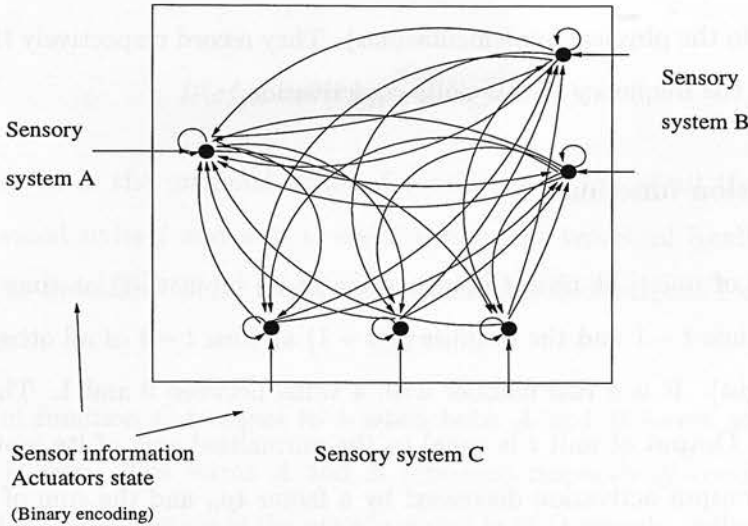


Figure 4.4: The DRAMA architecture.

operations for training and retrieval of the network (connections among units inside the same sensor system would not have improved the learning performance in these experiments, as most patterns inside the same system were orthogonal and thus inside connections would not be updated). In the following, we present the equations of the network for the general case, in which all units are interconnected. In chapter 5, we analyse the network performance through numerical simulations, using a fully connected network.

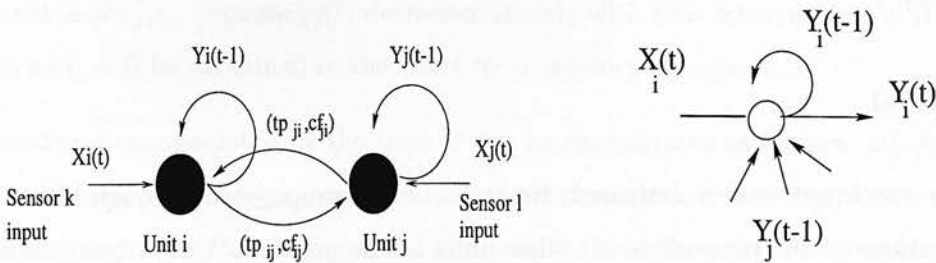


Figure 4.5: **Left:** Bidirectional connectivity of two network units. **Right:** One unit connectivity.

Similarly to time-delay neural networks [Day & Davenport 93, Lin *et al.* 92], each connection in the DRAMA network has two parameters associated with it instead of one: a time parameter ( $tp$ ) and a confidence factor ( $cf$ ) (see figure 4.5 left). Time parameters and confidence factors are positive numbers (real numbers in the simulation

and integers in the physical implementations). They record respectively the time delay between and the frequency of two units co-activation.

### Unit activation function

Output  $y_i(t)$  of unit  $i$  at time  $t$  is a function of its input  $x_i(t)$  at time  $t$ , its output  $y_i(t - 1)$  at time  $t - 1$  and the outputs  $y_j(t - 1)$  at time  $t - 1$  of all other units  $j$  (see figure 4.5 right). It is a real number with a value between 0 and 1. The equation is given in 4.2. Output of unit  $i$  is equal to the normalised sum of its input activation, its previous output activation decreased by a factor  $tp_{ii}$  and the sum of activation of other winning units, that is units which have passed the conditions encapsulated by the function  $G$ .

$$y_i(t) = F(x_i(t) + tp_{ii} \cdot y_i(t - 1) + \sum_{j \neq i} G(tp_{ji}, cf_{ji}, y_j(t - 1))) \quad (4.2)$$

where  $F$ , the transfer function, is the identity function for input value less than 1 and saturates to 1 for value greater than 1,  $F(x) = x$  if  $x \leq 1$ , otherwise  $F(x) = 1$ , and  $G$  is the retrieving function whose equation is given below in equation 4.3 and explained in the following paragraph. The indices notation used in the equations should be interpreted as follows:  $cf_{ji}$  is the confidence factor of the connection leading from unit  $j$  to unit  $i$ .

### Retrieval

When one input unit is activated, its activation is propagated through the internal connections of the network to all other units of the network. Unit  $i$  becomes active, i.e.  $y_i = 1$  under the effect of activation of unit  $j$  if function  $G$  applied on the output of unit  $j$  in equation (4.2) has value 1. The retrieving function  $G$  depends on the value of the connections parameters  $tp_{ji}$  and  $cf_{ji}$  and the output  $y_j$  of unit  $j$  and is defined as follows:

$$G(tp_{ji}, cf_{ji}, y_j(t - 1)) = A(tp_{ji}) \cdot B(cf_{ji}) \quad (4.3)$$

$$A(tp_{ji}) = 1 - \theta(|y_j(t - 1) - tp_{ji}|, e)$$

$$B(cf_{ji}) = \theta(cf_{ji}, \frac{\max_{y_j > 0}(cf_{ji})}{T})$$

where  $\max_k(cf_{ji})$  is the maximal value of confidence factor of all the connections between activated units  $j$  and unit  $i$ , which satisfy the temporal condition encoded in  $A(tp_{ji})$ . The function  $\theta(x, H)$  is a threshold function that outputs 1 when  $x \geq H$  and otherwise 0.

The output of function  $G$  is equal to 1 when both  $A$  and  $B$  terms are equal to 1, otherwise it is zero. The terms  $A$  and  $B$  represent respectively conditions on the temporal and spatial structure of the units' activity in the network. Following equation 4.2, the unit  $i$  can be activated by other units  $j$  only when both conditions of function  $G$  are satisfied; that is when  $A$  and  $B$  are equal to 1.

The condition encapsulated by the term  $A$  can be paraphrased as follows:  $A(tp_{ji}) = 1$  if the time delay for which the activation of unit  $j$  has been memorised before being correlated to the activation of unit  $i$  is equal to the time encoded in the time parameter  $tp_{ji}$  within an interval error  $e$ . In other words, a unit  $i$  can be activated by a unit  $j$  only within a short time window around the recorded time delay between these two units co-activation. This is illustrated in the example of figure 4.7. See explanation in caption. The time delay since which a unit  $j$  has been activated is indirectly encoded in the value of  $y_j(t)$ , because  $y_j(t)$  decreases linearly with time when no new activation occurs, as it will be explained in the *short term memory* paragraph.

The condition encapsulated by the term  $B$  can be paraphrased as follows:  $B(cf_{ji}) = 1$  if the confidence factors  $cf_{ji}$  associated with the connection between one activated unit  $j$  and unit  $i$  is greater than or equal to  $1/T$  times the maximum of confidence factor for other activated connections,  $\max_{y_j > 0}(cf_{ji})$ . In other words, a unit  $i$  can be activated by a unit  $j$  iff the connection strength (the value of the confidence factor) between these two units is greater than a threshold. The threshold is a fixed percentage of the maximal value of confidence factor for all connections from active units  $j$  to unit  $i$ . Figure 4.6 illustrates this threshold-based retrieval mechanism with two examples. See caption for explanations.

The conditions encapsulated by the terms  $A$  and  $B$  are illustrated by figure 4.7 which

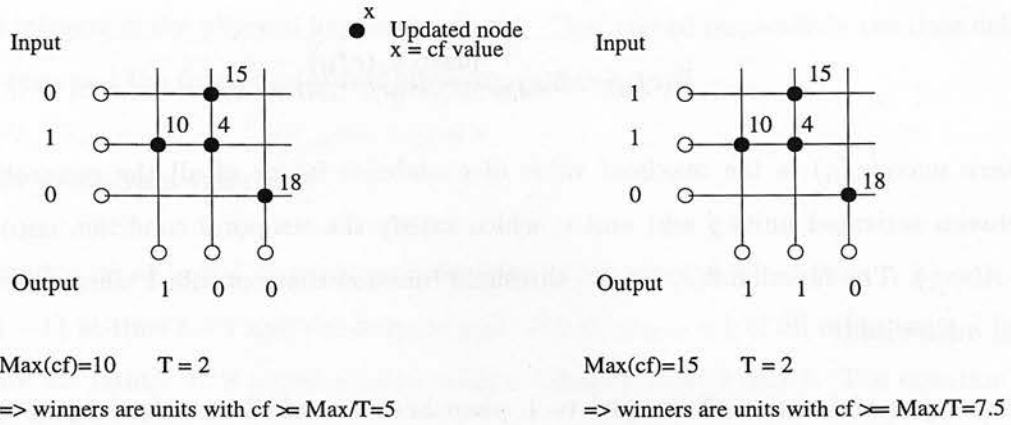


Figure 4.6: **Threshold-based retrieval mechanism.** An output unit is activated iff the confidence factor (cf) value of one of its connections with an active input unit is greater or equal to  $\text{Max}/T$ . Max is equal to the maximal value of confidence factor for all connections from all active input units. In the example on the left, Max is equal to 10 which is the maximal cf value for the second input line (the only active line). In the example on the right, Max is equal to 15 which is the maximal cf value for the first and second input lines (the only active lines). The threshold  $T$  has a fixed value of 2.

shows a schematic representation of the propagation of unit  $j$  activity along the network connection. The unit  $j$  activity passes first the filter on time (represented by the factor  $A$ ); that is its activity is propagated further through the connection once a time delay  $tp_{ji}$  has passed since unit  $j$  activation time. Unit  $j$  activity is then passed through the threshold-based retrieval mechanism (represented by the term  $B$ ); the threshold acts on the value of the connection's confidence factor. Unit  $j$  eventually activates unit  $i$ , iff it passes the two conditions  $A$  and  $B$ .

The effect of the two terms  $A(tp_{ji})$  and  $B(cf_{ji})$ , and in particular of the threshold  $T$  and  $e$ , on the memory capacity will be discussed further in pages 85, 91 and an algorithm for calculating the parameters  $T$  and  $e$  on-line will be presented in this section. Note that in experiments of chapter 11 on sequence learning, a third condition for the output unit activation was introduced, which required that the number of active units in the network was equal to that for which the  $G$  function output 1. The effect of this condition of the network ability for learning time series is discussed in section 11.3.2.

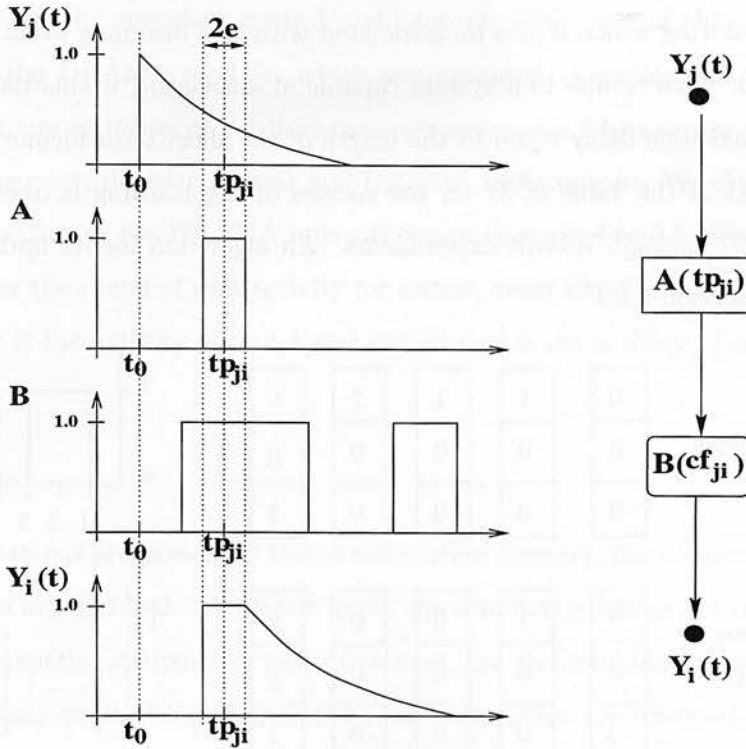


Figure 4.7: **Propagation of the unit activity along the network connections:** At time  $t_0$ , unit  $j$  fires. Its activity satisfies the condition on time delay, encapsulated by the term  $A$  of equation 4.3 (i.e.  $A$  outputs 1), during a time window ( $tp_{ji} \pm e$ ) around the recorded time delay  $tp_{ji}$  between these two units' co-activation. If it satisfies the threshold condition on confidence factor  $cf$  encapsulated by the term  $B$  of equation 4.3 (i.e.  $B$  outputs 1) during this time window, it activates unit  $i$ .  $B$  outputs 1 if the activity of all units  $j$  is such that it satisfies the threshold criteria of  $B$ , see explanations of figure 4.6. In the figure the values of  $B$  are set to 1 or 0 for the purpose of the example.

### Short Term memory

The self-connections on the units of the network provide a short-term memory of each unit activation. If unit  $i$  receives no external activation from its input ( $x_i = 0$ ) or other units' outputs ( $y_{j \neq i} = 0$ ), then its output activity is equal to  $y_i(t) = tp_{ii} \cdot y_i(t - 1)$ , that is it decreases by a ratio proportional to its temporal parameter  $tp_{ii}$ . Its value returns to 0 when the maximal decimal capacity of the system has been reached or, before that, if a limit of number of processing cycles has been set for keeping a record of the unit activity (that is fixing the duration of short-term memory). Therefore, once information from a sensor or actuator has triggered the event detector, it is then further memorised for a period  $M$  (fixed by the decrease of activation along the self-recurrent

connections) during which it can be associated with any incoming event in any other sensor system. This results in a system capable of associating events delayed in time with a maximal time delay equal to the length of the short-term memory (STM), i.e.  $M$ . The effect of the value of  $M$  on the success of the learning is discussed further in section 7.2.2 through robotic experiments. An algorithm for its update on-line is evaluated in section 5.2.1.

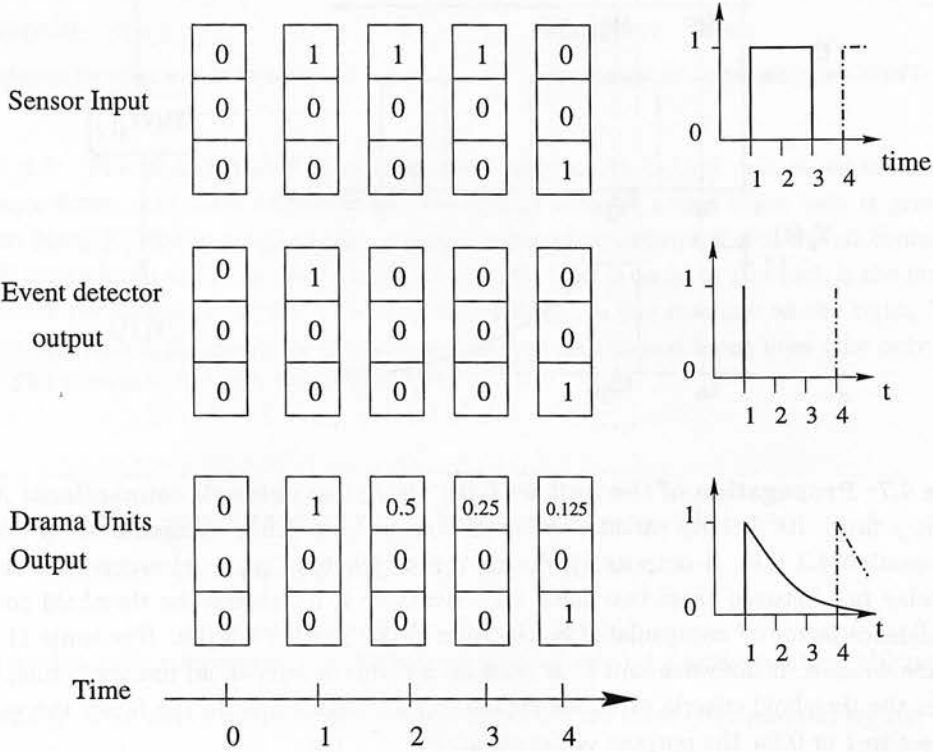


Figure 4.8: Transcription of sensor input into event detector output and DRAMA units output.

In summary, the output  $y_i$  of a unit  $i$  in the network takes values between 0 and 1:  $y_i(t) = 1$  when (i) an event has just been detected ( $x_i(t) = 1$ ) or (ii) when the sum of activation provided by the other units is sufficient to pass the two thresholds of time and confidence factor, represented by the  $G$  function. A value inferior to 1 represents the memory of a past full activation (value 1). For example,  $y_i(t) = (tp_{ii})^{t-t'} \cdot y_i(t-t') = 0.9^3 \cdot 1 = 0.7290$  means that unit  $i$  has been activated 3 time steps ago, when the decrease rate of the activation along the recurrent connections is equal to 0.9. Note that, in the experiments, all  $tp_{ii}$  are set equal to the same value, thus providing the same memory duration for all nodes.

Coming back to the complete control architecture, composed of the event detector module and the DRAMA module, which we presented in section 4.3.4, we show in figure 4.8 the transcription of a 4-time steps sequence of a 3-bits sensor input into the corresponding event detector output and DRAMA units output. We choose a ratio of decrease activation in the DRAMA unit self connection equal to 0.5. The diagrams on the right show the shape of unit activity for sensor, event detector and DRAMA units (straight line is the activity of unit 1 and dotted line is the activity of unit 3).

## Training

When two patterns are presented to the associative memory, the connections between coactive units are updated; on the one hand, the confidence factors are incremented to represent the spatial structure of unit activation, i.e. the combination of unit activity of each pattern; on the other hand, the time parameters are updated to record the time delay between the units activation, i.e. the time delay between the two patterns' occurrences. The connection parameters are asymmetric and, thus, associations are directional. Training is dynamic and occurs each time the output of one event detector module is activated. Each input pattern is memorised for a period of  $M$  cycles through the self-recurrent connections (see explanation in page 4.3.5) and is correlated with all other patterns appearing during this period. Time parameters and confidence factors are updated following Hebbian rules: once a unit  $i$  is activated, i.e. its output is maximal  $y_i = 1$  (recall that the unit's output takes values between 0 and 1), afferent connections to this unit from previously or simultaneously activated units  $j$ , i.e. those for which  $y_j > 0$ , are updated. Only the connections directed to (and not from) the newly activated unit  $i$  are updated. Connections leading to the most recently activated unit are updated and this only during the first cycle in which this unit is fully activated. The update rules for each parameter are given in equations 4.4 and 4.5.

$$tp_{ji}(t) = \frac{tp_{ji}(t-1) \cdot \frac{cf_{ji}}{a} + \frac{y_j(t)}{y_i(t)}}{\frac{cf_{ji}}{a} + 1} \quad (4.4)$$

$$cf_{ji}(t) = cf_{ji}(t-1) + a \quad (4.5)$$

The time parameter  $tp$  records the time delay between the activation of the two units which are linked by the connections; the short-term memory mechanism causes all  $y_j$  values to decrease at each cycle by the same factor, as explained in *short term memory* paragraph, thus the ratio between  $y_i$  and  $y_j$  gives a notion of their relative delay of activation. The time parameter value is calculated as the arithmetic mean value of time delay over all training data and its value is contained between  $(tp_{ii})^M$  and 1; the closer the two events, the bigger the time parameter;  $tp_{ij} = 1$  when the two events are simultaneous.

Note that the update rule for the connection parameters is not strictly Hebbian, in the sense presented in page 4.2.1. In agreement with the first part of the rule, the parameters are updated whenever two units are co-activated. However, the update of the time parameter does not lead to its increase (strengthening of the connection) as it is the case for the confidence factor and it is stipulated by the Hebb's rule. Rather, the update leads to the calculation of the mean value of time delay of the units; activation over all training examples. In addition, the parameters are not updated in the case of asynchronous activation (on the contrary, asynchronous activation means co-activation in our case, because of the memory of activation), as in the second part of the rule. However, because we used the term Hebbian rule in the journal publication of the model [Billard & Hayes 99] and wish to keep this text consistent with the published one, we use the term Hebbian update rule to refer to the training algorithm in the rest of the thesis. We apologise for any confusion arising from this labelling.

The confidence factor keeps a memory of the frequency of a pattern's occurrence. Its value is incremented at each updating step by a fixed quantity  $a$ .<sup>8</sup> In the experiment, the increase of the confidence factor is linear following a fixed slope of value  $a$ .<sup>9</sup> At the beginning of the experiment, the values of confidence factors and time parameters are set to 0 for all connections apart from the self-recurrent one, which have predefined values for these parameters determining the duration of short-term memory of the unit

---

<sup>8</sup> In order to prevent the confidence factor values from becoming too large in the experiments, all values are rescaled by dividing by a factor of 100 when they reach the value of 1000; the increase factor  $a$  is also rescaled by the same factor to keep the same proportional increase between the time parameters.

<sup>9</sup> An interesting option is to make the slope proportional to the value of  $cf_{ji}$ , the more confident the greater the increase. This would speed up the learning and may increase the robustness of the model against noisy data, by giving a greater influence to nodes that are more often activated.

activation.

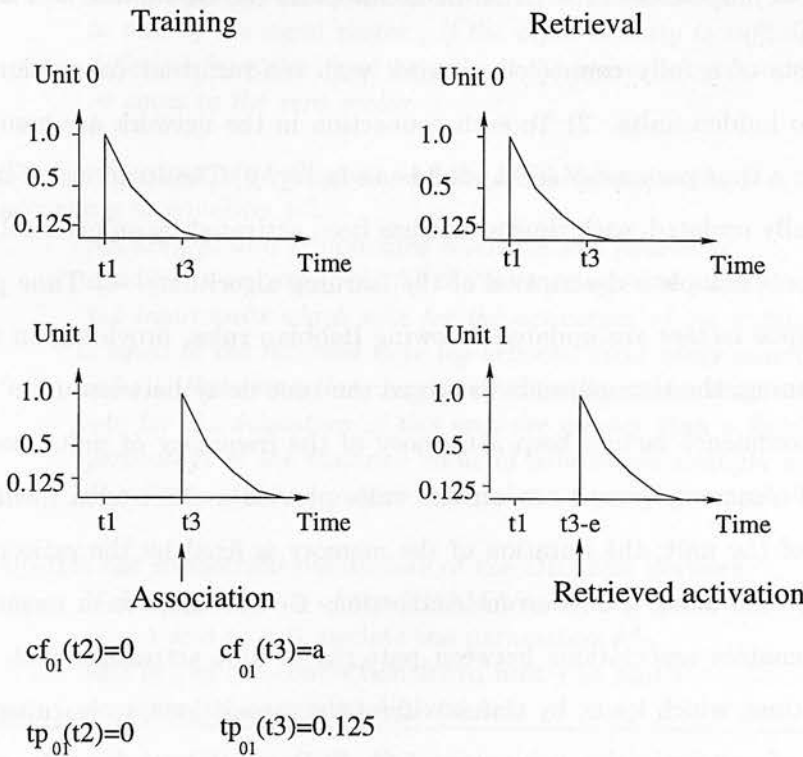


Figure 4.9: Example of association between unit 0 and unit 1.

In figure 4.9 left, we show an example of association between unit 0 and 1. Unit 0 is activated at time 0, while unit 1 is activated at time 3. Association is done at time three. The parameters of the connection from unit 0 (last activated) to unit 1 (most recently activated), i.e.  $cf_{01}$  and  $tp_{01}$ , are updated. Before association, the value of  $cf_{01}$  and  $tp_{01}$  are both zero (no correlation yet). After association,  $cf_{01} = a$ , where  $a$  is the increase factor of equation 4.5, and  $tp_{01} = \frac{0.125}{1} = 0.125$ , where 0.125 is the activity level of unit 0 output after three decrease steps (the ratio of decrease along the self connection is equal to 0.5). On the right hand side of figure 4.9, we show retrieval of activity of unit 1 after activating unit 0 at time 1. Following condition of factor  $A$  of equation 4.3, unit 1 is reactivated at time 3 minus  $e$ , the error on time delay between the two unit coactivation.

## 4.4 Summary

The principal properties of the DRAMA architecture can be summarised as follows:

- 1) It consists of a fully connected network with self-recurrent connections on each unit and no hidden units.
- 2) To each connection in the network are associated two parameters: a time parameter and a confidence factor.
- 3) The structure of the network is dynamically updated, each time a unit has been activated by an external input (see table 4.2 for a complete description of the learning algorithm).
- 4) Time parameters and confidence factors are updated following Hebbian rules, providing an associative type of learning; the time parameters record the time delay between units' activation while the confidence factors keep a memory of the frequency of units' coactivation.
- 5) The self-recurrent connections on the units provide a short-term memory of the activation of the unit; the duration of the memory is fixed by the ratio of decrease of the activation along the recurrent connection.
- 6) The short-term memory of unit activation enables associations between patterns of unit activation that have been delayed in time, which leads, by transitivity of the associations, to learning sequences of patterns of unit activation (see section 5.2).
- 7) Data retrieval depends on the value of the time parameters and confidence factors associated with the connections, which act as separate filters on the spatial and temporal features of the input; output units are activated when the two following conditions are satisfied: (i) the time delay since the input's time of occurrence is equal to the memorised temporal correlation and (ii) the confidence factor values of all active input units are greater than a fixed percentage of the maximal value of confidence factor of all active units in the network at the time of retrieval.

### 4.4.1 Discussion

As mentioned in section 4.2, the DRAMA architecture has several characteristics in common with both associative memory models and recurrent neural networks. It differs however from any existing models because 1) it combines a recurrent structure while using Hebbian learning rules<sup>10</sup> and because 2) the connections of the network

---

<sup>10</sup> To our knowledge, such a model has never been studied so far.

Table 4.2: DRAMA training algorithm

1:	<p>Present an input <math>I</math> to the system. Compute the output of the corresponding event detector following equation 4.1.</p> <p><i>The output vector of the event detector is either equal to that of the input vector, if the input activity is sufficiently different from the input activity at the previous cycle, or equal to the zero vector</i></p>
2:	<p>Compute output <math>y_i</math> of all units <math>i</math> of the DRAMA network, according to equation 4.2.</p> <p><i>An Output unit is activated when the two following conditions are satisfied: (i) the time delay since activation of the input units which vote for the activation of the output units is equal to the recorded time lag between these units coactivation and (ii) the connection weights of all active input units which vote for the activation of this unit are greater than a fixed percentage of the maximal value of connection strength between all active units and output units at the time of retrieval.</i></p>
3:	<p>Update the connection parameters of the DRAMA network:</p> <p>If <math>\exists i</math> and <math>j</math> (units of the DRAMA network), s.t. <math>y_i = 1</math> and <math>y_j &gt; 0</math>, update the parameters <math>cf_{ji}</math> and <math>tp_{ji}</math> of the connection <b>from</b> unit <math>j</math> to unit <math>i</math> according to equations 4.5 and 4.4.</p>

are associated with two parameters (instead of one) which keep a separate record of the spatial and temporal structure of the input patterns.

Similarly to other Hebbian networks, training and retrieving algorithms used in DRAMA are computational fast and cheap. The network has similar properties to that of associative memory models, such as the Willshaw network: Hebbian update rule, statistical retrieval process based on a threshold strategy, bidirectionality of the association and multiple dimensionality of the input-output patterns. However, it differs from usual Hebbian network as it uses a recurrent structure and encapsulates explicitly a notion of the time in one of the connection parameters.

The DRAMA architecture is a fully connected recurrent neural network without hidden units. Similarly to usual RNNs, the recurrent structure of the DRAMA architecture keeps a short-term memory of previous unit activation, which provides a notion of contextual activation. This provides the model with the ability for learning time series

of inputs. The short-term memory of unit activity is used to associate inputs, whose occurrence was delayed in time, and to retrieve the sequence of inputs following the same temporal patterns as observed in the training sequence.

The DRAMA architecture is different from usual RNNs because it uses an unsupervised learning algorithm, namely a Hebbian rule, for the update of the connection parameters. By contrast to backpropagation algorithm for the weight update, the Hebbian rule is time and input type independent and takes only one time step of computation. Similarly to dynamical RNNs, the DRAMA architecture encapsulates explicitly the time in the time parameters, associated with each network connection. However, by contrast to other dynamical RNNs, the time parameter in the DRAMA architecture is used as part of a threshold-based retrieval algorithm, a competitive process based on a threshold strategy which depends on the current activity of all units of the system (while in usual dynamical RNNs, the time parameter acts on each input unit independently on the activity of other units).

The performance of the DRAMA architecture are analysed theoretically and through numerical simulations in chapter 5.

## Chapter 5

# Evaluation of DRAMA performance

This chapter evaluates formally and through numerical simulations the performance of the DRAMA architecture. The mathematical description of the model is given in chapter 3. We first discuss the model as an associative memory and then as a model for time series prediction. As part of this discussion, we present examples of the model's performance in storing different types of patterns. We finish this chapter with a summary of the main properties of the model and a discussion of its performance in comparison to a list of desired criteria for learning models.

### 5.1 Model functionality

The DRAMA architecture functions as an associative memory, which associates pairs of input-output patterns with delayed time of occurrence, leading to learning of time series. It is a fully connected network with recurrent connections, which provides a short-term memory of unit activation. The network's input patterns are thus recorded for a short delay, during which they are associated with any new input pattern incoming during this delay. The time delay between each pattern occurrence and the structure of unit activity of each pattern are learned separately in the two parameters attached to each network connection, namely the *time parameter* and the *confidence factor*. Once an input-output pair has been learned, presentation of the input to the net retrieves the output after the recorded time delay. Retrieval of the output units' activity results from

a threshold-based retrieval mechanism applied to the spatial and temporal structure of the pattern of input units' activity. We next consider the influence of each of these two thresholds on correct retrieval separately.

### 5.1.1 The role of the spatial filter

If the time delay between input and output patterns is constant, then the term  $A$  in equation 4.3 is always equal to 1, and thus the retrieval function  $G$  depends only on the term  $B$ ; that is, it applies only to the spatial structure of input units' activity. In this case, the model has similarity with Hebbian networks, such as the Willshaw net, which use a threshold-based retrieval strategy. Correlated occurrences of two input patterns are distinguished from randomly generated ones by keeping a record of the frequency of correlated activation of these patterns' units (increment of the confidence factor ( $cf$ ) parameter of the connection). A pair of input-output patterns is said to be correlated once the two patterns have been associated more often than a minimal noise threshold, corresponding to the threshold  $max(cf)/T$  of equation 4.3 in our case. Correct retrieval of the output pattern of the pair, given the input one, depends on the proportion, relative to the above threshold, of correct over noisy associations each pattern has with other patterns. Correct storage of pattern pairs is then influenced by the choice of threshold used in the retrieval function and by the choice of encoding for the patterns, as the ratio of  $cf$  parameters will depend on the proportion of overlap between the two patterns' encoding. This section discusses the influence of the pattern encoding on the network's ability to store patterns correctly. We consider three cases, namely when the patterns overlap on both their inputs and their outputs, when they overlap on their inputs only and when they overlap on their outputs only. The influence on correct retrieval of the threshold  $max(cf)/T$  will be discussed in section 5.1.4.

#### Pattern overlap on both input and output

Let us define a training pattern as an input-output pair. Two overlapping patterns can overlap either on their inputs, on their outputs or on both. In the latter case, the patterns share a common activated connection which links the common activated input and output units. Training the network with patterns overlapping on both input

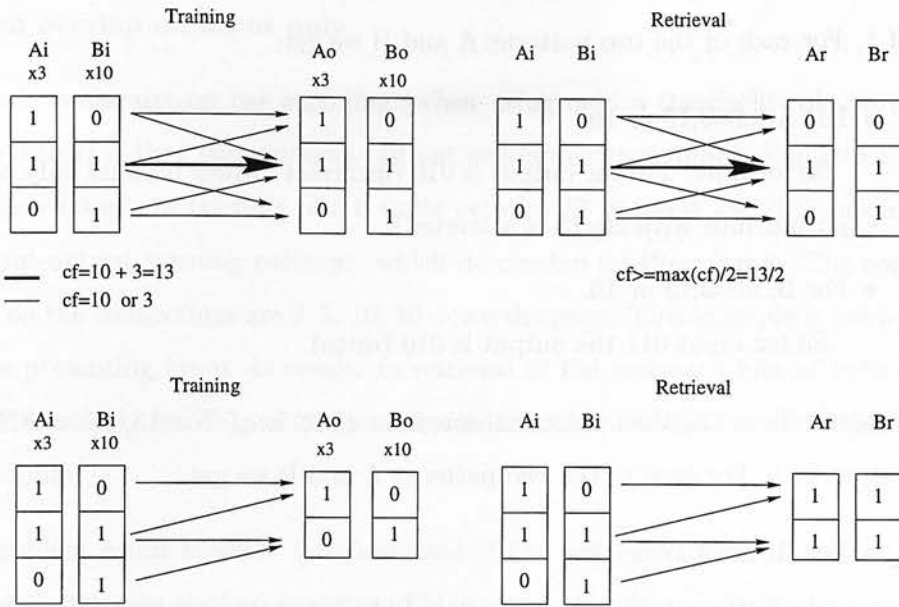


Figure 5.1: Examples of training pattern configurations which lead to unsuccessful learning when retrieval is based only on a threshold rule on the confidence factor. Bottom example becomes successful 1) when retrieval is based on both a threshold rule on the confidence factor and a threshold rule on the time parameter, or 2) when retrieval is based only on a threshold rule on the confidence factor and on a threshold rule on the number of winning units (see text for explanations).

and output results in unsuccessful learning of at least one of the two patterns. In this case, the common connection is more often activated than all other connections, as it is activated at each presentation of one of the overlapping patterns, and consequently the value of its associated confidence factor is bigger than those of the rest of the connections. If the ratio between confidence factors of overlapping and non overlapping connections is greater than  $T$  (see equation 4.3), then the network fails to retrieve correctly the whole pattern, activating only the common connections.

Figure 5.1 top shows an example of training of a 6 units network (3 units in input and 3 units in output) with two input-output training patterns ( $A_i \rightarrow A_o$  and  $B_i \rightarrow B_o$ ), which overlap on both their input and output. Given a frequency of activation of 3 and 10 respectively for the first and second pattern, the confidence factors on the connections in figure 5.1 top are 3, 3, 3, 13, 10, 10, 10 down the page. Now, let us look at three different values of  $T$ .

1. Suppose  $T = 1$ , then the condition from (4.3) is  $cf \geq 13/1 = 13$  for an output

of 1. For each of the two patterns A and B we get:

- For A,  $cf=3,13$  or 10.

So for input 110 the output is 010 (incorrect answer because only common output unit with  $cf=13$  is activated).

- For B,  $cf=3,13$  or 10.

So for input 011 the output is 010 (ditto).

2. Suppose  $T = 1.5$ , then the condition from (4.3) is  $cf \geq 13/1.5 = 8.7$  for an output of 1. For each of the two patterns A and B we get:

- For A,  $cf=3,13$  or 10.

So for input 110 the output is 011 (incorrect answer because now the B-connections with  $cf=10$  are also activated).

- For B,  $cf=3,13$  or 10.

So for input 011 the output is 011 (the correct answer, the A connections are not activated).

3. Suppose  $T = 4.5$ , then the condition from (4.3) is  $cf \geq 13/4.5 = 2.9$  for an output of 1. For each of the two patterns A and B we get:

- For A,  $cf=3,13$  or 10.

So for input 110 the output is 111 (incorrect answer because now both the A and B-connections are activated).

- For B,  $cf=3,13$  or 10.

So for input 011 the output is 111 (ditto).

To sum up, the region of  $T$  splits up into:

- $T$  between 1 and  $13/10 = 1.3$ ; output is 010 in both cases.
- $T$  between  $13/10$  and  $13/3 = 4.3$ ; output is 011 in both cases.
- $T$  greater than  $13/3$ ; output 111 is in both cases.

Thus, whichever value is taken for  $T$ , learning of both patterns is not possible.

### Pattern overlap on input only

The main constraint on the encoding (when using only a threshold rule on the confidence factor) is that two patterns can not overlap on their input. Figure 5.1 bottom shows an example of training of a 5 units network (3 in input and 2 in output) with two input-output training patterns, which do overlap on their input. The confidence factors on the connections are 3, 3, 10, 10 down the page. This example is not learnable because presenting input  $A_i$  results in retrieval of the activated bits of both  $A_o$  and  $B_o$  (and similarly for  $B_i$ ), as the common input unit votes for the activation of both output units.

This problem could however be eliminated if the net could use some other rule to distinguish between the two patterns. This second rule should exploit another source of information than the spatial form of activity of the units that is exploited by the threshold of confidence factor parameters; in the experiments we used either the information given by counting the sum of active units or the information on the time delay of input-output occurrence given by the time parameter. Information on the sum of active units was exploited by using the same threshold rule as in the Willshaw network's retrieval mechanism (see explanation section 4.3.1, p. 59). For recall, the retrieval mechanism in the Willshaw network requires that only the output units which have a number of positive connection weights greater than or equal to the number of active inputs are activated. The effect of that rule on the capacity of the network will be further discussed on page 85.

In the experiments of chapter 11, we will use three threshold rules, namely the two threshold rules on confidence factor and on time parameter of (4.3) and the Willshaw net's threshold rule on the sum of active units. This third additional condition/rule requires then that the number of input units which satisfy the two conditions on confidence factors and time parameters is equal to the number of active input units. Note that the number of units which satisfy the conditions on  $cf$  and  $tp$  of (4.2) cannot be greater than the number of active units, as the retrieval rule of (4.2) applies only to the active units, see explanations of page 68. We will see in section 5.1.5 that using the information on the time of occurrence of the input-output pair, given by the connection time parameter, has a similar effect on the retrieval as would a threshold on the input

unit activity.

### Patterns overlap on output only

Finally, it remains to consider the case in which patterns overlap on their outputs only. In that case, all patterns are correctly learnt, as the activation of the output units during retrieval is based only on the activation of the input units which is unambiguous. Let us consider the following example: let the two input-output pairs  $A$  and  $B$  of a 2 by 2 network be  $A : 10 \rightarrow 10$  and  $B : 01 \rightarrow 11$ .  $A$  and  $B$  overlap on the first bit of their outputs but are orthogonal on their inputs. Let us take a value for  $T$  such that the confidence factors of all connections are greater than the threshold  $\max(cf)/T$  (i.e. all connections satisfy the threshold rule on confidence factor). Then, presenting input of  $A$  (10) activates the first input unit which votes univocally for the activation of the first output unit. The connection parameter from input unit 1 to output unit 1 is the only one which is non zero (it was increased during presentation of pattern  $A$ ); the parameters of all other connections starting from the first active line are zero (there is in fact only one such connection in the example which links input unit 1 to output unit 2); consequently, only the output unit 1 will be activated under the vote of input unit 1. Similarly, when presenting input of pattern  $B$ , the only active input unit (unit 2) votes for the activation of both output units 1 and 2, as both connection parameters have been increased during presentation of pattern  $B$  (they have the same value as they have been increased the same number of times; thus, both pass the threshold).

### Orthogonal pattern

Note that the case where the patterns are orthogonal (that is, they overlap neither on their input nor output) is trivial. All patterns are perfectly learnt. For illustration, one can take the above example with  $A : 10 \rightarrow 01$  and  $B : 01 \rightarrow 10$  and apply the same reasoning as in the case in which the patterns overlap only on their outputs. That is, take a value for  $T$  such that all connection parameters pass the threshold  $\max(cf)/T$  and work out the retrieved output pattern, given each input pattern.

### Summary

To summarise, if we use a threshold rule on the confidence factor only, patterns that overlap on input (whether overlapping on output or not) cannot all be learnt. To extend the number of overlapping patterns that can be learnt we include threshold rules on the time parameter and/or the number of active units. In this case, some patterns that overlap on input or on input and output can also be learnt. The conditions on the type of these patterns will be discussed in the following sections.

### 5.1.2 Considerations on factors affecting the ability of the network to store binary patterns without time

The previous discussion allows us to now consider factors affecting how the DRAMA network can store binary patterns without time. Following Graham & Willshaw [Graham & Willshaw 96], we define the network *capacity* as the number of input-output patterns that can be stored before there is one bit in error in the recall pattern output. In this section, we discuss the effect of different choices of input-output pattern combinations together with different choices of threshold rules on the capacity. We try to determine the form which input-output patterns should take in order to reach the maximal capacity of the network. In the experiments, this information helps us to determine the encoding of the robot's sensor-actuator state and the encoding of the signals (words) in the experiments so as to allow correct pairing of signals (words) with other sensor-actuator measurements. Moreover, this can be used to optimise the ratio between the number of nodes of the network (which overloads the stack memory) and the number of words which the robot can be taught. Note that this discussion is by no means a comprehensive analysis of the network capacity, in which one would determine analytically or numerically the capacity of the network as a function of the encoding. Such an analysis would be very interesting (especially in the case when the encoding is unknown). It was however beyond the scope of this thesis.

Let us first consider the case in which retrieval is based only on the threshold rule on the confidence factors; that is, the retrieval equation 4.3 uses only the factor  $B$  ( $A = 1$ ) and there is no threshold rule on the number of active units, as discussed in the previous section. In that section we showed that recall is perfect if and only if none

$A_i$	$B_i$	$C_i$		$A_o$	$B_o$	$C_o$
1	0	1		1	0	0
1	1	1		0	1	0
0	1	1		0	0	1

Table 5.1: Incorrect example for storage of 3 patterns with overlapping inputs.

of the training patterns has any active input units in common. With only the threshold on confidence factor, the maximal capacity is, thus, obtained when each pattern has only one active input unit. The number of such patterns which can be formed given a network of  $N$  units is simply equal to  $N$ . These patterns corresponds to the basis of the input space.

Let us now consider the case in which we use both the threshold rule on confidence factor and the threshold rule on the number of active units as in the Willshaw network's retrieval mechanism (see explanation section 4.3.1, p. 59). Using this second threshold rule, which requires that the number of active connections to an output unit is equal to the number of active input units, improves the capacity of the network by eliminating the requirement of using only orthogonal input patterns. In this case, learning of patterns with overlapping inputs is made possible, as long as, in any pair of patterns, each pattern has **at least** one active input unit which is not active in the other.

### Example

Let us first consider an example which is not learnable: Let us consider a 6 units network (3 units in input, 3 units in output) in which we store three patterns:  $A : 110 \rightarrow 100, B : 011 \rightarrow 010, C : 111 \rightarrow 001$  (see table 5.1). Let us assume that the threshold on confidence factor is such that all connection parameters pass it. Then, presenting  $C_i = 111$  recalls correctly 001, as the three active input units all vote for the activation of the third one. However, presenting  $B_i = 011$  recalls 011 (incorrect output), because the two active input units vote for the activation of both the second output unit (following training of pattern  $B$ ) and the third one (following training of pattern  $C$ ). Similarly, presenting  $A_i = 110$  recalls 101 (incorrect output). Learning of pattern  $A$  and  $B$  fails here because both  $A$  and  $B$  are sub-patterns of  $C$  and cannot be distinguished from it.

$A_i$	$B_i$	$C_i$		$A_o$	$B_o$	$C_o$
1	0	1		1	0	0
1	1	0		0	1	0
0	1	1		0	0	1

Table 5.2: Correct example for storage of 3 patterns with overlapping inputs.

If we take the three patterns such that they all have one active input unit which is not active in the other, e.g.:  $A : 110 \rightarrow 100$ ,  $B : 011 \rightarrow 010$ ,  $C : 101 \rightarrow 001$  (see table 5.2), this 'extra' free input unit allows now to disambiguate between the activity of the two input units common to each pattern, following the threshold rule on the number of active input units. Presenting  $A_i = 110$  recalls 100 because both input unit 1 and input unit 2 vote for the activation of output unit 1. Note that input unit 1 votes also for the activation of output unit 3 (following training of pattern  $C$ ), but as input unit 2 does not vote for it (the number of voting input units, here 1, is not equal to the number of active input units, here 2), output unit 3 is not activated.

Similarly, the example of figure 5.1 bottom on page 81 becomes learnable when introducing this second threshold rule. In that example, retrieval of both output patterns  $A_o, B_o$  is correct because the second output unit (the bottom one in the pattern  $A_o$  and the top one in the pattern  $B_o$ ) is no longer activated. These units are no longer activated because only one input unit votes (i.e. the confidence factor of the connection passes the threshold) for their activation in each case, while two are required as there are two active input units.

The maximal number of patterns satisfying this condition, which can be formed given a network of  $I$  input units, is equal to the maximal number of possible combinations of  $p$  active input units which can be made using from  $p = 1$  to  $p = I$  active units per pattern:

$$\max_{p=1}^I \binom{I}{p} = \max_{p=1}^I \frac{I!}{(I-p)! \cdot p!} \quad (5.1)$$

This quantity is maximal when  $(I-p)! \cdot p!$  is minimal, which is the case when  $I-p = p$ , i.e.  $p = \frac{I}{2}$  for even  $I$  and  $p = \frac{I+1}{2}$  for odd  $I$ .

This statement can be better viewed in the light of an example. Let us consider a network with 6 input units. The maximum number of possible combinations is obtained

when using  $I/2 = 6/2 = 3$  active input units. The possible learnable combinations are then: 111000, 110100, 10010, 110001, 101100, 101010, 101001, 100110, 100101, 100011, 011100, 011010, 011001, 010110, 010101, 010011, 001110, 001101, 001011, 000111. There are thus  $20 = 6!/(3! \cdot 3!)$  possible combinations.

### No overlap on both inputs and outputs

However, when overlapping inputs are used we must in addition ensure that we do not use patterns that also overlap on their output (see section 5.1.2). The capacity of the network therefore has a more stringent upper bound that follows from this condition.

Two patterns that overlap on input and output use the network connections between the overlapping input and output units twice, and these multiply-used connections are the source of the problem. If we then insist that each network connection be used only once, we ensure that patterns do not overlap on both input and output at the same time. There are  $N \cdot (N - 1)$  connections in a network of  $N$  units with asymmetric connections. Thus, the maximal capacity of the network is equal to:

$$C = \min\left\{\frac{I!}{\frac{I}{2}! \cdot \frac{I}{2}!}; N^2 - N\right\}$$

To restate, the above presented capacity considerations can be helpful in allowing us to determine the best sensor-actuator state encodings for small  $N$ , but these are by no means a complete analysis of the network's capacity.

### 5.1.3 Storage of competitive rules

It is important to note that, in our model, the confidence factor threshold of activation on the output, calculated as  $\max_{y_j > 0}(cf_{ij})/T$  (see equation 4.3), is not an invariant but depends on the units activity at the time of retrieval.  $\max_{y_j > 0}(cf_{ij})$  is calculated as the maximal value of confidence factor for all the connections associated with units  $i$  that are activated at the time of the retrieval. If we view learning of a correlation between two units  $i$  and  $j$  as a causality rule of the type  $(y_i(t) = 1 \rightarrow y_j(t + \delta(t)) = 1)$ , then this retrieval method allows storage of several competing rules, whose activation depends

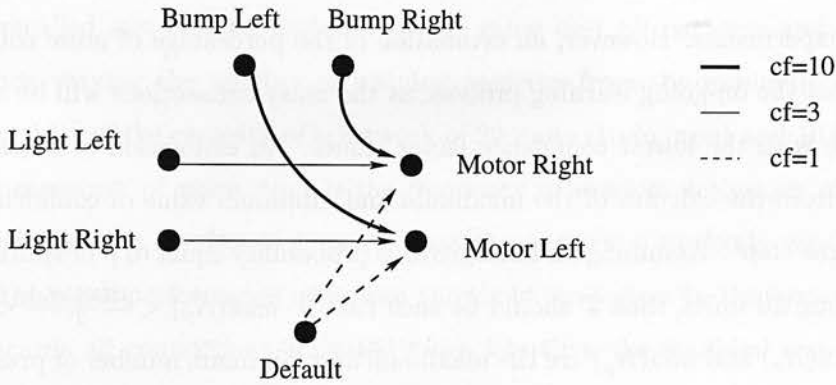


Figure 5.2: Values of the connection parameters ( $cf$ ) for determining basic behaviours of obstacle avoidance, light phototaxis and wandering. The predominance of one behaviour over the other is determined by the ratio of their respective connection strength.

on them winning the competition on confidence factor at the time of the retrieval. In the robotic experiments reported in chapters 7 to 11, this retrieving method is used to determine the robot's behaviour. Several sensor-motor rules are predefined to provide obstacle avoidance with bumpers and infra-red, light phototaxis and wandering (constantly active default unit), as shown in figure 5.2. The predominance of each rule over one another is determined by their mutual ratio of confidence factors values. At each cycle, the motor output is determined by the winning rule over all competing rules (the competing rule are those whose sensor input is currently activated). The obstacle avoidance behaviour (determined by infra-red sensor reading and bumper activation) is dominant, as the ratio of its confidence factor to all other behaviours is greater than  $1/T$  ( $T = 0.5$  in the experiments).

#### 5.1.4 The importance of choosing the threshold correctly

The capacity of the network to recall correctly all stored patterns depends on choosing a good threshold  $\max(cf)/T$  of activation on the confidence factor values of the connections. It should be high enough to discard activity of spurious connections, that is connections that have been activated by spurious active units, and low enough to allow for competitive activation of rule-like patterns (as described earlier). Estimating correctly the percentage of noise (spurious unit activity) in the system, and thus the correct value for  $T$ , before learning is often not possible, especially in unsupervised

learning experiments. However, an estimation of the percentage of noise could be obtained from the on-going learning process, as the noisy connections will be associated in average with the lowest confidence factor values. An estimation of  $T$  can then be obtained from the calculus of the maximum and minimum value of confidence factor, at each time step<sup>1</sup>. Assuming an homogeneous probability equal to  $\rho$  of spurious activation among all units, then  $T$  should be such that  $\rho \cdot \max(N_p) < \frac{\max(N_p)}{T} < \min(N_p)$  where  $\max(N_p)$  and  $\min(N_p)$  are the maximum and minimum number of presentations of a pattern over all patterns. Thus, a possible estimation of  $T$  from the measured confidence factor values is such that:

$$\frac{\max_{\{y_i>0\}}(cf_{ij})}{T_a} = 2 \cdot \text{mean}_{\{y_i>0\}}(cf_{ij}) - \frac{\max_{\{y_i>0\}}(cf_{ij}) + \min_{\{y_i>0\}}(cf_{ij})}{2} \quad (5.2)$$

where  $\max_{\{y_i>0\}}(cf_{ij})$ ,  $\text{mean}_{\{y_i>0\}}(cf_{ij})$  and  $\min_{\{y_i>0\}}(cf_{ij})$  are the maximum, mean and minimum values of confidence factor over all activated units at the time of retrieval.

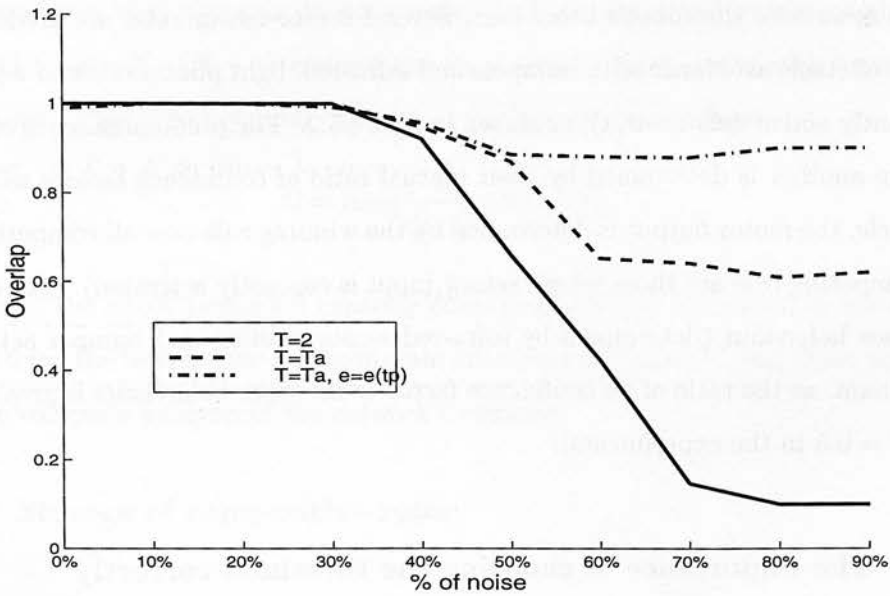


Figure 5.3: Overlap between desired and actual output during recall of a 20 units network (10 In, 10 Out) for different values of noise (spurious unit activity) and three different threshold strategies: (1)  $T = 2$ , no threshold on time; (2)  $T = T_a$ , no threshold on time; (3)  $T = T_a$ ,  $e = e(tp)$ .

In figure 5.3, we show the recall performance of the network, defined as the overlap

<sup>1</sup> Note that a minimal boundary for the confidence factor is required (e.g. the global minimal value of all confidence factors) for propagation of activation, in order to eliminate activation of random rules when there is no other rule to compete with them.

between recalled and trained patterns (mean value over all patterns and over 1000 runs), when varying the number of training patterns from the minimum (1) to the maximum (10)<sup>2</sup> of the capacity of a network of 20 units (10 in input and 10 in output) and the percentage of noise, that is the frequency of random activation of any unit during training. An overlap of 1 means that the pattern is perfectly retrieved. We compare the recall performance using two threshold strategies: In the first one  $T = 2$ , which discards all connections activated twice less than the maximal one; that is it accepts a maximal ratio of 50% of spurious input unit activation. In the second case,  $T = T_a$  approximates the ratio between regions of high and low confidence factors, following equation 5.2, which correspond to correctly and incorrectly activated connections. Results of figure 5.3 show that pattern recall is perfect up to a proportion of 30% of noise. The performance is in average better with the  $T_a$  strategy, as it allows better recall with a bigger proportion of noise. On page 150, we discuss further the influence of the threshold  $T$  on the estimation of success of the physical experiments of section 7.2.2.

The importance of correctly setting the threshold, such as to maximise the recall success of Hebbian like and Willshaw type associative memory has been discussed by a number of authors. For instance, [Dayan & Willshaw 91], [Buckingham & Willshaw 92], [Graham & Willshaw 95], [Sommer & Palm 98], [Schwenker *et al.* 96] evaluated theoretically and through numerical simulations different training rules and threshold strategies to optimise the signal/noise ratio during recall of a Willshaw net type of network, that is, a network using clipped Hebbian rule as presented in section 4.3.1. It would now be interesting to pursue further the above presented study of the threshold influence on the DRAMA retrieval performance, such as to make a complete and formal analysis of the problem.

### 5.1.5 The role of the temporal filter

The factor  $A(tp_{ij}^{kl})$  of the retrieving function  $G$  in equation (4.3) acts as a filter on the time lag between consecutive activation of two units  $i$  and  $j$ . It outputs 1 when the time

---

<sup>2</sup> The maximum of the capacity is  $10 + 10$ , but we consider only one direction of association in the simulation.

lag between activation of unit  $i$  and unit  $j$  is such that  $|y_i(t) - tp_{ij}(t)| < e$ , otherwise 0.  $tp_{ij}$  represents the mean value over all of the time lag of activation, while the factor  $e$  is the estimated error on the value of  $tp_{ij}$  whose formal definition is given below in equation 5.3. Correct retrieval of an input-output pair depends then on a correct evaluation of  $e$ . It should be big enough to include the maximal variation of time lag over all input-output pairs, while small enough to allow precise time prediction of the pattern output occurrence. Similarly to  $T$ , the threshold  $e$  on  $tp$  is not an invariant but is calculated at each time step according to equation 5.3. Equation 5.3 is similar to equation 5.2 which calculates  $T$ .  $e$  is an estimation of the threshold between spurious (too big and random) and relevant (with consistent variance) variations of the recorded time delay  $tp$  of activation of each connection. The variance of the time delay of activation for each connection is evaluated at each training step (i.e. whenever the parameters  $cf_{ij}$  and  $tp_{ij}$  of the connection from unit  $i$  to unit  $j$  are updated) by calculating the  $e_{ij}$  factors which appear in equation 5.3. The  $e_{ij}$  are the mean variation of time delay over all training examples.

$$e = \text{mean}_{y_i > 0}(e_{ij}), \quad e_{ij} = e_{ij}(t) = \frac{e_{ij}(t-1) \cdot \frac{cf_{ij}}{a} + |tp_{ij} - \frac{y_j(t)}{y_i(t)}|}{\frac{cf_{ij}}{a} + 1} \quad (5.3)$$

where  $\text{mean}_{y_i > 0}(e_{ij})$  is the mean value of  $e_{ij}$  over all activated units  $i$  and  $a$  is a fixed quantity by which the confidence factors are incremented at each time step.

In figure 5.3, we show that the recall performance of the network improves when using the time threshold in addition to the threshold  $T = T_a$  on  $cf$ . Retrieval is perfect up to a proportion of 30% of noisy data; otherwise the performance decreases up to a proportion of 90% overlap between retrieved and training data. Using the threshold rule on time parameter in addition to the threshold rule on confidence factor, therefore, improves the robustness of the network against noise. This is due to the fact that spurious activation of units during training affects retrieval less as the incorrectly activated connections can now be eliminated by two instead of one threshold rule. This can be better viewed with an example.

### Example

Suppose we store the input-output pattern pairs  $A$  and  $B$  with time delays  $T_a$  and  $T_b$

t=0	t=0		$t = T_a$	$t = T_b$
$A_i$	$B_i$		$A_o$	$B_o$
1	0		1	0
1	1		0	1
0	1		0	0

Table 5.3: Example of storage of 2 patterns  $A$  and  $B$  with different input-output time lag of occurrence,  $T_a$ ,  $T_b$ .

t=0	$\rightarrow$	$T_a - e$	$\rightarrow$	$T_b - e$
$A_i$	$\rightarrow$	$A_o$	$\rightarrow$	$B_o$
1		1		0
1		0		1
0		0		0

Table 5.4: Retrieval of output patterns following example of table 5.3.

as shown in table 5.3. Let us now consider the three possible different situations (note that we assume that  $T_a$  and  $T_b$  are different; if not, the discussion is the same as when considering only the spatial constraint of the confidence factor, see section 5.1.1).

1. Suppose  $A$  and  $B$  have been presented the same number of times, each activated connection has the same confidence factor. Then, presenting  $A_i$  is followed first by the correct activation of  $A_o$  after a time delay  $T_a - e$  ( $e$  is the error on time). However, as the 2nd input unit votes also for the activation of the 2nd output unit (following storage of the  $B$  pattern), the output  $B_o$  is also retrieved (erroneously) at time  $T_b - e$  (if  $T_b$  is shorter than  $T_a$ , the activation of  $B_o$  would then precede that of  $A_o$ ). Table 5.4 shows the retrieval result. Similarly, presenting  $B_i$  retrieves both  $A_o$  and  $B_o$  after the correct time delays.
2. Suppose  $B$  has been presented less often than  $A$  and the ratio of confidence factor between  $A$  and  $B$  is **lower** than the threshold  $\max(cf)/T$  of equation 4.3, then presenting  $A_i$  will no longer retrieve  $B_o$  as the connection between the 2nd input unit and the 2nd output unit will not pass the threshold on confidence factor. However, presenting  $B_i$  will no longer retrieve  $B_o$  and will retrieve only  $A_o$  after  $T_a - e$ , because then only the connection between the 2nd input unit and 1st output unit passes the threshold.

3. Suppose  $B$  has been presented less often than  $A$  and the ratio of confidence factor between  $A$  and  $B$  is **greater** than the threshold  $\max(cf)/T$  of equation 4.3, retrieval is the same as in the first case, that is  $A_i$  retrieves both  $A_o$  and  $B_o$ . However, if  $B$  has been presented with a variable time lag, that is with a  $T_b$  largely bigger than  $e$  and not constant, then presenting  $A_i$  no longer retrieves  $B_o$  (although it passes the threshold on confidence factor) because it does not satisfy the threshold on time which requires a variation of time delay smaller than or equal to  $e$ .

The third case illustrates our previous statement which said that the threshold on time parameter improves the robustness of the network as it allows more spurious patterns to be rejected. In our example  $B$  is a spurious pattern learnt together with training the net on  $A$ . The threshold on confidence factor  $\max(cf)/T$  does not reject the pattern  $B$  (this can happen as the threshold is updated continuously during training and makes an estimation of the average of noise). However, it is rejected by the threshold on time parameter. As  $B$  results from random/spurious activation of the network's units (as it should be the case for what we consider a spurious pattern, that is a non-correlated co-activation of units), the time delay of activation of the input-output pair varies greatly. The variation is so big that it is then eliminated by the threshold  $e$ .

The second case also shows an example of situation in which the network is successful at eliminating  $B$  assuming that  $B$  is a spurious pattern. However, if  $B$  is a relevant pattern, case 2 is an example of unsuccessful learning of two patterns overlapping on their inputs (only the first pattern  $A$  is correctly learnt). Case 1 shows successful retrieval of the two pairs  $A_i \rightarrow A_o$  and  $B_i \rightarrow B_o$  (that is correct activation of the output units of the corresponding pattern after the correct time delay), but accompanied with spurious activation of other units after or before activation of the correct ones. This example shows that patterns which overlap on their inputs can sometimes be correctly learnt (and retrieved) when using the information on time delay of occurrence of a pattern. It is thus an improvement as compared to the case when using only information on the spatial structure of the pattern, i.e. the confidence factor (recall that in that case only orthogonal patterns could be learnt). However, this is only half a solution, as correct retrieval of one output pattern is followed by spurious activation

of the other pattern output. In order to eliminate the activation of the second pattern, one should use the threshold rule on the number of active units as discussed on page 5.2.

### 5.1.6 Space and time efficiency

An important characteristic of the DRAMA architecture is that it is computationally fast and cheap. We here evaluate more formally what we mean by this.

Training of the network requires a number of time steps of computation equal to two times the number of connections, to update the connection parameters  $cf$  and  $tp$ . This is faster than the usual backpropagation algorithm which requires  $n$  times the number of connections, where  $n$  is the number of steps needed to reduce the error to a minimum and is often bigger than 2. Retrieval of unit activity is also relatively fast, as it does not require several time steps of computation (no convergence as e.g. in a Hopfield net). Retrieval requires summing twice over all units in order to first calculate the mean, maximal and minimal values of the  $cf$  and  $tp$  parameters which determine the values of threshold factors and to then sum the vote for activation of each unit.

Good time efficiency of a neural network model has usually the disadvantage of bad space efficiency. The decrease of the model's capacity in front of overlapping pattern encoding (as discussed in section 5.1.2) leads to a poor space efficiency. By using the three conditions on time and structure of the excitatory units pattern and on equal number of voting units, we were able to improve the capacity of the model to reach a maximal use of the encoding proportional to the square of the number of units of the network. It is, however, lower than the maximal combinatorial capacity of a network of  $N$  units, which is equal to  $N!$ .

Space efficiency is also determined by the number of global variables which are used by the network. The DRAMA architecture requires space for six times the number of connections to store the three parameters associated to all (directional) connections of the network. This is a higher number of global variables than usual NNs which use only one parameter per connection.

## 5.2 Sequence learning

In the previous section, we discussed the model performance for associating two input-output patterns. We consider here the model performance at learning sequences of more than two patterns. For this discussion, we consider a variation of the DRAMA retrieval algorithm described in section 4.3.5, p. 68. In addition to the two conditions on the confidence factor and time parameters encapsulated by the function  $G$  in equation 4.3 (p. 68), we add a third condition which requires that all units activated at time of retrieval agree on the activation of the output unit for this to be activated. This condition is similar to the threshold-based retrieval algorithm used in the original version of the Willshaw network (see section 4.3.1). This version of retrieval algorithm is used in the experiments reported in chapter 11.

The present version allows us to point out the two following facts: 1) the duration of the sequence that can be learned by the network is not restricted to that of the short-term memory  $M$ , as by transitivity of the associations a sequence of  $n$  steps can be derived from the association of shorter consecutive sequences; 2) the sequence is not restricted to include only strictly different patterns, but can be composed of several occurrences of the same pattern(s) (e.g. the sequence  $ABCDEFCDG$  with repetition of the sub-pattern  $CD$ ). This second result is due to the third condition, which requires the agreement of all voting units for the activation of the output pattern. Let us consider the sequence example  $ABCDEFCDG$ , with the following timing  $A(t) \rightarrow B(t+1) \rightarrow C(t+2) \rightarrow D(t+3) \rightarrow E(t+4) \rightarrow F(t+5) \rightarrow C(t+6) \rightarrow D(t+7) \rightarrow G(t+8)$  and a short-term memory duration equal to 3 steps, such that the following associations are being made among others:  $B \rightarrow E, F \rightarrow G$ . Activation of patterns  $E$  and  $G$  after activation of the subgroup  $CD$  is determined by the vote of patterns  $B$  and  $F$  respectively, in addition to the votes of  $C$  and  $D$ .

Correct retrieval of sequences which loop on one of the patterns or on a subgroup of patterns creates specific conditions for the patterns' structure and timing of the sequence. Let the training sequence be of the form  $ABCDEFCDG$ . Then learning is successful if:

1. The memory duration  $M$  is long enough to allow association between the patterns preceding and following the subsequence on which the sequence loop (i.e.  $B$ ,  $E$  and  $F$ ,  $G$  in the example).
2. If the number of occurrences of the same sub-pattern in the sequence is inferior to the threshold on confidence factors  $T$  so that it becomes activated: i.e in the example if  $N_{CD} = 2 < T$ , so that  $\frac{cf_{BE}=cf_{CE}=cf_{DE}}{cf_{CD}} \geq T$ .
3. If the pattern or subgroup of patterns on which the sequence loops does not occur at the beginning of the sequence, as a pattern previous to the loop is necessary to determine the activation of the correct subsequent pattern.

Note that learning a sequence with an internal loop is equivalent to learning two sequences with a common sub-pattern (e.g.  $ABCDF$  and  $FCDG$ ). Previous results imply then that the number of sequences that can be learned is not restricted, as long as the structure and temporal pattern of occurrence satisfy the previously mentioned conditions on  $M$ ,  $e$  and  $T$ .

### 5.2.1 Learning algorithm for the short-term memory parameter

The previous sections pointed out the importance of correctly choosing the values for the three learning parameters of our system, namely  $T$ ,  $M$  and  $e$  and determined bounds for these values relative to the structure of the training patterns and the percentage of noise in the system. However, as mentioned earlier, it is seldom the case that we can access this information before learning. Therefore, it is desirable to define a learning algorithm for tuning these parameters on-line, that is together with the associative learning process. In section 5.1.4, page 89, we gave a possible example of how to calculate on-line the values of the thresholds  $T$  and  $e$ .  $M$  can also be updated on-line according to the following algorithm:

$$\text{If } e \leq 0.1 \cdot M \text{ then } M = M + \sum_i (p_i - y_i) \quad (5.4)$$

where  $p_i$  is the predicted activation of unit  $i$ .  $M$  is updated only once the error on time  $e$  has settled to a small value.  $M$  is either increased or decreased depending on whether there has been more unpredicted unit activations (a zero activity prediction

while measuring a unit activation) than incorrectly predicted unit activations (a non zero activity prediction while measuring no unit activation).

Performance of the learning algorithm was evaluated through simulation for learning two types of sequences, namely *ABCDE* and *ABCDEFCDG*, for different noise proportions (the variation of the time delay of activation of each pattern) and with different starting value for the memory duration (10 times lower or bigger than the correct one). Results showed that the algorithm had converged after less than 50 trials up to 40% and 20% (by respect to each sequence) of noise. Statistical fluctuations around the correct value of  $M$  were observed when learning the sequence *ABCDEFCDG* with more than 20% of noise. Figure 5.4 top shows the variation of the parameters  $e$  and  $M$  and the error (number of incorrect prediction) while learning the sequence *ABCDEFCDG* with 20% of noise (variation in the timing of pattern occurrence) in the input. Convergence of the parameter values is achieved after 36 presentations of the sequence, that is the error is equal to zero. The time threshold has settled at a minimal value of 2, which is equal to the maximal variation in the timing of pattern occurrence (noise), and the value of the short-term memory has settled to a value comprised between the minimal and maximal value required by the conditions of section 5.2. Figure 5.4 bottom shows superposed plots of the retrieved (straight line) and training (dotted line) pattern activation when learning the sequence *ABCDEFCDG* under 20% of noise. The figure shows snapshots of (left) the three first cycles of the training, i.e. before convergence, and (right) the 37th training cycle, i.e. after convergence. The retrieved pattern activity begins only at the second cycle for pattern *C* and is incorrect in the third cycle as patterns *E* and *G* are activated twice instead of once. Retrieved and training pattern activity match for all patterns, apart from *A*, in the 37th cycle. Pattern *A* is the activation pattern for retrieval of the series; pattern *A* is not retrieved as it has not been correlated to any other pattern (the time lag between the end of one series, *G* pattern, and beginning of a new one, *A* pattern, is too long for *A* and *G* to be associated). The retrieved activations of patterns *B*, *D*, *E*, *F* and *G* occur slightly earlier than the training ones, in the margin of the 20% of noise (the variation of the time delay of activation of each pattern).

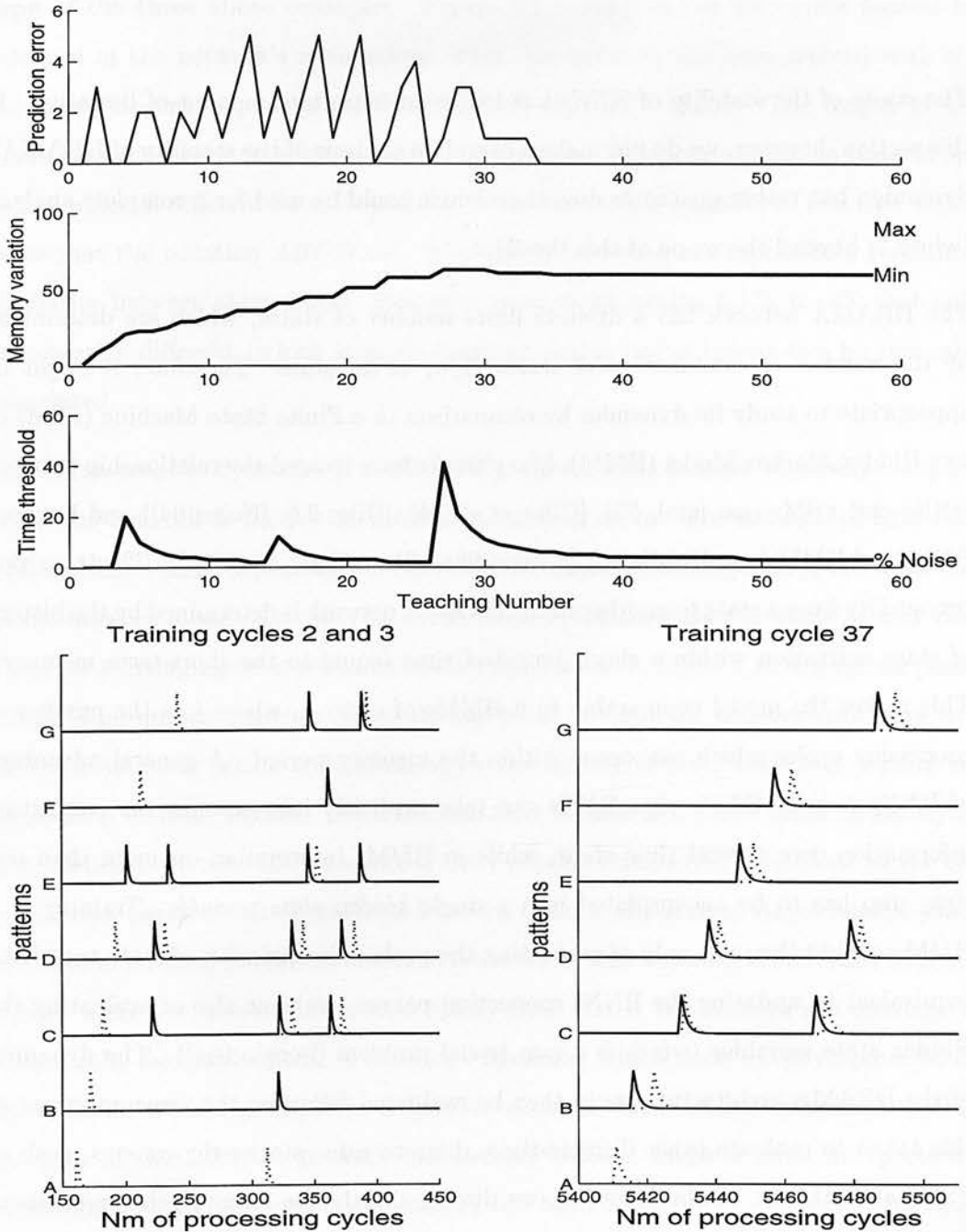


Figure 5.4: **Top:** Variation of short-term memory, error and time threshold during the training of the sequence *ABCDEFCDG*. **Bottom:** Superposed plots of the retrieved (straight line) and training (dotted line) pattern activation along the training; the figure shows snapshots of (left) the three first cycles of the training and (right) the 37th training cycle, that is after convergence.

### 5.2.2 The dynamics of the network

The study of the stability of RNNs has led to an important amount of literature. In this section, however, we do not make a complete analysis of the stability of DRAMA's dynamics, but rather give some directions which could be used for a complete analysis (which is beyond the scope of this thesis).

The DRAMA network has a discrete finite number of states, which are determined by the number of combination of states (1/0) of its units. Therefore, it might be appropriate to study its dynamics by comparison to a Finite State Machine (FSM) or to a Hidden Markov Model (HMM). Many works have treated the relationship between RNNs and FSMs (see [etal. 95], [Giles *et al.* 94], [Giles 92], [Kolen 94]) and between RNNs and HMM (e.g. [Bridle 90], [Bengio 96a], [Rumelhart *et al.* 86]). The transition probability from a state to another in the DRAMA network is determined by the history of state activation within a short period of time (equal to the short-term memory). This makes the model comparable to a HMMs of order  $k$ , where  $k$  is the number of processing cycles which can occur within the memory period. A general advantage of RNNs over HMMs is that RNNs can take explicitly into account the contextual information over several time steps, while in HMMs information on more than one time step has to be encapsulated into a single *hidden state variable*. Training of a HMM consists then not only of evaluating the probability function of state transition (equivalent to updating the RNNs connection parameters) but also of evaluating the Hidden state variables (which is a non trivial problem [Bengio 96a]). The dynamics of the DRAMA architecture could then be evaluated following the same approach as this taken to evaluate other discrete-time, discrete value stochastic systems, such as FSMs and HMMs. In the following, we discuss shortly the form which the orbits of the DRAMA network's state dynamics could take.

Figure 5.5 top shows three types of orbits which result from learning time series of the type 1) *ABA*, cyclic orbit; 2) *ABCDE* and *HJCKE*, crossing orbits on state *C*; 3) *ABCDEFCDG*, orbit with a loop on states *CD* and divergence on state *D*. The second and third examples result from the property of the DRAMA architecture at learning different time series with common sub-patterns (see section 5.2). This

characteristic of the architecture allows it also to learn combination of series of the type of the three above examples. Figure 5.5 bottom shows the orbits formed by retrieval of the network's associations, when the network has been trained with the three overlapping sequences  $ABCDEFCDG$ ,  $HIEJK$ ,  $MIM$ . The length of the curve between two points in the graph represent the time to go from one state to the other, which is different for each pattern, as the network is sensitive to real time delay. Note that the notation  $ABCD$  etc. is arbitrary and does not represent a structural similarity between these states. However, recall from section 5.1.2, p. 85, that only structurally different (which have no common active units) inputs can be mutually associated.

As we can see, the dynamics of the DRAMA network can become relatively complex, when the network is trained with overlapping time series. The example of figure 5.5 bottom shows that, depending on the initial conditions (starting state), the dynamics of the network forms either a cyclic orbit, transition from state  $I$  to state  $M$  and vice-versa, or a diverging orbit with one of the three states  $D$ ,  $E$  and  $I$  as divergence point. There is no unique attractor state. At any point of time, the network converges to a different attractor which is determined by the history of state activation during the short-term memory period. The transition from a state to another relies on satisfying the conditions on time and structure of these previous states (i.e. conditions on the two connection parameters). Transition between structurally similar states (states which are close in terms of Euclidean distance) is not easier than between structurally different states. Moreover, starting from structurally close states, the network can converge to structurally far apart states. A network's dynamics which leads to convergence to far apart states when starting from neighbour states is called *chaotic* [Haykin 94]. The stability of the DRAMA network can become chaotic, under a loose interpretation of the above definition, when the network has been trained over the maximum of its capacity. In this case, the network has been trained with numerous overlapping time series, which have involved all states of the network. Each state of the network will then correspond to a divergence point for two or more orbits, similarly to the points  $E$ ,  $D$ , and  $I$  of figure 5.5. In this case, there might be no stable state any more, that is all series will be linked to each other and retrieval of one will lead to infinite retrieval of all other series. In particular, if the conditions of the neuronal activation function (see



p. 68) on the time and structure of the sequence patterns are not respected, it is to be expected that the retrieval dynamics might be very sensitive to initial conditions. For instance, small variation of the values of the thresholds on time and structure,  $e$  and  $T$ , might lead to very different retrieval series.

### 5.3 Summary

The discussion of the previous sections of this chapter have shown the following functional characteristics for the DRAMA architecture:

- 1) The capacity of the network decreases importantly with overlapping encoding of the data; an orthogonal encoding is then preferable when possible, however leading to a poor space efficiency. The maximal capacity of the network is proportional to one time its size for storing pairs of input-output patterns with fixed delay and proportional to the square of its size for storing pairs with variable time delay.
- 2) Using the full capacity of the network, retrieval performance is perfect up to a proportion of 30% of noisy data; otherwise the performance decreases up to a minimal proportion of 90% overlap between retrieved and training data.
- 3) The model can learn sequences of pattern activation of the following types: (i)  $ABCDEF$ , i.e. a  $n$  steps sequence whose duration can be longer than the short-term memory, as it can be derived, by transitivity of the associations, from the association of shorter consecutive sequences (ii) a sequence composed of several occurrences of the same pattern(s) (e.g.  $ABCDEFCDG$  with repetition of the subpattern  $CD$  or  $ABBBBCD$  with three occurrences of the pattern  $B$ ).
- 4) Retrieval depends on correctly choosing the values of the three learning parameters, namely short-term memory duration, threshold on time parameter and threshold on confidence factor parameter; these values depends on the proportion of noise (imprecise sequence timing and spurious unit activation) and on the sequence type; theoretical boundaries are determined for these parameters.
- 5) An algorithm for tuning the learning parameters simultaneously to updating the network connection parameters is defined and validated through numerical simulations.
- 6) Finally, because training uses a one time-step algorithm, the model is computationally fast and inexpensive, which allows its implementation for real time computation and on-line learning in a basic hardware system. We describe such an implementation in

the chapters 6 to 11.

In table 6.1, we show the complete training algorithm of the DRAMA architecture with on-line training of the threshold and memory parameters.

Table 5.5: Complete Training algorithm

1:	Present an input $I$ to the system. Compute the output of the corresponding event detector following equation 4.1.
2:	Compute output $y_i$ of all units $i$ of the DRAMA network, according to equation 4.2.
3:	Update the connection parameters of the DRAMA network: If $\exists i$ and $j$ (units of the DRAMA network), s.t. $y_i = 1$ and $y_j > 0$ , <b>update</b> the parameters $cf_{ji}$ and $tp_{ji}$ of the connection <b>from unit <math>j</math> to <math>i</math></b> according to equations 4.5 and 4.4.
4:	Update the learning parameters $T$ , $e$ and $M$ according to equations 5.2, 5.3 and 5.4.

### 5.3.1 Discussion

In chapter 3, we presented the functional properties of the DRAMA architecture and discussed these by comparison to other NN models for associative learning and time series prediction. In previous sections of this chapter, we discussed the performance of the model. We conclude this discussion by comparing the DRAMA properties as a learning architecture to Schaal's list of important criteria for a NN model [Schaal 99]. Our choice of referring to this list was motivated by the fact that several of these criteria are relevant to applications on robot learning (Schaal does not refer to a particular application).

- *Autonomy: How many parameters of the learning approach need human or heuristic adjustment, like, for instance, learning rates?*

The values of the threshold parameters and of the short-term memory need to be carefully chosen to allow successful learning. In section 5.2.1, we have proposed an algorithm for on-line tuning of their values.

- *Bias: Is the learning algorithm inherently biased towards certain solutions, or, if*

*desired, can bias be incorporated easily?*

The learning algorithm is not biased toward any particular direction of association. Moreover, the general learning, which uses an infinite short-term memory, is not biased to make any specific association. However, it can be biased by restraining the direction of association between different sensor systems, by fixing the value of short-term memory to allow association only between events not older than a fixed limit and by fixing the value of the threshold parameters to limit by hand the acceptable percentage of noise of the data. Such bias have been used in some of the experiments, reported in chapters 6 to 11, in order to speed up learning, when sufficient information was available to fix in advance these parameters.

- *Continuous Learning: Can the learning system be trained forever without reaching a limit of its adaptation capacity or degrading its performance?*

The DRAMA network is particularly well suited for continuous learning, such as adaptive learning of a robot (see experiments of chapters 7 to 11). New information is incorporated to previous information smoothly as the effect of the weight update. New information will erase previous information only if the two information are incompatible (following constraints on the input regularity mentioned in section 5.1.2). The capacity of the model is however limited to a maximal number of binary associations of input pairs of order 1 of the network size (see p. 85) and a maximal number of time series of input of order 2 of the network size (see section p. 91).

- *Capacity: How many parameters are required to represent a given amount of experience?*

Information is contained in the active bits of the input patterns to the network. Representing experience (activation by an input pattern) in the DRAMA network consists of keeping a record of coactivation of the network units. There need two parameters, the time parameter and confidence factor associated with each connection, to represent each experience of coactivation of two units.

- *Plasticity: How well can learning follow a dynamically changing environment?*

The time needed for the system to answer correctly to new information corresponds to the time for this information to erase the memory of previous one, which is incompatible with the new information. This corresponds to the cases when the correlation involves perceptual stimuli which have been previously correlated with other stimuli.

Let us consider the example of a robot which learns to predict new perceptions given an action pattern, see before. The robot will begin to predict the correct stimuli once it has perceived the action-stimuli pair a number of times equal to a percentage (fixed by the threshold  $T$  of the neuronal activation function 4.2, see p. 68) of the number of times it perceived the previous action-stimuli pair. The speed of adaptation is thus fixed by the threshold on activation and the frequency of action-stimuli occurrence. The robot's adaptation will thus never be immediate (unless the new stimuli pair is compatible with previous learning). Moreover, if the threshold is high and the number of action-stimuli occurrences is slow compared to the dynamics of change of the environment, the robot might fail to adapt on time.

- *Generalisation: How well does the learning box infer an appropriate output for an unknown input?*

Presenting an unknown (not training) input might result in 1) no further unit activation if the input shares no subpattern of unit activity with any of the training inputs; 2) retrieval of the complete output of the training input pattern with which the input shares common subpattern of unit activation (e.g. learning input-output pattern  $[0011] \rightarrow [1100]$ , then presenting  $[0010]$  will retrieve  $[1100]$ , see section 5.1.2 for explanations).

- *Incremental Learning: Is learning possible in an experience by experience way?*

Incremental learning is the way Hebbian learning works and so the DRAMA architecture.

- *Interference: Does previous learner knowledge degrade if new data is incorporated in the learning box?*

Previous knowledge will be forgotten if the new knowledge is incompatible with

the new one and this after that the new knowledge has been presented a sufficient amount of time (see response to the above *plasticity* question).

- *Interpretation: When opening the 'black box', is it possible to interpret the way the correlation between input and output data has been established?*

Interpretation of the correlation between inputs pair in the DRAMA network is straightforward. It consists of looking at the values of the connection parameters between active units of each input.

- *Lookup Speed: How quickly can a response be formed, i.e., what is the computational complexity of a lookup?*

The retrieval algorithm uses one time step computation. Thus, the response of the system is immediate.

- *Scaling: How well does the learning approach scale when increasing the dimensionality of the problem?*

If increasing the dimensionality of the problem consists of increasing the number of features of the input patterns and consequently the number of units of the network (each unit encoding for a feature), then the learning algorithm is the same as with a lower number of network units. Scaling up of the network size and thus of the learning approach is limited only by the capacity of the system used for the computation of storing the required number of global variables. Note that the number of features which can be encoded by a given number of network units is limited, see evaluation of the network capacity section 5.1.2.

- *Real time learning: How quickly can the learning box extract the relevant information from new experiences and incorporate it into its representation?*

Relevant information, that is the spatial and temporal aspect of the training pattern, is immediately incorporated into the network representation through the update of the time parameter and confidence factor parameters of the network connections.

- *Statistical Measures: Is it possible to make qualitative and quantitative, usually statistical statements, about the quality of what has been learned?*

The learning principle of the DRAMA architecture is based on statistical theory, similarly to other Hebbian networks; that is, learning results from process of statistical elimination among all possible input-output pairs, where the most likely pairs, i.e. the most frequently observed, are chosen. The quality of the learning, that is record of the correct correlation between units, can be evaluated quantitatively and qualitatively by comparing the parameter values of the connection between correlated and uncorrelated units. Qualitative statement about the the fact that two units have been correlated can be made by observing no zero value for the parameters of the connection linking these units. Quantitative statement about the correctness of the learning can be made by evaluating the ratio of parameters values between correctly and incorrectly correlated units. Correct learning has occurred when the ratio is greater than the threshold used by the retrieval function (see section 7.1.3).

## Chapter 6

# Overview of the robotic experiments

The DRAMA architecture described in chapter 4 provides a general control architecture framework for an autonomous robot. It allows on-line learning of spatio-temporal regularities across the multiple sensor-actuator modalities of the robot; that is, learning of time series of sensor-actuator, actuator-sensor, sensor-sensor and actuator-actuator inputs. It provides control of the robot's behaviour through retrieval of learned or predefined sensor-actuator sequences. Basic behaviour can be determined by fixing the connections of the DRAMA network between specific sensor and actuator systems of the robot. The DRAMA architecture is general in the sense that its structure and functioning make no assumptions on the type of robots, i.e. the robot's sensors, actuators and body structure, which should be used.

In the next five chapters (chapters 7 to 11), we report on the implementation of DRAMA in different robotic experiments carried out in simulated and physical environments. We use three types of physical robots: a doll-shaped robot and two pairs of wheeled vehicles, whose structure is made of LEGO (for one pair) and of Fischer-Technik (for the other pair). The shape and sensors of each of these robots are different. Experiments with the wheel-based vehicles are done in different physical environments: a hilly area (section 7.1.2), a series of corridors (section 7.2.1 and chapter 10) and an office environment (section 9.3).

In the experiments, the DRAMA architecture is used to dynamically control the robot's behaviour and learning. At each time step (one time step equals one processing cycle),

the robot's behaviour is determined by retrieving the motor outputs given the current sensor inputs. Basic behaviours of obstacle avoidance and phototaxis are predefined by setting the parameter values of connections between specific sensor-motor units of the DRAMA network. Learning is performed continuously by updating the parameters of the DRAMA network's connections following variations measured in the robot's sensor-motor state.

The first section of this chapter gives a detailed description of the robots' hardware and the simulator software implementation, which we use in experiments of chapters 7 to 10. The hardware of the doll robot, called *Robota*, which is used in experiments of chapter 11 is described in section 11.1. The second section describes the processing of the robot's sensors and actuators through the DRAMA architecture. The third section briefly summarises the content of each experiment, which are reported in the next chapters.

## 6.1 Hardware and Software description

We describe here the hardware of the LEGO wheel-based robots and the simulator software, which are used in experiments reported in chapters 7 to 10 (the exceptions are the experiments reported in sections 7.1.2 and 9.1, which use FischerTechnik robots; these robots are described in [Billard & Dautenhahn 97a]).

### 6.1.1 The robots

We use two autonomous LEGO robots, which act respectively as teacher and learner in the experiments. Each robot is equipped with one frontal infra-red sensor and one frontal bumper (sensitive to right/left direction of bumps) to detect obstacles. They have two sets of two light detectors. One set is used for the two robots to detect each other (and to follow each other by phototaxis, see section 7.1.1); the detectors are placed on the front of the learner robot and on the back of the teacher robot. Each robot carries a bright halogen bulb, which they can detect within 40 cm distance. The second set of light detectors is located underneath the robots' bodies, together with a small light bulb. This set-up is used to detect different ground surfaces which result

in a variation of the light bulb reflectivity.

Each robot is provided with a compass, made of two earth magnetic field detectors (product of Speake & Co Ltd.) placed on a plane at an angle of 90 degrees to one another. The compass has a maximal sensitivity of 3 to 5 degrees. In the experiment, it is used to distinguish between bearings of 22.5 degrees. The robots have a radio transceiver (emitter and receiver), made of a low power UHF data transceiver module (product of Radiometrix Ltd.). It is used to transmit signals between the teacher robot or the human instructor and the learner robot (i.e. to transmit teaching signals) and from the learner robot to a radio transceiver on the bench (i.e. to collect data). The radio transmission is based on a protocol developed by Alexander Colquhoun [Colquhoun 96]. Each transmission is composed of two bytes. In the experiments, the first byte is used to distinguish between transmission from the teacher robot and the learner robot; these are recorded by a radio receiver placed on the bench, in order to evaluate the progress of the teaching. The second byte is used to encode the signal sent by the teacher robot to the learner robot, which represents the word the learner robot has to associate with other of its sensor measurements (which stand for the meaning of the word). The signals are composed of a 1-byte radio signal with only 1-bit activated, i.e. orthogonal encoding. For instance, the two signals standing for "North" and "Stop" in the experiments of section 7.1.3 are encoded as follows: 'North' = (10000000), 'Stop' = (00000001). In our agents' 'language' 8-bits can encode 8 'words'.

As the DRAMA architecture capacity decreases in the face of overlapping data (see section 5.1.1), we chose to use an orthogonal encoding for all sensor measurements apart from those giving the motor activity (see section 6.2 for a description of the motor activity encoding). The sensor measurements of each sensor are sampled, such that each bit of the bit-string which represents the sensor measurement corresponds to a specific zone of sensitivity (see sections 4.3.3 and 6.2 for a description of the sensor encoding). The range and sensitivity of all the sensors used by each robot are given in the table 6.1, together with the length of the bit-strings used to represent each sensor information. The robot controller is made of a micro-controller with 512k byte EPROM space and 128k byte Static RAM. The CPU (central Processing Unit) is a

Phillips 93C100 series 68000 compatible running at 30 MHz.

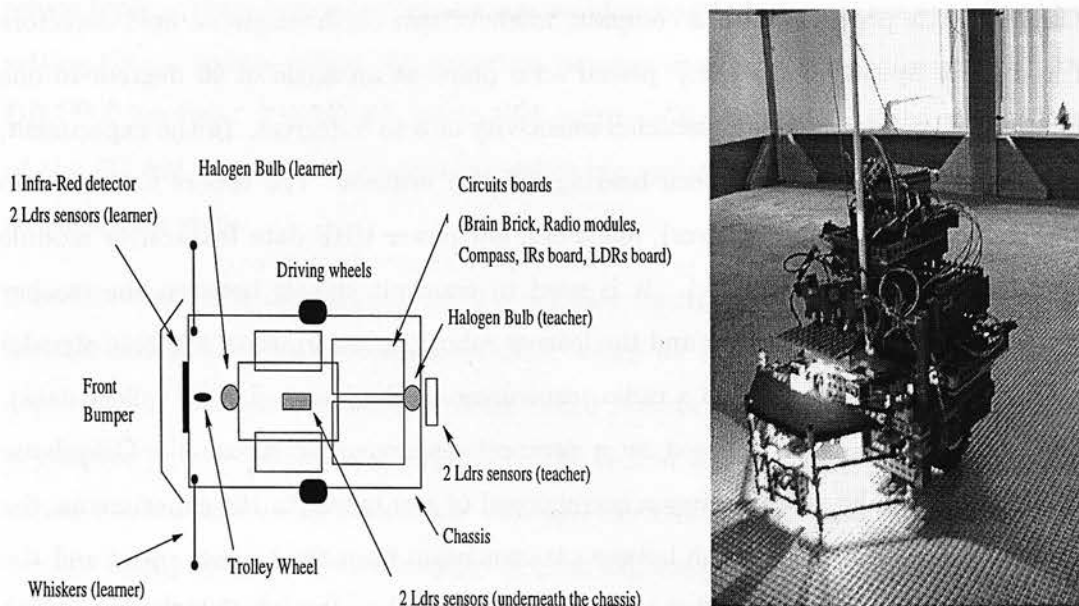


Figure 6.1: **Top:** schema of the LEGO robots' structure. **Bottom:** picture of the learner robot inside the Dodgem cage.

Table 6.1: Range and sensitivity of the robots' sensors

Sensor type	Physical Exp.		Simulations	
	Sensitivity	Encoding	Sensitivity	Encoding
Bumpers	0cm (touch contact)	active/inactive (1 bit)	None	-
Whiskers	0-15cm lateral	active/inactive (1 bit)	None	-
Infra-red	15 <sup>0</sup> and 0-40cm	2 levels (2 bits)	180 <sup>0</sup> and 0-45cm	8 quadrants/bits
Light front	180 <sup>0</sup> and 0 – 10 <sup>5</sup> Lux	2 levels/sensor (4 bits)	180 <sup>0</sup> and 0-45cm	8 quadrants/bits
Colours	None	-	3 types	3 bits
Light/Incl.	0 – 10 <sup>5</sup> Lux	2 levels (2 bits)	3 levels	3 bits
Compass	bearings of 3 – 5 <sup>0</sup>	16 bits	bearings of 22.5 <sup>0</sup>	16 bits
Energy	None	-	14 levels	14 bits
Radio	418 MHz, whole arena	8 bits	whole arena	3 to 31 bits

The experiments are carried out inside a rectangular cage of 2.5m by 2m by 0.5m, in which the robots are continuously recharged similarly to the system used in the 'Dodgem' (bumping cars) game. The cage is made of a wood structure, with roof and ground made of an aluminium grid. Roof and bottom of the arena are electrified, creating a potential difference of 10V between them. The robots carry a long stick touching both ends of the cage from which they receive the current to power their

battery and light bulb. In figure 6.1, we show a schema of the robots' structure and a picture of the learner robot inside the Dodgem cage.

We estimate that about 10 to 20% of the sensor measurements are noisy: 80% of the radio transmissions are correctly received, i.e. if a signal is received, then it is received perfectly. The noise corresponds to the case in which an emitted signal has not been received. The quadrants given by the compass are correctly detected in about 90% of the cases, incorrect readings being due either to a non homogeneity of the magnetic field in the room, resulting from magnetic emission of the electrical machines in the laboratory, or due to undesirable magnetic effects produced by the robot's motors and the powering of the Dodgem cage. The above mentioned value of uncertainty of the sensor measurements was evaluated by recording these sensor measurements over 1 hour, allowing the robot to wander freely in the environment. A radio signal was sent every 2 seconds from the bench. The noise ratio is equal to the percentage of received messages during this period. The detection of lighting direction was imprecise, as it was very sensitive to variation of external lighting in the room (which is frequent in Scotland!). The whiskers were observed to flicker (i.e. be activated), while not in contact with an obstacle, as a result of the robot's sharp change of direction or acceleration.

In order to improve the robot's light detectors and compass sensitivity we implemented a mechanism to let the robot turn on the spot four times at the beginning of each experiment, in order to measure the mean value of light intensity (ambient light for frontal detectors and ground reflexivity for the detectors underneath the chassis) and of the maximal and minimal components of the magnetic field. From its estimation of the light intensity variation across the room (on averaging the variation of its sampling over a 360 degrees rotation), the robot would determine the threshold above which the light intensity is produced by the lamp of the second robot. The robot would sample its compass measurements into 16 equal size zones of sensitivity, delimited by the measured mean values of maximal and minimal compass responses, which correspond to the maximal and minimal components of magnetic field.

### 6.1.2 The simulator

Simulation studies are carried out in a 2-D simulator, whose graphical representation is done using the MATLAB environment. The simulated environment is a simplified representation of the physical environment and consists of a rectangular arena whose size and content change for each experiment. The number of robots interacting in the arena varies also from one experiment to another, from two robots (chapter 7) to nine robots (chapter 8).

The simulated robots are provided with the same sensor systems as the real robots (apart from not having bumpers): light detectors for mutual recognition, infra red sensors for obstacle avoidance, a radio transducer as the means of communication and a compass to measure bearings of 22.5 degrees. Infra-red and light detectors are defined as a cone of vision of 180 degrees, which is segmented into eight quadrants. The measurement of the sensor is given by an 8-bit string where each bit corresponds to the value measured in each of the eight quadrants (e.g. infra-red=(11000000) stands for an infra-red activation of the first two quadrants). The range of sensitivities of the sensors of the simulated robots is given in table 6.1.

In the simulations, the behaviour of each robot is calculated separately, using the same code as on the physical robots. Code is written in C and is processed serially. The code of each experiment is stored in the DAI archives for further reference (see /hame/audeb/audebdemo/ directory restricted to internal use; see the DAI for getting access to it). The control of the robot's actions and learning through the DRAMA architecture is described in section 6.2. The robots' sensor perceptions are simulated by defining a specific measurement routine for each sensor. In order to reproduce reality more faithfully, the simulated robots are provided with sensors showing a similar range of sensitivity as those of the physical robots. For instance, similarly to the physical robots, the sensitivity of their compass allows to distinguish only between bearings of 22.5 degrees. Similarly, the sensitivity of their light detectors is limited, which results in imprecise alignment of the robots one behind the other when they follow each other by means of phototaxis (see section 7.1.1 for explanation of the following routine). The algorithm for the code of each experiment is given at the beginning of the main C files,

which are stored in the DAI archives.

## Note

All the electronic components, apart from the manufactured ones mentioned above in the text, were built at the department of Artificial Intelligence (DAI) at the University of Edinburgh. The Dodgem cage and the compass sensors were designed by the author and built by the mechanical and electrical workshops of the DAI. The LEGO structure of the robots have been built by the author with the help of Auke Jan Ijspeert. The simulator environment was written by the author.

## 6.2 DRAMA processing of the robot's sensor and actuator information

This section describes the processing of the robot's sensor information through the DRAMA architecture which leads to the control of the robot's actions and of its learning. We assume that the reader has read chapter 4 and, thus, understands the functioning of the DRAMA architecture.

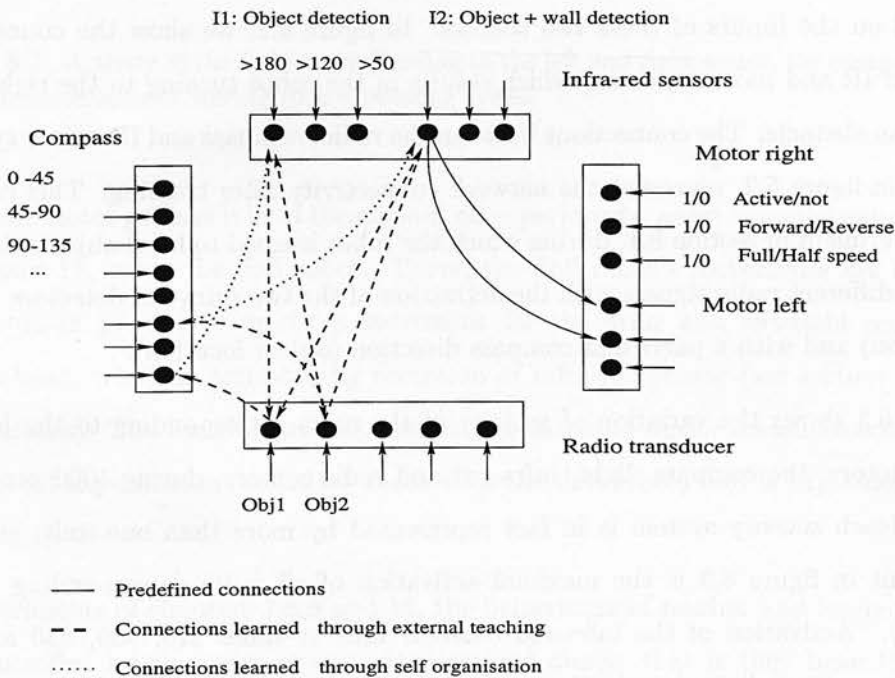


Figure 6.2: DRAMA connections state in the experiments.

Figure 6.2 gives a schematic representation of the DRAMA architecture as used in the experiments. We restrict the schema to depicting 4 sensor systems (radio, motors, compass, infra-red sensor) for reasons of clarity of the picture. In each experiment, the number of sensors used by the DRAMA architecture differs depending on the need of the particular experiment. At the beginning of each experiment report, the exact number and type of sensors is mentioned.

At each processing cycle, the DRAMA network output to the robot's motors is calculated, in order to determine the motor activity defined by the activity of the DRAMA motor units. The motor activity is encoded in a 3-bit string. Bit 1 determines the state of activity of the motor (*active/not* = 1/0), bit 2 encodes for the direction (*forward/reverse* = 1/0) while the third bit determines the speed (*full/half* = 1/0). Basic behaviours, such as obstacle avoidance and mutual following of the robots (phototaxis with light detectors) are predefined by setting the connection parameters (namely the confidence factors and time parameters) between the infra-red (IR) and light detector systems and the motor system. In order to perform a purely reactive behaviour, the thresholds of the event detector modules of the IR and light sensor systems were set to zero. Thus, the motor activity results from the threshold-based retrieval mechanism applied on the inputs of these two sensors. In figure 6.2, we show the connectivity between IR and motor systems, which results in the robot turning to the right when facing an obstacle. The connections between the radio, compass and IR sensor systems, shown in figure 6.2, represent the network connectivity after training. This refers to the experiment of section 8.1, during which the robot learned to label objects, by associating different radio signals with the activation of the two infra-red detectors (object detection) and with a particular compass direction (object location).

Figure 6.3 shows the variation of activity of the units corresponding to the left and right motors, the compass, light, infra-red and radio sensors, during 1000 processing cycles (each sensory system is in fact represented by more than one unit; what we represent in figure 6.3 is the maximal activation of all units corresponding to this system). Activation of the infra-red detector unit at times 210, 350, 430 and 780 produce an immediate deactivation of the right motor. The robot turns to the right when it faces an obstacle, as was predefined by setting up the values of the connection

parameters. As a result of the robot's rotation, a new value for the compass is measured at time 450. Light detection (which corresponds to detecting the second robot) at time 380 and 500, produces a deactivation of left and right motors alternatively. As a result, the robot aligns behind the other robot.

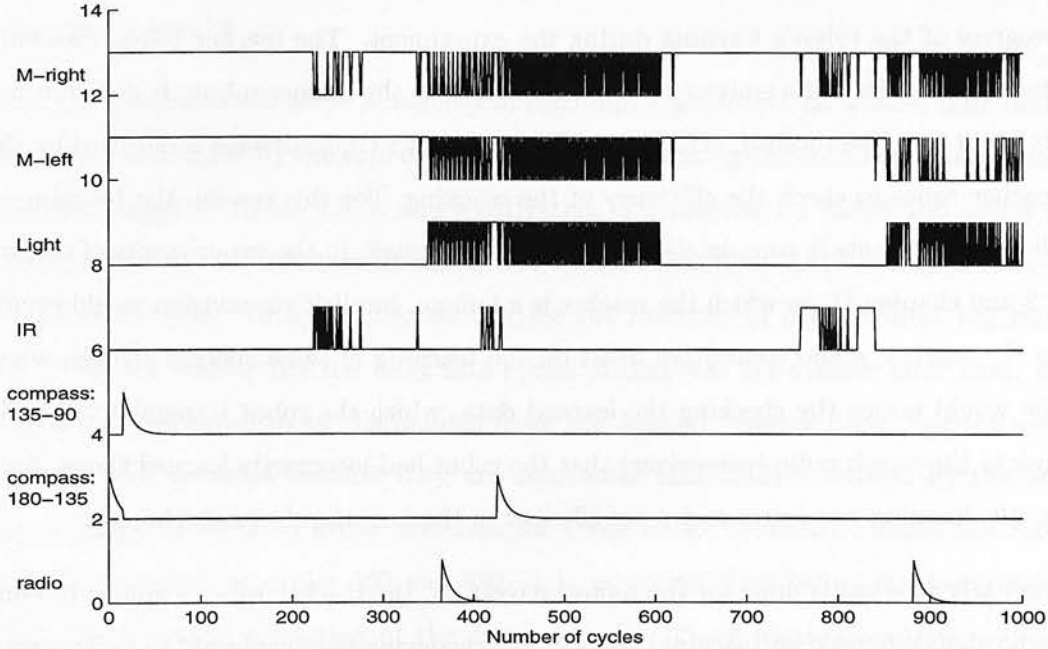


Figure 6.3: Activity of the units corresponding to the left and right motor, the compass, light and inclination sensors, during 1000 processing cycles.

The same motor process is used throughout all experiments, apart from the experiments of chapter 11, using the doll robot. There, the doll robot's movements are directed by a built-in process of up-down movement for the arms and left-right movement for the head, which is activated by reception of infra-red sensor (see section 11.2 for explanations). The learning process, that is association of sensor measurements, is the same in all experiments (including those with the doll robot) and is explained in the following.

In experiments of chapters 7,8,9 and 10, the behaviours of teacher and learner robots are controlled by the same process as described above, that is they have the same predefined basic behaviours. However, no learning mechanism is used for the teacher and its knowledge of the vocabulary is defined by setting the connections between

the radio sensor (words are radio signals) and other sensors, which correspond to the sensor measurement which the words describe. The learner robot uses both the input and the output of the radio module to receive the teacher's signals and to emit its answer (which corresponds to its retrieval of the learned radio signal given its current sensor measurements), which is recorded by the experimenter in order to evaluate the progress of the robot's learning during the experiment. The teacher robot uses only the output of its radio emitter to send the signals to the learner robot. It does not use its input from the receiver. That is, the learner robot's transmissions is not used by the teacher robot to check the efficiency of the teaching. For this reason, the teaching in these experiments is completely unsupervised. However, in the experiments of section 7.2 and chapter 11, in which the teacher is a human, implicit supervision would occur, as the teacher would sometimes insist on the teaching of some specific signals, when she would notice (by checking the learned data, which the robot transmits via radio link to the bench radio transceiver) that the robot had incorrectly learned those. As a result, learning was faster and more efficient in the human-robot experiments.

Similarly to what is done for the motor activation, the teacher robot's ability to emit radio signals (speaking/teaching) results from retrieving the output of the radio sensor system, given the robot's current sensor-motor state. The teacher 'speaks' only when it sees the learner. The inhibition of the activation of the radio output is obtained by giving a very high value to the confidence factor of the connections between the light detector units (learner recognition) and the radio output units. The input to this connection is 1 as long as the learner is not in view, otherwise zero. Therefore, when the learner is not in view, the activation of the light units wins the competition of activation (because of its very high confidence factor value), inhibiting activation from other sensor units. As a result, all the radio units are activated, which by definition would produce no output.

Learning occurs when one unit in one sensor system is newly activated, that is when the output of the event detector associated to this sensor system is activated. The time parameters and confidence factors of the network's connections linking previously or simultaneously activated units to the newly activated unit are then updated following equations 4.4 and 4.5 given in chapter 4. The time parameters give a measure of

the mean time delay between consecutive activation of the two units, while the confidence factors record the frequency of co-activation of the two units. Note that, in the experiments presented in chapters 7 and 8, we do not analyse the values of the time parameters when we determine the success of the learning. These parameters are mainly useful for the recording of sequences and are used in the experiments of chapters 10 and 11.

The DRAMA network keeps a memory of each unit's activation for a fixed time delay, which is determined by the rate of activation decrease along the unit's self-connection. In the example of figure 6.3, a unit's activation is conserved for about 100 cycles by the effect of the recurrent connections. The level of activity decreases by a ratio of 0.9 at each cycle. In figure 6.3, we can see the decrease of activation of the radio and compass units (only the forty first cycles of decrease are visible; after that, the values are too small to be distinguish from the X-axis). Motor, light and infra-red units do not decrease because they are constantly maximally activated by the new input (since there is no event detection for these sensor systems). When the radio unit is activated, at cycles 380 and 890, it is associated just before its deactivation with the following activation of the compass unit. These bidirectional associations between simultaneous and sequential sensory activations lead to the learning of the vocabulary, where radio bit-strings are associated with particular sensor activities. Simultaneously, other associations occur between other sensor systems, which represent physical regularities of the environment (e.g. associating the objects' features with the objects' locations, which are given in polar coordinates in terms of compass and energy measurements, as in section 8.1). Figure 6.2 shows an example of connections learned after training of the network, that associate infra-red detector measurements (object recognition) to two different compass measurements (location) and to two different radio measurements (name), as described earlier in this section.

### 6.3 Summary of the experiments

We carried out several sets of experiments, in which we investigated different characteristics of the learning architecture. The experiments are reported in chapters 7 to 11, each chapter presenting a different subject of study. The order of the chapters

correspond to both the chronological order of the experiments and to the increase in the complexity of the study.

In chapters 7 and 8, we evaluate the network's capacity at making spatial association across the the sensor-actuator state space of the robot. We first study (chapter 7) the transmission of a vocabulary, i.e. a set of labels for categorising the robot's perceptions, from a teacher robot to a learner robot and then from a human instructor to a robot. In chapter 8, we investigate, through simulations, scaling up of the transmission of the vocabulary to a group of robots. In chapter 9, we present three experiments, which implement different algorithms for using the robot's understanding of the vocabulary to command and to further teach the robot. In chapters 10 and 11, we evaluate the network's ability for making spatio-temporal association and for learning time series of the robot's perceptions. These experiments are carried out using a vehicle robot (chapter 10) and a doll-shaped robot (chapter 11). We here give a brief overview of these experiments.

**Chapter 7: Grounding of a robot's perceptions and actions:** An autonomous mobile robot is taught a vocabulary to describe its internal perceptions (inclination, orientation, movement) and external perceptions (object detection with light or infra-red sensors). The robot associates radio signals (words), emitted by a teacher agent, another robot (section 7.1) or a human instructor (section 7.2). Teaching occurs as part of an imitative strategy, namely the learner robot follows the teacher robot. Figure 6.4 illustrates these experiments. On the left, we see the learner robot following the teacher robot, which climbs up a hill. The teacher robot sends a radio signal to label the inclination perception of going up. On the right, we see the human instructor carrying a lamp, with which she guides the robot and a radio emitter, with which she sends different signals to label the robot's different perceptions of orientation. Experiments are carried out in simulated and physical environments. In section 7.1.1, we report on two experiments in which we use a simpler version of the learning architecture, as compared to DRAMA. This first architecture was a preliminary version of DRAMA, the full version of which was designed to improve some of the limitations of the first architecture, which were pointed out by the results of these first two experiments.

All experiments of chapters 7 to 11 apart from those of sections 7.1.1 and 9.1 use the complete DRAMA architecture described in chapter 4.

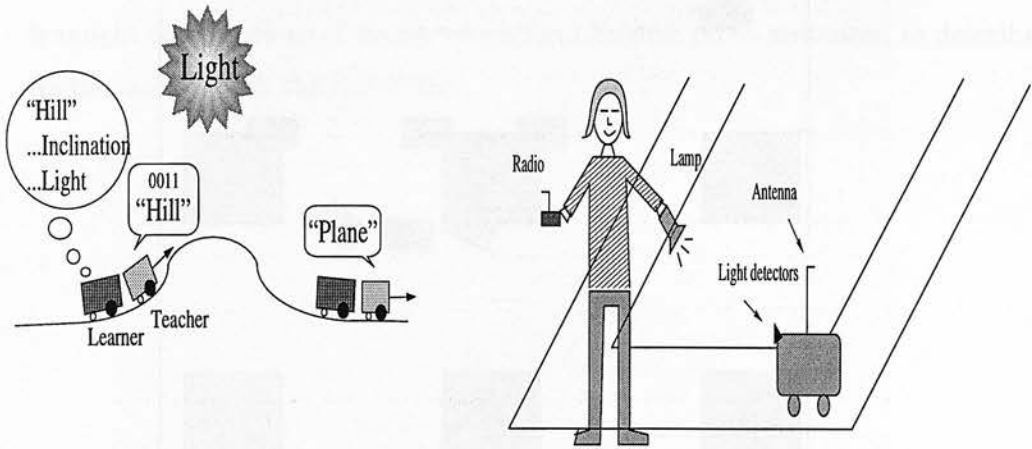


Figure 6.4: Experiments of chapter 7: transmission of a vocabulary from a teacher robot (left) or human instructor (right) to a learner robot.

**Chapter 8: Grounding of perceptions of a group of robots:** A group of nine robots wander in an environment, which contains nine hills (big coloured patches in figure 6.5 left). There is one teacher robot, in red, and eight learner robots. The teacher teaches a vocabulary to label each of the nine hills. We study the conditions under which each learner robot can become teacher when sufficiently confident in its learning. In addition, we investigate the benefit of the imitative strategy on the success of the learning.

**Chapter 9: Using the robot's communicative skills:** In three case-study experiments, the robot uses its understanding of the vocabulary learned in three different experiments of chapters 7 and 8 in order (1) to find another robot, when this one is telling its location: being on the hill or on the plane (this follows the experiment illustrated in figure 6.4 left); (2) to learn the location of objects, when it hears another robot saying it (this follows the experiment illustrated in figure 6.5); (3) to obey a human instructor, who send it radio signals to direct its travel across an office environment (this follows the experiment illustrated in figure 6.4 right). Experiments are carried out with physical robots for the first

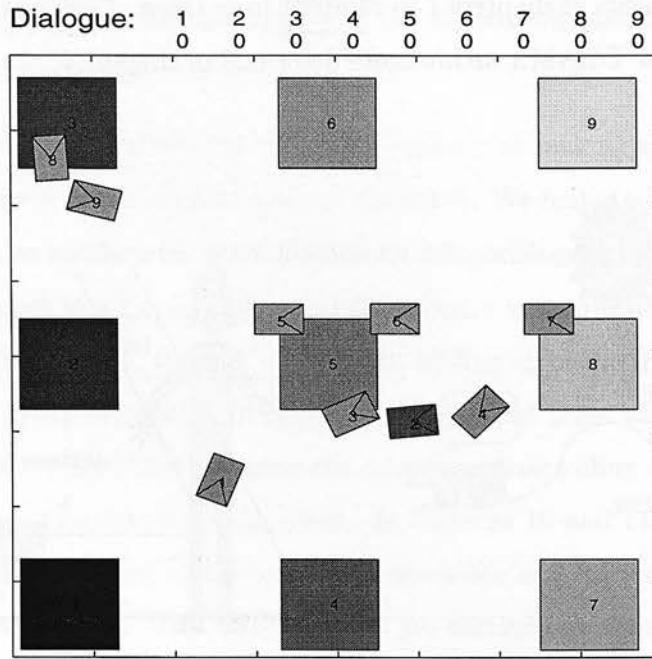


Figure 6.5: Experiment of chapter 8: Transmission of a vocabulary in a group of robot.

and third one and in simulation for the second.

The experiment of section 9.1 uses a simpler version of the learning architecture as compared to DRAMA. This experiment follows from experiments of section 7.1.1 and uses the same architecture as those experiments.

**Chapter 10: Grounding of the robot's sequence of perceptions:** An autonomous robot learns time series of perceptions, when travelling in a series of three corridors. Similarly to experiment of chapter 7, the learner robot follows a teacher robot, which emits radio signals each time it passes from a corridor to the other (in order to label its orientation). The learner robot learns the timing and ordering in which compass, radio and light (for the aluminium plate lying in the second corridor) measurements occur as a result of its regular wandering in the series of corridors and of the regular radio signal emission of the teacher robot. Experiments are carried out both in simulation and in physical set-up. Figure 6.6 top shows a picture of the two robots when engaging in the second corridor, in front of the aluminium plate.

**Chapter 11: Experiment with Robota, the doll robot:** An autonomous doll-

shaped robot, called Robota is taught by a human instructor to perform and label different sequences of actions (dance patterns). Teaching occurs through an imitation game, whereby the robot mirrors the instructor arm and head movements (see illustration of figure 6.6 bottom). In a second experiment, the robot is taught combinations of words, which form English proto-sentences, to describe its interaction with the instructor.



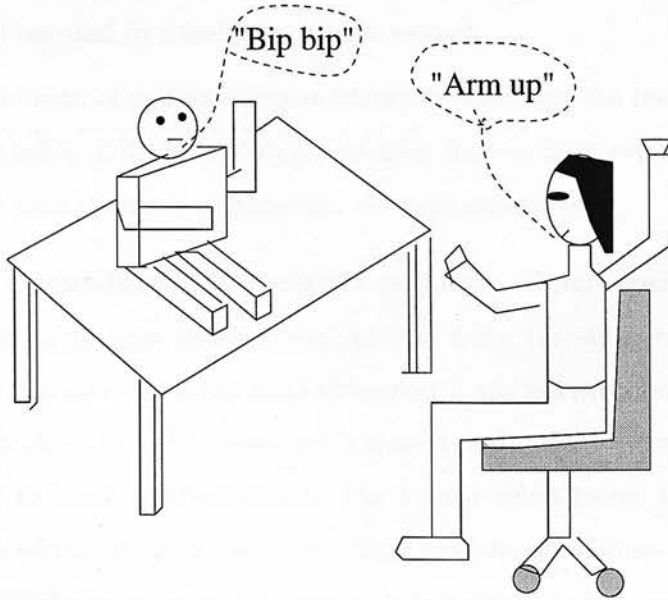
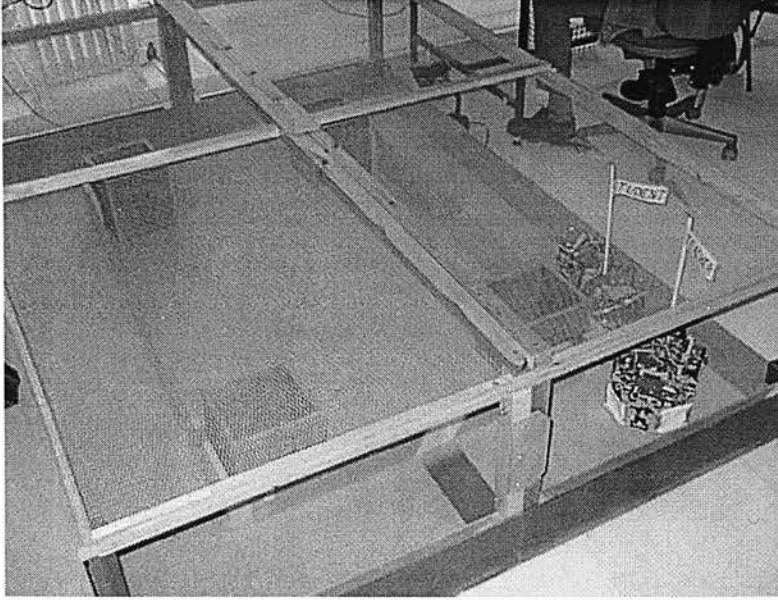


Figure 6.6: **Top:** Experiment of chapter 10: teacher and learner robots travel in a series of three corridors. **Bottom:** Experiment of chapter 11: a human instructor teaches Robota, the doll robot, a vocabulary to label sequences of arm and head movements.

## Chapter 7

# Grounding of a robot's perceptions and actions

In this chapter, we report on the implementation of the DRAMA architecture in experiments in which an autonomous mobile robot is taught a vocabulary to describe its perceptions and actions. The robot grounds the meaning of signals given by an instructor, another robot or a human, onto different states of its sensors and actuators. These experiments intend to evaluate the network's ability for making spatial association across the sensor-actuator state space of the robot. Experiments are carried out both in simulated and physical environments. We report on two sets of experiments.

In the first set (section 7.1), a learner robot is taught by a teacher robot. Three experiments are carried out. In the first experiment, the learner robot is taught a vocabulary to describe its movements. In the second experiment, the robot is taught a vocabulary to describe different perceptions of inclination. These two experiments use a simpler version of the learning architecture, as compared to DRAMA. This first architecture was a preliminary version of DRAMA, the full version of which was designed to improve some of the limitations of the first architecture, which were pointed out by the results of these first two experiments. In the third experiment, the robot is controlled by the DRAMA architecture and it is taught a vocabulary to describe its movements and its orientation. In the second set of experiments (section 7.2), the learner robot is taught by a human instructor. Two experiments are carried out. In the first experiment, the robot is taught a vocabulary to describe its orientation relative to four quadrants of a compass. In the second, it is taught a vocabulary to describe

objects in its environment. In section 7.3, we discuss and summarise the main findings of these experiments.

## 7.1 Transmitting a vocabulary from a teacher robot to a learner robot

This section reports on three experiments in which a learner robot is taught by a teacher robot. Teaching occurs as part of a teacher-learner scenario based on an imitative strategy, namely following by mutual phototaxis of the two agents. The same scenario is used in experiments of chapters 8 and 9. In this section, we describe first the teaching scenario and then report on each experiment separately. The first two experiments are described together in section 7.1.2, as we use for both of them a simpler version of the learning architecture (as compared to DRAMA). We then report in section 7.1.3 on a reproduction of the first experiment of section 7.1.2, while using the DRAMA architecture and compare the results of both experiments. The experiments of section 7.1.3 are carried out using both physical and simulated environments.

### 7.1.1 Imitative strategy

Transmission of the vocabulary occurs as part of an implicit imitation of the teacher robot's movements by the learner robot, which results from the learner following the teacher. Following proceeds from mutual phototaxis of the two agents. Each robot detects the other by tracking a light attached to the other robot. The light bulbs and detectors are placed on the front of the learner robot and on the back of the teacher robot. Thus, the teacher robot is always leading, followed closely by the learner robot. Because tracking is mutual, it results in a smooth binding between the two agents. The agents seldom lose sight of each other, as if the learner runs slower the teacher waits for him (that is, the robot stops for a short delay when it does not see lights behind it any more), and they find each other easily as they are both looking for each other. The robots align one behind the other, by adjusting their position using the difference in light intensity between their two light detectors, which are placed at 45 degrees (see figure 6.1 left).

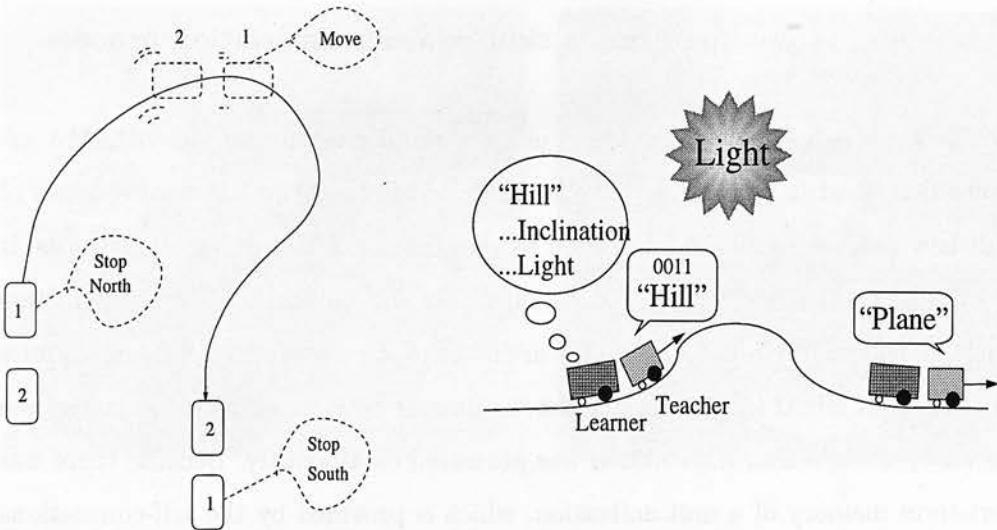


Figure 7.1: **Teaching Scenario:** A teacher robot (1) teaches a learner robot (2) a vocabulary to describe (left) its actions and orientation and (right) its inclination and light intensity measurements. While following the teacher, the learner robot grounds the teacher's 'words' onto its own observations.

While the two agents wander randomly in the environment, following each other, the teacher sends radio signals, i.e. 'words', to describe its external perceptions, e.g. light intensity measurements, or internal perceptions, e.g. orientation or inclination (see figure 7.1). The learner attaches a meaning to the teacher's signals in terms of its own perceptions; that is its own perception of light and its own measure of inclination or orientation. While they are bounded by the following process, learner and teacher agents are set in a position from which they share a common context of both external (facing the same direction) and internal (performing the same movement, travel the same distance and over the same ground) perceptions. This implicit similarity between the two agents' perceptions is what enables the learner to make sense of the teacher's words, as the teacher talks only of what it senses, unaware of the learner's actual perceptions. It is thus an unsupervised teaching strategy. Learning of word-observation pairs results from the statistical associative process provided by the DRAMA architecture (see section 4.3.5 for explanations), where incorrect associations, which arise from a mismatch between the agents' observations, are discarded compared to correct ones by a process of statistical elimination depending on their relative frequency of occurrence.

### 7.1.2 Symbol grounding using a simultaneous association process

Two preliminary experiments were done using a simpler version of the DRAMA architecture described in chapter 4. The learning architecture we use here consists of the Willshaw net, see section 4.3.1, added to the confidence factor. In other words, it corresponds to the DRAMA architecture using only the confidence factor parameters and without recurrent connections. This architecture has no event detector modules (which is used in DRAMA to segment the continuous flow of sensor information, see section 4.3.4), thus, sensor information was processed continuously. Because there was no short-term memory of a unit activation, which is provided by the self-connections on the units in the DRAMA architecture (see section 4.3.5), association is done only between simultaneously measured sensor information. A complete description of this architecture is given in [Billard 96].

The first experiment was carried out at VUB with LEGO robots at the department of Artificial Intelligence (DAI) at the University of Edinburgh. The second experiment was done with FischerTechnik robots at the Artificial Intelligence Laboratory at the Vrije Universiteit Brussel (VUB-AI). A complete report of these two experiments can be found in DAI technical reports [Billard 96] and [Billard & Dautenhahn 97b] respectively. Shorter reports of these experiments were published in conference proceedings [Billard & Hayes 97b, Billard & Dautenhahn 97a]. In this section, we summarise briefly these experiments and their results, in order to point out the advantages of DRAMA over the simpler architecture for improving the performance of the learning, which is demonstrated in the experiments of section 7.1.2.

#### Experimental set-up

The first experiment, carried out with LEGO robots, consisted of teaching the robot a vocabulary to describe its four actions of *moving forward*, *turning left*, *turning right* and *stopping*. This experiment was done at DAI, using a round arena, encircled by walls with no obstacles inside. In the second experiment, carried out with FischerTechnik robots, the learner robot was taught to distinguish between three inclination states, which corresponded to moving on the plane and climbing up and down a hill. The

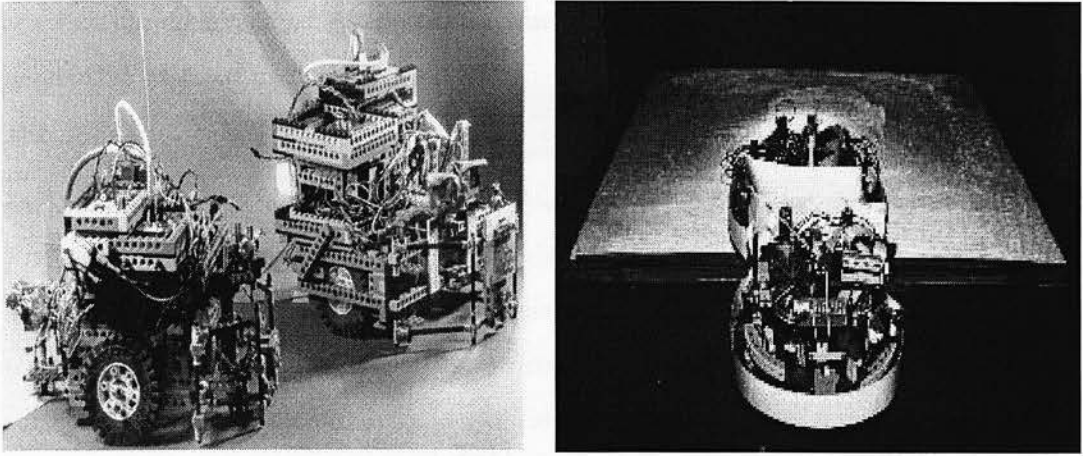


Figure 7.2: **Left:** Picture of the two LEGO robots. **Right:** Picture of the two FischerTechnik robots, beginning to climb the hill

environment, at the VUB-AI, consisted of a rectangular arena with a hill of 17 degrees inclination in the middle. In figure 7.2, we show two pictures of the teacher and learner robots in their respective environments.

### Results and discussion

Figure 7.3 shows the results of this experiment. The learning success of this experiment was measured in two ways: 1) we placed the robot in seven different positions on the hill (which are shown on the schema of figure 7.3 top right) and measured its responses (i.e. the robot activated a light pattern on 8 LEDs, representing the encoding of the radio signal it retrieved, given its current inclination measurement; 2) we recorded the values of the confidence factors for each network connection during and after training. Figure 7.3 top right shows the learning results for each of the seven control positions on the hill (X-axis), which correspond to seven different inclinations. The Y-axis in figure 7.3 shows the three taught radio signals (the numbers are not significant). Results show that the learner robot has associated well three radio words with three different zones of inclination. It distinguishes between zones of upwards ('up'), i.e. the three first control positions, downwards ('down'), i.e. the three last positions, and plane ('flat') inclination, i.e. the control position number four. However, the learner and teacher robots' inclination measures at positions 4 and 7 (which correspond to the top and bottom of the hill respectively), attached to the same radio signal are shifted from

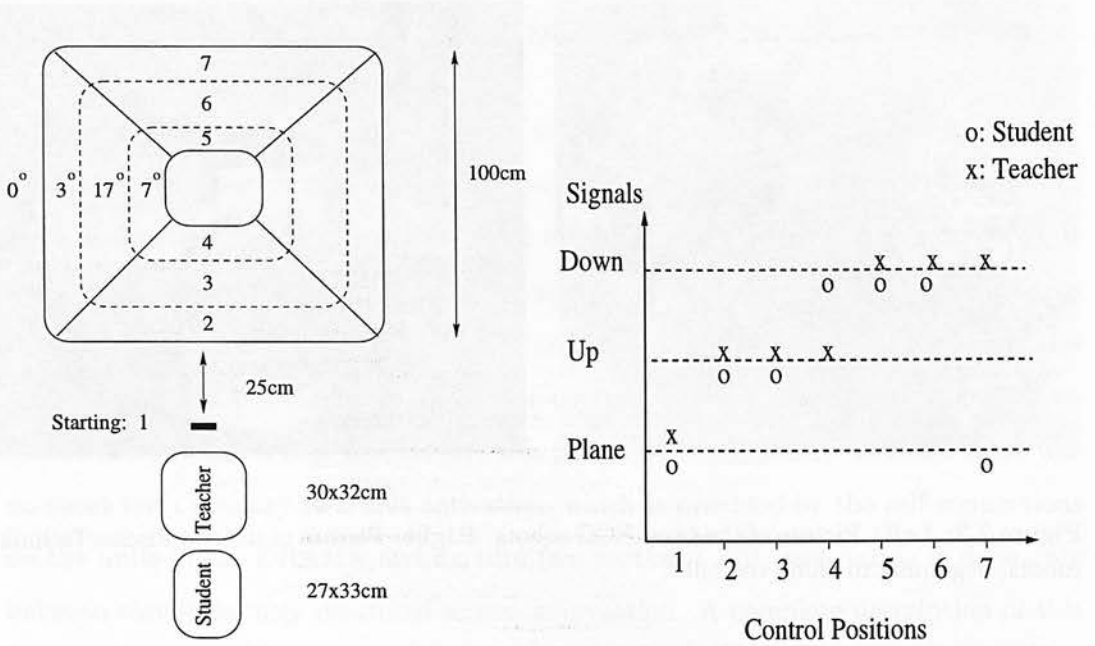


Figure 7.3: **Top left:** Schema of the hill and robots' dimensions and of the seven control positions with corresponding inclination value of the hill slope. **Top right:** Y-axis shows the robot's answer (light output standing for 'plane', 'up' or 'down' words) for each of the seven control positions. **Bottom:** Correlations between light sensor (Y-axis), inclination sensor (X-axis) and radio sensor (Colored areas on the plot). Z-axis gives the confidence factor values of connections between inclination and light sensors. The three colored areas (three levels of darkness) show the connections between radio sensor (one area for each of the words) and light and inclination sensors for which the confidence factor values are greater than the retrieval threshold.

one another, as expected given the delay and consequent dissimilarity in each robot's perception at these critical positions.

In this experiment, associations were made not only between radio and inclination sensor systems, but also between those systems and the light sensor system. Figure 7.3 bottom shows the result of these associations in a four dimensional graph. The Z axis represents the values of the confidence factors for all connections between light sensor units (Y-axis) and inclination sensor units (X-axis). The colours on the plot surface represent the connectivity (a confidence factor greater than zero) between radio and inclination sensor systems, one colour per radio signal. We see that each signal is associated with a distinct zone of inclination, which corresponds well to the notion meant by the signals, namely 'flat - down - up'. This result is equivalent to the association observed in figure 7.3 top right. In addition, we observe that a progressive increase of the light intensity had been associated with the increase of inclination in upwards direction. This is due to the fact that a light bulb had been mounted above the hill. When the robot was climbing up, it would then receive more light in its frontal light detector than when climbing down or running on the plane. These three dimensional associations between light, inclination and radio sensors will be used in the experiment of section 9.1 for directing the robot towards a dark and flat or bright and hilly area, depending on the radio signal it is sent.

In the two experiments (that of the DAI and of the VUB), results were successful to the extent that the learner robot well associated the different radio signals emitted by the teacher robot with different motor states and inclination measurements. However, the associations made by the learner differed from that of the teacher, as the learner did not associate the radio signals with the same motor states and inclination values as meant by the teacher. Because of the spatial displacement between the two robots, while following each other, the learner robot's perception of inclination and of actions would become similar to that of the teacher only after a delay, the time to reach the same spatial position (for the inclination) and the time to react to the observation of a change in the travel direction of the teacher (for the actions). Because, in these experiments, association was done only between the learner robot's measurements which occurred simultaneously to the teacher robot's radio emissions, correct learning relied on having

a longer period of time during which each robot would share similar perceptions (e.g., when the two robots are both turning or both climbing up the hill) than the period of time during which the two robots' perceptions would be dissimilar (which would correspond to the time required for the learner robot to reach the teacher's position). In this case, the number of correct associations, which corresponds to the period of time during which the robots share the same sensor/actuator measurements, would exceed the number of incorrect associations. Consequently, the confidence factors associated with the correctly activated connections of the network would be more significant than those of the incorrectly activated connections and would, therefore, win the threshold-based competition for output units' activation. In the experiments done at the DAI, in which the robot learned words for its actions, the overlap between these two critical periods of time was just sufficient. In 60% of the experiments, the robot learned correctly the four words (see [Billard 96]). However, the ratio between the number of correct and incorrect associations (measured by the ratio of confidence factors) was almost one for the words corresponding to *turning right* and *turning left*, that is, the robot had associated almost as often the incorrect word-action pairs as the correct one. In other words, these experiments had between 40-50% the noise, i.e. incorrect matchings, which made their chance of success very scarce. In the experiments done at the VUB-AI, the overlap between the two critical periods of time was not sufficient. As a result, the learner robot's mapping between inclination zone and radio signal was delayed from that of the teacher, since when the teacher robot was climbing up the hill the learner robot was still on the flat ground and when it was climbing down the learner robot was still climbing up (as it is the case in figure 7.2 right).

The failure or poor chance of success of these two sets of experiments lead us to the following conclusions: 1) the architecture should be improved, so as to allow association between sensor measurements delayed in time; this would then allow the learner robot to associate the teacher's signal with the correct perception which it would make only after a time period following the teacher's signal emission. 2) The robot's sensor information should be preprocessed, so as to reduce the number of associations with incorrect sensor measurements. For instance, in the DAI and VUB experiments, relevant measurements corresponded to the first change in the motor state or inclination measurement after the teacher's signal emission (like the passage from moving forward

to turning or from flat ground to hilly one).

Following the findings of this first set of experiments, we improved the learning architecture leading to the DRAMA architecture, described in chapters 4 and 5. By adding a preprocessing module to segment the robot's sensor measurements (the event detector modules) and a short-term memory of the sensor activation (the self connections on the network's units), we eliminated the problems related to continuous association of simultaneous measurements. In DRAMA, association is done between 'novel' sensor stimuli (which is a variation in a sensor measurement) which occurs inside a time period equal to the duration of short-term memory. In section 7.1.3, we report on experiments which replicated that of section 7.1.2, using the DRAMA architecture and showed an improvement of the results. The rest of this thesis describes experiments which validated several other properties of the DRAMA architecture.

### 7.1.3 Symbol grounding using sequence processing of sensory variation

We carried out experiments, in which the robot was taught an eight-word vocabulary to describe its actions of *moving*, *stopping*, *turning to the left/right* and its orientation relative to the four quadrants of a compass *North*, *South*, *East* *West*. These experiments were carried out in physical and simulated environments. In the following we describe first the experimental set-up and then present the experiments' results. These experiments were reported in [Billard & Hayes 98, Billard & Hayes 97a, Billard & Hayes 97c].

#### The experimental set-up

The set-up of the physical experiments consists of the Dodgem cage with the two LEGO robots described in section 6.1.1. The sensors which the physical and simulated robots are provided with in this experiment are the infra-red detector and frontal bumpers (only for the physical robots) for obstacle avoidance, the light detectors for mutual following by phototaxis and the compass. The sensitivity and range of these sensors are given in table 6.1. The encoding of the sensor and actuator information in the DRAMA network is as follow. There are 41 units ( $2 + 1 + 4 + 16 + 10 + 8 =$  Bumpers +

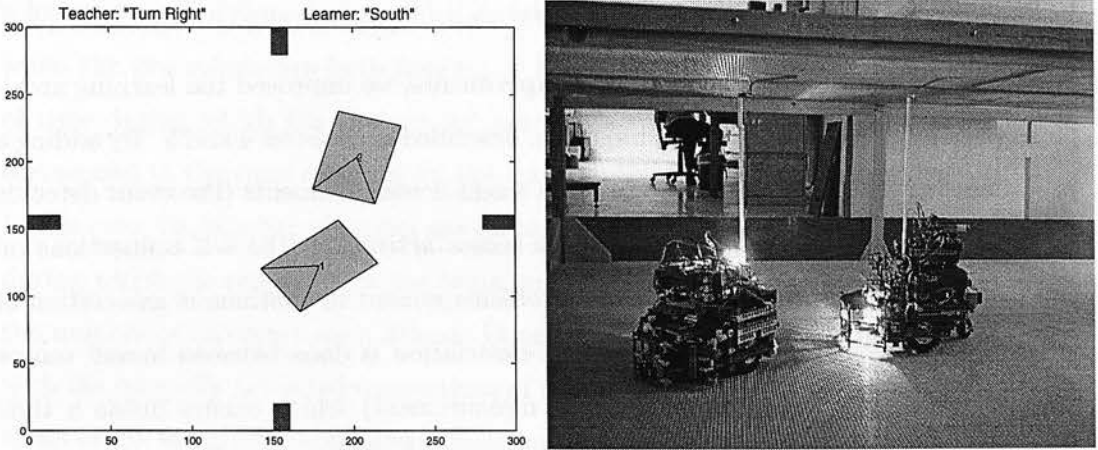


Figure 7.4: Simulated and physical environments in experiments on teaching the robot a vocabulary for its actions and orientation.

IR-sensor + Light-detectors + Compass-quadrants + Motor-states + Radio-sensor) in the network of the physical experiments. There are two units for each frontal bumper which become active after a bump. There is one unit for the frontal IR which becomes active when the IR detector measures a value greater than 150, which corresponds to detecting an obstacle 5 cm away. There are 4 units for the frontal light detector which correspond to two light levels (below or over a threshold fixed by the robot for distinguishing ambient light from the light of the second robot, see section 6.1.1 for explanations). There are 5 motor states per motor (stop - slow/fast forward/reverse). This encoding is fixed by the hardware specifications. There are 8 units for the radio sensors. A signal consists of one byte with only one unit active (orthogonal encoding of the signals). The measurements of the compass are divided in 16 zones and each unit corresponds to a measurement contained in the range defined by the zone. It is an orthogonal encoding, see the example of page 64. There can be only one unit of compass active at a time. Note that the same encoding as used for the compass is also used for the inclination and energy sensors' measurements in the simulated robots (inclination and energy sensors are used in other experiments of this chapter and following chapters but not in the present experiment).

There are 55 units ( $1+8+8+8+16+8+6 = \text{Default-speed} + \text{IRs} + \text{Light-detectors (front + back)} + \text{Compass-quadrants} + \text{Radio-sensor} + \text{Motor-states}$ ) in the simulation. The default unit is constantly active and is used to drive the robot's default

movement of going forward. The eight units of the IRs and light sensors correspond to the activation of these sensors in each of the eight quadrants (frontal or rear view). There can be more than one unit active at a time depending on the obstacles repartition around the robot (detected by IRs) and on the second robot's location relative to the observing robot (detected by light sensors). Radio signals are orthogonal with one unit active per signal. See page 115 for an example of radio encoding and an explanation of the encoding of the motor states. Figure 7.4 shows a graphical representation of the simulated environment and a picture of the physical one.

The short-term memory duration of unit activity is the same for all units and is equal to 300 processing cycles in both simulated and physical controllers.

The simulated environment consists of a rectangular arena measuring 300 by 300 units. There are 4 obstacles on the sides of the arena (dark rectangles in the figure), which represent the columns supporting the roof of the Dogem cage in the physical set-up. The robots are represented as rectangles of 30 by 20 units, a triangle depicting the front. In the figure, we see the teacher robot (in front) followed by the learner robot (This image was produced during a simulation, using MATLAB graphic environment). The teacher, which is turning, sends the radio signal corresponding to 'turning right'. The learner, when following the teacher, senses the signal 'move' (that is, it retrieves the radio signal for move) as it has not yet begun to turn. Once it will begin to turn, it will associate the teacher's signal with its new actuator state (turning right). It does not associate the teacher's signal with its current state (moving forward), as this state has not changed for longer than the memory duration and has thus been forgotten. When the motor state will change for the state 'turning right', it will exceed the threshold of the event detector of the motor system and will be transmitted to the DRAMA module for association. See section 4.3.4 for further explanations.

### **Results of simulations**

A set of 100 runs, each run simulating 10000 processing cycles of the robots (about 4 hours of real experiments) was carried out to test the speed and stability of the learning. The success or failure of learning in one experiment is determined by looking at the values of the confidence factors on the connections between the radio sensor system

and the rest of the sensor and actuator systems of the agent (light, infra-red, compass, motors). The experiment is said to be successful when only the connections between the correct radio unit, standing for the particular word, and its associated sensor-actuator (compass and motors) unit combination have a confidence factor value greater than or equal to half the maximal one for all connections leading to these units. In other words, learning of the vocabulary is a success if the number of correct associations (correct matching between radio signals and sensor-actuator measurements) is greater than the number of incorrect ones, i.e. if the noise (incorrect examples) is below 50% of the total number of examples in a run.

We measure the success of the learning at each time step by calculating the ratio between the parameters (confidence factors) of correctly and incorrectly activated connections. At each presentation of a new example (new association word-perception), the confidence factor parameters are updated. The study of the variation of the confidence factors values during the experiment informs us about the variation of the percentage of noise, incorrect word-perception association, and consequently of the stability of the learning. Noise, in our case an incorrect example, is produced when teacher and learner are not perfectly aligned and therefore measure different compass orientations or perform different movements.

Figure 7.5 left shows the variation of the confidence factor (cf) values during a run (mean value over all runs) for learning of the action words, which we compare to the results of the physical experiment (figure 7.5 right) for learning the same words, see discussion of the next section. The Y-axis is the ratio between the cf value associated with the correct word-perception connection (i.e. connection between the radio sensor unit and compass or motor units) and the maximal value of cf attached to all other connections between the word and other sensor measurements, i.e.  $\text{cf}(\text{correct correlations}) / (\text{cf}(\text{correct correlations}) + \max(\text{cf}(\text{incorrect correlations})))$ . Data are the mean values over all words and over all runs. A ratio greater than 0.5 means that the particular sensor measurement has been associated with the correct word more often than with other words, which is considered as successful learning. Figure 7.5 left shows that learning is robust, with only small fluctuations in the ratios, and it steadily increases as more teaching examples are given. Once the cf value has

exceeded the threshold of 0.5, i.e. the word has been learned, it stays larger than the incorrect threshold with a standard deviation of 0.23 (see table 7.1).

	<b>Simulations</b> Mean & Std	<b>Physic. exp</b> Mean & Std
Ratio of cf	0.62 ± 0.23	0.71 ± 0.18
Nm words/total	0.80 ± 0.19	0.96 ± 0.10
Nm examples	21 ± 31	70 ± 16

Table 7.1: Results of simulations and physical experiments. See text for legend.

The slow increase of the slope (figure 7.5 left) at its beginning is due to the fact that we superpose the learning curves of 100 different runs and that the learning does not start at the same point of time for each run. This depends on the time needed for the robots to find each other, as each run begins with the robot placed in a random location in the arena. The slope of each single curve is, however, very steep. In most cases, the first teaching would be a correct example. The curve would then start from the maximal value, i.e. 1, decreasing slightly afterwards due to the effect of noise. Note that the value of the ratio at time zero is always zero, as the teacher robot needs at least one processing cycle before retrieving the radio output, i.e. emitting a signal, which is then processed by the learner agent only at the following time step.

### Results of physical experiments

A set of five physical experiments was carried out. Each experiment consisted of two teaching phases. In the first phase, six words were taught (*stop, move, South, North, East, West*) and, in the second, 3 words (*stop, turn left, turn right*). Each experiment lasted for about 1 hour 30 minutes (45 minutes for each phase) and was stopped when about 90 teaching examples<sup>1</sup> had been done (the exact number of teaching examples is unknown because the radio transmission was imperfect and it was difficult to detect whether the radio signal had been received by the robot.).

Table 7.1 compares the results of simulated and physical experiments during the first phase of the learning. The measures in the table correspond to: 1) The ratio between the confidence factor value of the correct word and the maximal value of confidence

<sup>1</sup> A teaching example/episode corresponds to one transmission of a radio-signal by the teacher robot.

factor attached to all words for a given sensor measurement (mean value over all words and all runs),

i.e.  $\text{cf}(\text{correct correlations}) / (\text{cf}(\text{correct correlations}) + \max(\text{cf}(\text{incorrect correlations})))$ , same value as shown in figure 7.5. 2) The ratio between the number of words learned at the end of the run and the total vocabulary (mean value over all runs). 3) The number of teaching examples per run (mean value over the runs).

Results of simulations and physical experiments are qualitatively similar. The mean values of confidence factor for all words over each run are respectively 0.62 and 0.71. The standard deviation values represent the fluctuation of the cf values during the run. The qualitative similarity between physical and simulated results shows that the simulation studies gave a good account of the percentage of noise occurring in the physical experiments. To reiterate, noise corresponds to matches between the two robots' sensor perceptions, which result mainly from the imprecise following and alignment of the two robots. The number of words learned at the end of the runs is on average better in the physical simulation than in the simulations and there are on average three times more teaching examples during a physical run<sup>2</sup>. The reason for this difference might be due to the small number of physical experiments (poor statistics). But the main reason is surely that learning in the physical experiments was helped by placing the robots one behind the other one from the beginning<sup>3</sup>. The robots would then seldom lose each other during a run, which would maximise the number of teaching examples given during a run, consequently increasing the chances of correctly learning the words.

In figure 7.5 right we show the variation of the confidence factor values (mean value for all words and separate curves for each word) in the second phase of the experiments. As for the left graph, the Y axis represents the ratio between the confidence factor values of the correct word-measurement pairing and the maximal value of confidence factor values attached to all words for a given sensor measurement. Data are mean values over all words and over all runs. A ratio greater than 0.5 means that the

---

<sup>2</sup> The large value of standard deviation for the number of teaching examples in the simulations (line 3 of table 7.1) is due to outlier values of 300 (teaching examples per run) in the data.

<sup>3</sup> This was done in order to shorten the time of the experiments, the hardware being unlikely to be fully functional more than one hour of continuous running of the robots. The gears are made of plastic components and, therefore, tend to easily break when they have to endure a too long friction.

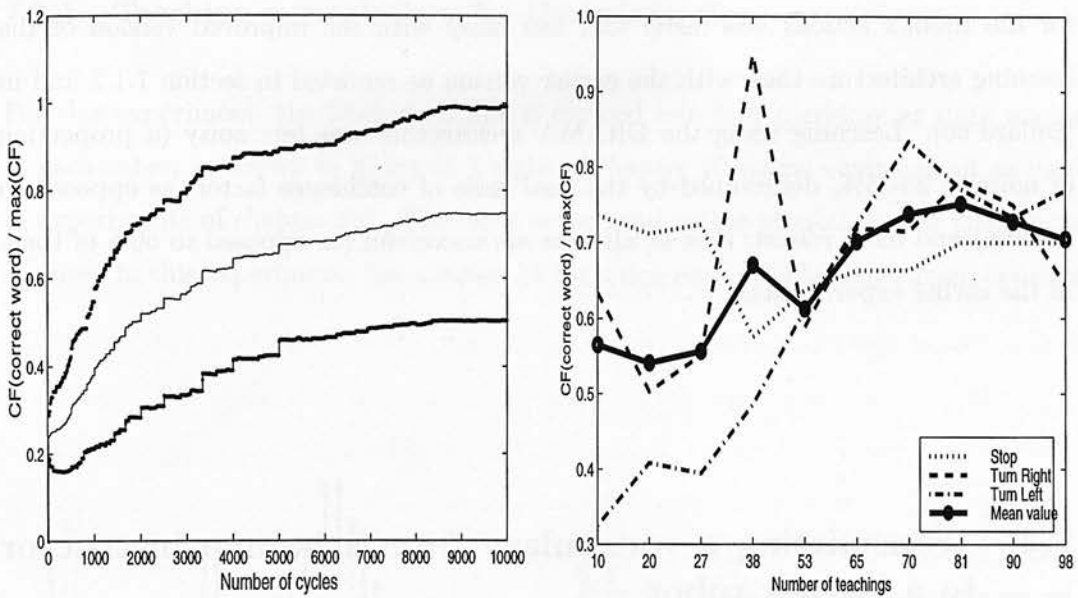


Figure 7.5: Confidence factor values during a run in simulation studies (left) and physical experiments (right), for the second phase of teaching. **Left:** The three curves are the mean value and the two extrema values (mean plus minus standard deviation) over all runs.

particular sensor measurement has been associated with the correct word more often than with other words. We observe that the three words are correctly learned after 38 teaching examples, which corresponds to about 15 to 20 minutes of experiment, which is faster than in the simulations. Again, this is probably due to our ‘helping’ the physical robots to align faster. However, the fluctuations of the values are stronger in the physical experiments, which means that the proportion of noise varies more in the physical set-up. The noise corresponds to the cases where the radio signals are associated with incorrect motor states or compass measurements. Because the movements of the physical robots are less smooth, incorrect alignment and imprecise following of the two robots is more pronounced and more frequent, which results in the robots facing and measuring different directions and producing different movements (e.g. turning in opposite directions). Note also that learning of the word *stop* is better and occurs already from the beginning, due to the fact that this word had already been taught during the first phase of the experiment and the memory was not erased between the two phases.

Results of the second phase of the experiments showed that learning of the vocabulary

for the robot's actions was faster and less noisy with the improved version of the learning architecture than with the earlier version as reported in section 7.1.2 and in [Billard 96]. Learning using the DRAMA architecture was less noisy (a proportion of noise of 25-35%, determined by the final ratio of confidence factor, as opposed to 40-50%) and more robust, that is, all runs are successful (as opposed to 60% of them in the earlier experiments).

## 7.2 Transmitting a vocabulary from a human instructor to a learner robot

In this section, we report on two experiments in which the robot is taught by a human instructor. In the previous section, we reported on experiments in which the robot was taught by another robot. There we used an imitative strategy, namely the two robots following one another by mutual phototaxis, to allow the learner robot to make similar observations to that of the teacher and, in particular, to make the observation relevant to the teaching. In the experiments we report here, the human instructor uses two methods to constrain the robot to make the desired sensor measurements at the desired moment. In the first experiment, the robot's travel direction is restricted by the robot following a series of four corridors at right angles to each other. Consequently, the robot is forced to direct its body following four different orientations, corresponding to that of each corridor, which it is taught to label with a four-word vocabulary.

In the second experiment, the robot is taught a vocabulary to describe objects in the environment. The human instructor directs the robot by moving a bright lamp in front of it, which the robot follows by phototaxis, similarly to the two robot experiments of section 7.1. By so doing, the instructor lets the robot approach each object one after the other, while sending each radio signal to label each object, each time the robot is close enough to the new object. The two experiments are carried out in the Dodgem cage with the learner LEGO robot, provided with all the sensors described in section 6.1.1.

### 7.2.1 Teaching a vocabulary for the orientation

For this experiment, the Dodgem arena is divided into four corridors at right angles to each other, as shown in figure 10.1 right of chapter 10 (same environment as used in experiments of chapter 10). The same sensor and motor capabilities for the robots are used in this experiment. See chapter 10 for a description. The experiment consists

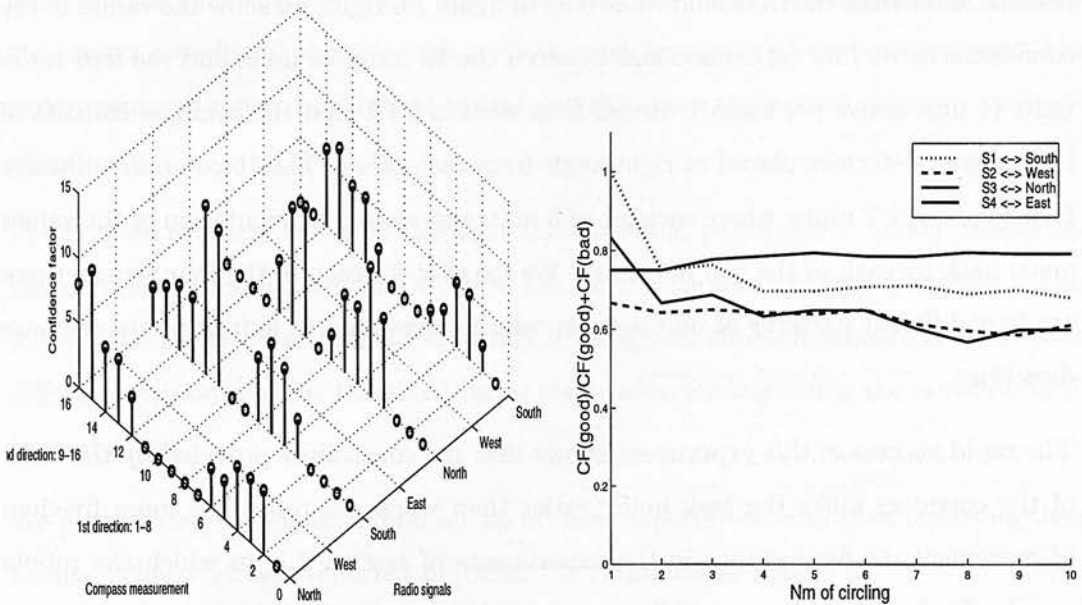


Figure 7.6: **Left:** Mean value of confidence factor parameters for each compass-radio signal pair. **Right:** Learning curve along each cycling around the cage for each of the four associations signal-compass orientation.

of letting the robot run along the series of corridors for ten complete circles around the cage. By means of a portable radio transceiver, the human instructor sends four different signals for when the robot is travelling in each of the four different corridors. These four signals have to be associated with the four different compass directions the robot measures when travelling inside the corridors. As the corridors are placed at right angles to each other, the measurements correspond to the four quadrants of the compass. After each complete circle around the cage, the robot is stopped and the state of its knowledge (i.e. the values of the network's confidence factors) recorded via an I2C connection from the robot's brain to the PC.

In figure 7.6 right, we show the progress of learning during teaching. The Y-axis gives

the values for the ratio between confidence factors of the correct connections and the maximal value of the confidence factors of incorrect connections, i.e.  $\text{cf}(\text{correct})/(\text{cf}(\text{correct})+\max(\text{cf}(\text{incorrect})))$ . We observe that learning of the four words is already successful after the first circling, that is, the correct connections have been activated more often than the incorrect ones, hence a ratio value greater than 0.5. Moreover, correlations are stable, as the ratio of confidence factor does not become lower than the threshold value 0.5. In figure 7.6 right, we show the values of the confidence factors for all connections between the 16 compass units and the four radio units (1 unit active per signal). Recall from section 6.1.1 that the compass consists of two magnet detectors placed at right angle from each other. The 16 compass units are formed of twice 8 units, where each set of 8 units represent a segmentation of the values given back by each of the two detectors. We see that for each of the four signals there are four different patterns of unit activity which correspond to four opposite compass directions.

The rapid success of this experiment shows that the constraints provided by the walls of the corridors make the task much easier than when the robot has more freedom of movement, as for instance in the experiments of section 7.1, in which the robots wander freely in a wide area. Following the walls prevents the robot from making too big sideways movements (this can happen as a result of its imprecise gearing) and thus limits the chances of incorrect sensor measurements during teaching. The experiment was repeated five times, showing the same result each time (that is fast and stable convergence). Figure 7.6 shows only one example of learning curve. This work has been published in [Billard & Dautenhahn 98a].

### 7.2.2 Teaching a vocabulary for describing objects

In this experiment, the robot is taught to distinguish between different objects by associating them with different labels, i.e. radio signals. We carry out a qualitative and quantitative study of the influence of environmental parameters and of one parameter internal to the robot's controller on the success of the experiment. We carry out simulated and physical experiments, while varying the environmental set-up (i.e. the objects' dispersion in the environment and the number of features common to two

objects) and the value of the short-term memory. In the theoretical analysis of the DRAMA architecture's performance of chapter 5, we showed the sensitivity of the network capacity to a non orthogonal pattern encoding (in an input-output pattern representation, learning is unsuccessful if two patterns encoding overlap both on their input and output, see section 5.1.1). We also pointed out the importance of carefully choosing the value of the short-term memory (which is fixed by the decay rate of the units' activation along the units' self connection), so that it is long enough to allow the correct match and short enough to distinguish between different associations. The experiments we report here intend to evaluate the above theoretical claims in a robotic experiment.

Further, we evaluate the influence that the threshold parameter  $T$  of the threshold-based retrieval mechanism in DRAMA has on the success of the experiment. This follows the study of section 5.1.4, which investigated through numerical simulation different variations of the threshold-based mechanism for improving the network recall capacity.

We present first the experimental set-up of these experiments and then report on their results. This work was reported in [Billard & Dautenhahn 98b].

### **Experimental set-up**

The physical set-up consists of the Dodgem cage, which contains three objects: one part of the arena is covered with an L-shaped aluminium foil (see figure 7.7 left), which the robot can detect through a set-up of a light bulb and 2 light detectors fixed under its body (the reflection of the bulb light being stronger on the aluminium foil than in the rest of the arena), as described in section 6.1.1. There is also a cardboard wall and a box, which are higher than the walls of the arena and which the robot can detect through the activation of whiskers placed on the side of its body.

The simulated environment consists of a rectangular arena whose sides are 15 times the robots' dimensions. It contains three hills, shown as big rectangles, and two boxes shown as small squares. In figure 7.7 right, we see the teacher robot followed by the learner robot, with hill 3 and box 2 on their left and box 1 on their right. Teacher and

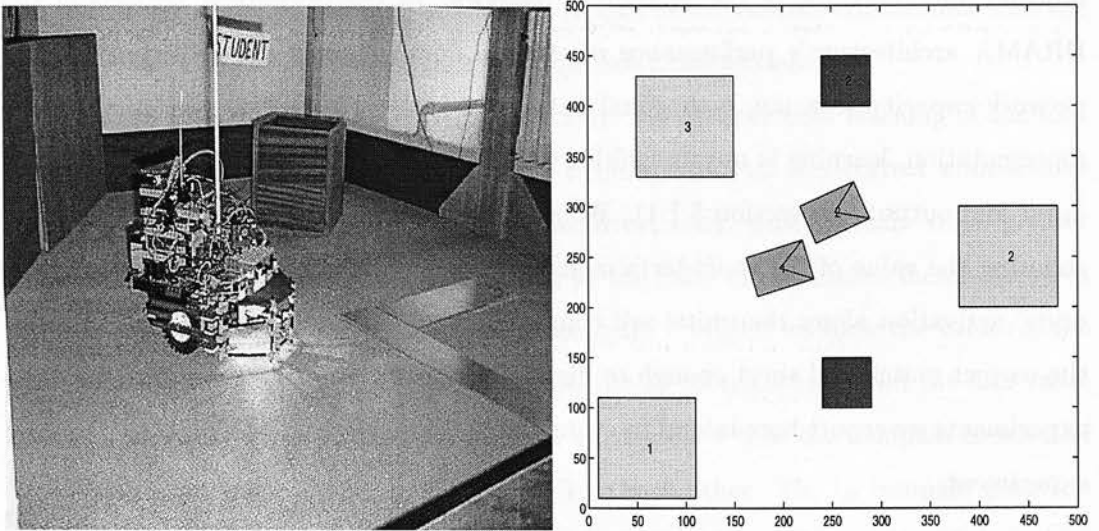


Figure 7.7: **Left:** The LEGO robot standing in front of an aluminium covered area. Behind it are the box and cardboard wall. **Right:** The simulated teacher robot (No 2) followed by the student robot (1), moving between hill 2 and 3.

learner robots are represented as rectangles with a triangle on the front. Each element (hill or box) has a distinct set of features, which allow the robots to distinguish between them: two of the hills have the same inclination but different colours, while the third hill has a different inclination but the same colour as the second one. The inclination of the hill influence the robot's behaviour. As the robot runs slower (by 20%) when climbing up the hill and faster (by 20%) when climbing down. The two boxes have different colours but the same shape. Boxes are distinguished from the walls by their shape, i.e. the upper infra-red detector signals presence of a box, which is taller than the walls, while the walls produce a response in only the lower one.

Physical and simulated robots are controlled by the DRAMA architecture, similarly to the description of section 6.2. A detailed description of the sensor sensitivity, range and their binary transcription in the neural architecture is given in table 6.1. There are 29 units ( $2 + 1 + 6 + 2 + 10 + 8 =$  Bumpers + IR + Light-detectors (front + below) + Side-switches + Motor-states + Radio-sensor) in the network of the physical experiments. There are 42 units ( $1+8+8+8+2+2+2+5+6 =$  Default-speed + IRs + Light-detectors (front + back) + Color-detector + Inclination-sensor + Shape-detectors (2 upper infra-reds) + Radio-sensor + Motor-states) in the simulation. The encoding

of the sensor and actuator information is described in detail in page 133.

## Results

We carried out nine simulation studies and five sets of physical experiments in which we varied the values of two environmental parameters, namely the *object dispersion* (OD) and the *object similarity* (OS), and of one learning parameter, namely the duration of short-term *memory* of sensor stimuli (M). The object dispersion (OD) is equal to the minimal distance between two objects over all pairs of objects. The object similarity (OS) is equal to the maximal number of features common to two objects. Table 7.2 gives the values of these parameters in each of the experiments. Note that simulation studies and physical experiments differ in two aspects: 1) they use a different environment (5 objects described by 2 features in the simulation and 3 objects described by a single feature in the physical setup)<sup>4</sup>; 2) in the simulations the learner robot is taught by another robot while, in the physical experiments, it is taught by a human instructor. However, the learning task is the same in the two setups; that is, the learner robot has to associate each teacher's radio signal with the correct combination of sensor perceptions which describe the particular object the signal is referring to. Each object in the environment is described by a distinct signal. Thus the experiments are successful when the 5 and 3 correct 'object-signal' associations have been made in the simulations and physical experiments respectively.

Figure 7.8 shows on the left graphical representations of the object dispersion of simulator experiments exp1-2-3. The three hills are represented as large squares on the sides of the arena, the boxes are the two small dark squares in between. On the right, we show plots of the values of the confidence factor, Z-axis, of each connection between the radio sensor units on Y-axis (1 unit activated = 1 word) and the colours, shape (IR) and inclination sensors units on X-axis (objects features) for each of these experiments. As shown by table 7.2, the farther the objects are from each other, the better the learning.

---

<sup>4</sup> The main reason why simulated and real environments differ is that, due to hardware restriction, we were limited in the number of distinct objects (with different features recognisable by the robot) we could have for the physical experiments. We used more objects in the simulations in order to study a more complex environment than was possible in the physical experiments.

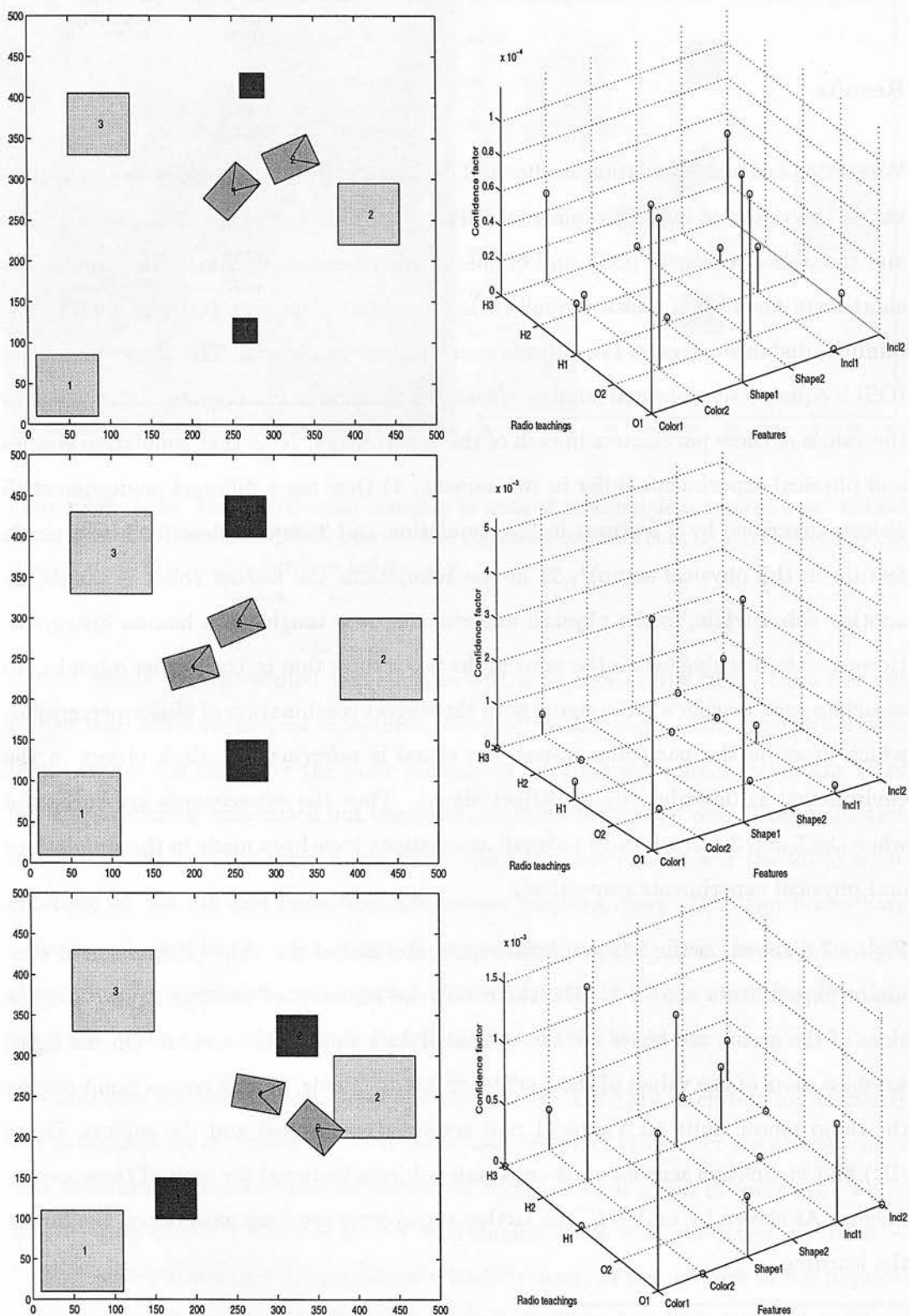


Figure 7.8: Comparison of three simulator experiments using different object dispersion.

For each simulation study, we realised a set of 30 runs, where a run simulates 200,000 processing cycles of the robots (about 55 hours of real experiments). For each run, the robots started in a different position, randomly generated, and their direction of movement was reset to a random value every 1000 cycles in order to allow homogeneous covering of the space. Each physical experiment lasted for about 30 to 45 minutes during which approximately<sup>5</sup> 40 teaching examples (20 for each object) were given. Each experiment started with the robot in the middle of the arena. The teacher, carrying a lamp, would then lead the robot through each of the locations (box, cardboard wall and aluminium foil) for twenty times, approaching each location by different paths so as to avoid bias in the teaching process. Table 7.2 shows the result for each signal-object association in each study. In the table, exp1-9 and p-exp1-5 stand respectively for the nine simulation studies and the 5 physical experiments. The values of OD and M are given in terms of body lengths (1 body length=30cm=3sec.). TS stand for total success of the experiment, which means that the complete vocabulary has been learned. B1/B2 stand for box 1 and 2, and H1/H2/H3 for Hill 1, 2 and 3. Y/N stands for Yes/No (correct or incorrect correlation between radio signal and object or successful/unsuccessful learning for column TS). 'Y-N' in the table 7.2 means that the ratio between confidence factor parameters of correct and incorrect connections, i.e.  $cf(\text{correct})/(cf(\text{correct})+\max(cf(\text{incorrect})))$ , was equal to 0.5. In other words, the robot has acquired the same confidence for good and bad answers.

The studies in which the parameters of object dispersion (*OD*) and short-term memory duration (*M*) are varied (exp1-7 of simulations and p-exp1-5 of the physical experiments), show that learning is unsuccessful when the proximity between the objects is lower than or equal to the distance that can be travelled during the short-term memory duration (exp2/3 and p-exp1/2). When the memory duration is too long relative to the object dispersion, learning is unsuccessful because too many observations are being remembered and associated with the incoming radio signals. In particular, because of the significant amount of experimental noise in the physical experiments, a value of  $M > 2$  leads already to unsuccessful results. Noise, i.e. incorrect matching signal-observation, was significant because the light detectors, which detect the aluminium

---

<sup>5</sup> The exact number of teaching examples is imprecise because the radio transmission was imperfect and it was difficult to detect whether the radio signal had been received by the robot.

Table 7.2: Results of experiment 7.2.2.

	OD	M	OD/M	OS	TS	B1	B2	H1	H2	H3
exp1	5.5	M=4.5	> 1	2	Y	Y	Y	Y	Y	Y
exp2	4.5	M=4.5	=1	2	N	Y	N	Y	N	Y
exp3	2.5	M=4.5	< 1	2	N	N	N	N	N	Y
exp4	5.5	M=1.5	>> 1	2	N	Y	Y	Y	N	N
exp5	5.5	M=2.5	> 1	2	N	Y	Y	N	N	Y
exp6	5.5	M=5.5	= 1	2	Y	Y	Y	N	Y	Y
exp7	5.5	M=33	<< 1	2	N	N	N	Y	N	N
exp8	4.5	M=4.5	=1	1	N	N	N	N	N	N
exp9	4.5	M=4.5	=1	0	Y	Y	Y	Y	Y	Y
p-exp1	1.5	M=2	< 1	0	N	N	N	N	-	-
p-exp2	2	M=2	=1	0	N	Y	Y	N	-	-
p-exp3	3	M=2	> 1	0	Y-N	Y	Y	Y-N	-	-
p-exp4	3	M=1.5	> 1	0	Y-N	Y	Y	Y-N	-	-
p-exp5	3	M=1	>> 1	0	Y	Y	Y	Y	-	-

foil, and the whiskers, which detect the box and the cardboard wall, could be activated even when the robot was far away from these objects (random reflection of the teacher's light on the floor, flicker of the whiskers under a shock). Thus, increasing the short-term memory duration increases the probability of incorrect activation of the sensors, consequently increasing the chances of incorrect associations.

On the other hand, the results of exp4/5 and p-exp3 show that learning can also be unsuccessful when  $M$  is too short, i.e. lower than a minimal value equal to 4.5 and 1 body length respectively for simulated and physical experiments. When the memory duration is too short, learning is unsuccessful because the time delay between the observation of the object and the reception of the radio teaching is too long. The radio signal is then forgotten before being associated with the observation of the object. Because teacher and learner do not share the same physical position, their relative perceptions of the world differ. An object visible to the teacher at the time of its teaching becomes visible to the learner only when it reaches the teacher's position, that is after having travelled a distance equal to about the teacher's body length plus the distance separating it from the teacher (about 1.5 body lengths). Thus, the duration of the short-term memory of events should be at least equal to the time needed for travelling this distance. A minimal value of 4.5 body lengths in the simulation means that the learner needs more time than expected to reach the physical position of the teacher at the moment of its teaching. This is due to the imprecise following of the two

agents. They lose time compared to when they are travelling in a straight line through performing sideways movements in order to adjust their position one behind the other. In the physical experiments, a minimal value of 1 body length only is enough, which results from the fact that the teaching is done by a human. Here the teacher has some notion of the learner's state of perception, by observing the location and orientation of the robot. The time delay between reception of the teaching signals and observation of the object depends therefore on the accuracy of the teacher's observation and prediction of the robot's behaviour and is smaller than with the two robots following each other. One may then think that the learning task is made easier when the coach is a human rather than another robot. However, this is only partially true. The learning task consists of discriminating between all the observations made in a time window  $\pm M$  around the occurrence of a radio teaching. This time window can be made shorter when teaching is given by a human, and for this reason the discriminating task might be easier as fewer observations would be made. However, it is important to understand that the human coach can never be certain that the robot has made or will make the expected observation before or after providing the teaching, nor can she be sure that the radio teaching has been correctly received (since she has no direct access to the robot's internal processing). Thus, the human coach does not have more knowledge of the success of the teaching episode than a teacher robot has, and the success of the experiment still lies in the learner's discriminating abilities.

Finally, exp8/9, in which the number of features common to several objects ( $OS$ ) was varied, show that learning improves when the overlap between the objects' description diminishes. This result is consistent with the theoretical prediction that the associative memory capacity should decrease in the face of data whose encoding descriptions overlap, as is the case when two object descriptions share common features.

In summary, the results of the experiments show that simulated studies and physical experiments are consistent in showing the same influence of the parameters  $OD$ ,  $M$  and  $OS$  on the success of the learning. In particular, the experiments indicate that  $OD$  and  $M$  are not independent and that an improvement in the learning can be obtained when their ratio is greater than or equal to 1, which means that the memory of the objects' observation is shorter or equal to the time needed to travel from one

to the other. They also show that increasing the value of  $OS$ , that is the overlap between object descriptions, decreases the chance of success of the learning. In other words, the more distinct the objects (their features) of teaching are, the better the learning. Similarly, the farther away the objects are from each other, the better the learning. Finally, the duration of short-term memory should be neither too short, in order to allow remembering of the teaching signals for association with subsequent sensor perceptions, nor too long, in order to reduce the associations of teaching signals with too old and thus irrelevant sensor perceptions.

### **Evaluation of the learning success dependency on the threshold $T$**

When discussing the model's capacity in section 5.1.1, we pointed out the importance of the threshold parameter  $T$  for determining the success of the learning. The value of  $T$  determines the minimal ratio between the values of confidence factors of correctly and incorrectly updated connections. Learning is unsuccessful when the percentage of noisy examples, i.e. incorrect update of connections, exceeds  $1/T$  times the total number of examples, because in this case the  $G$  function of the neuronal activation function given in equation 4.2 would output 0 when applied to the correctly updated connections, thus preventing the correct association to be retrieved. The value of  $T$  needs then to be carefully chosen, taking into account an estimation of the percentage of experimental noise. In table 7.9 top, we show the confidence factor values at the end of the run (mean values over all runs) for the simulation studies. We observe that some associations between radio units (each unit defines a different word) and boxes and hills' features are spurious. That is, for instance, the radio unit standing for hill3 is correlated with the combination of features of both box2 (colour2 + shape2) and hill3 (colour2 + inclination2), but with different values of confidence factor. Lowering too much the threshold on confidence factor could then have the effect of allowing retrieval of the two combinations of features rather than just the one for hill3 when presenting the radio signal for hill3. In table 7.9 bottom, we show the effect of varying the value of  $T$  on our determination of the correlation success for these results.

As mentioned previously, we estimate a proportion of at least 20% of experimental noise

	Features					
	Color1	Color2	shape1	shape2	Incl1	Incl2
<b>Box 1</b>	1.20	0.00	1.20	0.00	0.02	0.00
<b>Box 2</b>	0.00	0.90	0.00	0.90	0.00	0.12
<b>Hill 1</b>	0.25	0.00	0.08	0.00	0.25	0.00
<b>Hill 2</b>	0.00	0.15	0.00	0.00	0.15	0.00
<b>Hill 3</b>	0.00	0.55	0.00	0.05	0.00	0.55

	$1/T = 0.97$		$1/T = 0.5$		$1/T = 0.1$		$1/T = 0.01$	
	$\rightarrow R$	$R \rightarrow$	$\rightarrow R$	$R \rightarrow$	$\rightarrow R$	$R \rightarrow$	$\rightarrow R$	$R \rightarrow$
<b>Box 1</b>	Y	Y	Y	Y	Y	Y	N	Y
<b>Box 2</b>	Y	N	Y	Y	N	Y	N	N
<b>Hill 1</b>	Y	N	Y	Y	N	Y	N	N
<b>Hill 2</b>	Y	Y	Y	Y	Y	Y	Y	Y
<b>Hill 3</b>	Y	N	Y	Y	N	Y	N	N

Figure 7.9: **Top:** Confidence factor values for connections between radio units (1 unit = 1 ‘word’: 5 words for the three hills and the two boxes) and colours, shape and inclination sensors units (hills’ and boxes’ features). **Bottom:** Success of signal-object correlation given 4 different values of threshold  $T$ . Y/N stands for Yes/No (correct/incorrect correlation); R: radio sensor. ‘ $R \rightarrow$ ’ and ‘ $\rightarrow R$ ’ columns show results of learning for each direction of association (from radio sensor to object’s description and vice-versa).

due to hardware imperfection, onto which we should add the noise due to the imprecise following of the two agents resulting in incorrect matchings of sensor perceptions. Taking a threshold of  $1/T = 0.5$  allows correct retrieval of the data in the face of a maximum of 50% of noisy data. As expected, with this value for the threshold, only the correct correlations (from each object to its corresponding word and vice-versa) are correctly retrieved. However, when lowering the threshold, spurious correlation can also be retrieved. E.g., with a threshold  $1/T = 0.1$  the radio signal for hill3 retrieves the sensor features for both box2 and hill3 and with a threshold  $1/T = 0.01$  the combination of features for hill3 retrieves both signals for hill3 and box2. On the other hand, a too restrictive threshold, that is too high, means that some correlations are no longer retrieved, e.g. with  $T = 0.97$ , the radio signals for box2, hill2 and hill3 no longer retrieve the full set of features of the corresponding elements.

### 7.3 Discussion and synthesis

This chapter reported on sets of experiments in which a robot was taught a vocabulary to describe its perceptions. The vocabulary concerned both the robot’s proprioceptions, such as perception of body inclination, orientation and motor actions, and the robot’s

exteroception, such as light, infra-red measurements and touch perception. Each experiment (except exp. of 7.1.2) was carried out both in simulation and in a physical environment. Simulation studies allowed us to study the learning progression and stability over a longer period of time than possible with the physical environment. The main advantage of simulations over physical experiments is that they are repeatable, faster (simulating a 1 hour experiment takes about 5 minutes) and do not suffer unexpected hardware breakdown. The disadvantages in terms of model faithfulness are well known (for a more complete discussion of this see [Torrance 92a]). The experiments we reported in this chapter pointed out quantitative differences between simulated and physical environment, e.g. the boundary values for the short-term memory for which learning is successful, in section 7.2.2, differ between simulations and physical experiments. However, the results of simulations and physical experiments were consistent (qualitative similarity) in showing the success and stability of the learning.

Teaching was provided either by a teacher robot (section 7.2) or by a human instructor (section 7.1). The success of the experiments in which the robot was taught by a second robot demonstrated the plausibility of an unsupervised learning strategy based on an imitative scenario for transmitting a vocabulary from a teacher agent to a learner agent. The coupling of the two robots' movements through the mutual following strategy, which resulted in them sharing common perceptions, was a sufficient (and necessary, as it will be demonstrated in section 9.2) means for guiding the learner agent's attention to make the observations relevant to the teaching. This strategy allowed to reduce greatly the amount of irrelevant data, i.e. incorrect observations, to analyse. The residue of irrelevant data were then discarded from relevant ones, by comparing the frequency of occurrence of each observation (DRAMA associative process). Note, also, that by using two different pairs of robots (FischerTechnik and LEGO ones), with different shapes and sensors, in two different environments, we demonstrated that the teaching scenario is unspecific to a particular robotic platform. This will be further confirmed by the experiments of chapter 11, in which we use a doll-shaped robot.

The success of the experiments in which the robot was taught by a human instructor demonstrated the plausibility of the teaching scenario, i.e. the following strategy and the learning architecture, for transmitting a symbolic representation of external per-

ceptions (objects' observation) and internal perceptions (orientation measurement) between heterogeneous agents. Heterogeneity refers here to agents which differ in both their internal functioning and their physical body (sensors, shapes, etc.).

In the physical experiment of section 7.2.1, the learner robot is not explicitly directed by the human instructor, contrarily to what happens in the other experiments of this chapter, in which the learner robot follows and thus is guided by the teacher robot. However, the constraints imposed by the corridor walls in the human-robot experiment are similar to those implied by the following behaviour, as both strategies result in implicitly directing the learner robot's movements and orientation. In building this constrained environment, the human instructor uses her high level perceptual and cognitive capabilities to infer the result of her guiding the robot's perceptions. However, the success of the simple following behaviour in the experiments of section 7.1 shows that this 'mind reading' capability is not necessary for the success of the experiment, as, in these studies, the teacher robot has no knowledge of the learner robot's perceptions. In the physical experiments, the human instructor had no direct access to the robot's internal processing during the experiment and could therefore not be certain that the robot had made the correct measurements (correct reception of the radio signal and correct compass measurement) and thus that the teaching episode was successful. Therefore, in both simulated and physical experiments, learning is unsupervised and the success of the experiment depends (1) on the constraints on the robot's behaviour which reduce the amount of analysed data and (2) on the robot's discriminating abilities.

Note that using a more supervised strategy, such as having the teacher robot using the learner robot's transmissions to guide its teaching (by it repeating more often the incorrectly learned signals), would not necessarily lead to a better learning. Given that the robots can not have access to each other's perceptions, then the teacher robot might interpret incorrectly the learner robot's transmissions. For instance, if the learner's signal does not correspond to what the teacher senses, the teacher might deduce that the learner has learned incorrectly the signal's meaning. However, this deduction might be incorrect, as the learner could be sending the correct signal, given its current perceptions, which often differ from those of the teacher, since both agents

share a different physical position in the environment and thus have a different view of the environment.

In chapter 5, we pointed out the importance of correctly choosing the values of the learning parameters, namely the duration of the short-term and the retrieval thresholds  $T$  and  $e$  of the activation function 4.2, on the success of learning.

Finally, in the experiments of section 7.2.2, we reported on simulated and physical experiments, in which we varied the environmental constraints and the duration of the short-term memory and studied how they influenced the results of learning. We could then relate cases of success and failure of learning to particular choices of environmental constraints (objects' relative dispersion and featural descriptions), and values of the duration of short-term memory of events and could then determine bounds on these parameters inside which learning would be successful. Further, we evaluated the influence of the threshold parameter  $T$  on our determination of the success of this experiment. These studies allowed us to validate the claims, made in chapter 5, on the importance of correctly choosing the values of the retrieval thresholds, the short-term memory parameter and the patterns' encoding (in order to minimise the overlap between two patterns) on the success of the experiment.

## Chapter 8

# Grounding of perceptions of a group of robots

This chapter reports on simulation studies in which we study how the imitative teaching scenario, described in section 7.1.1 and used in the experiments of section 7.1 for transmitting a vocabulary from one teacher to one learner agent, scales up to grounding communication among a group of robots. We first describe the experimental set-up and procedure and then report on the result of two sets of simulations. In the first set, we study how a common understanding of the vocabulary can spread among a group of agents, when starting with one teacher agent and when each successful learner agent can in turn become another teacher. In the second set of simulations, we study the influence of the following strategy on the learning performance for different learning tasks.

### 8.1 Learning the vocabulary

In this experiment, the agents learn a vocabulary to differentiate between coloured patches and to describe their location in terms of distance and orientation, i.e. in polar coordinates relative to a homing point. The vocabulary is transmitted from a teacher agent, which has a complete knowledge of the vocabulary from the start, to eight learner agents which have no knowledge of the vocabulary to begin with. Once a student becomes confident enough in its learning of the correct signal-meaning correlation (that is after having observed the same signal-sensor measurement pair a

sufficient number of times), it becomes teacher in its turn. The teacher agent does not learn; that is, its definition of the vocabulary remains unchanged during the whole experiment. The learner agents, however, carry on learning even when they begin to teach. Thus, a ‘bad’ teacher, i.e. a learner agent which has become confident in incorrect signal-meaning pairs, can become a ‘good’ teacher later in the experiment if it can update again its confidence on the correctness of its word-meaning pairing under correct external teaching (and reversely a good teacher can become a bad teacher under opposite circumstances). The level of confidence under which the learner becomes a teacher is varied in the experiments and its effect on the learning performance of the whole population investigated.

### 8.1.1 Experimental set-up

For these experiments, we use the simulated environment described previously in section 6.1.2. We define nine robots, which are provided with light and infra-red, compass, inclination, energy level and radio sensors. The sensitivity, range and encoding of these sensors are given in table 6.1. Note that, because we use a binary encoding for the sensor measurements, the objects’ polar coordinates, calculated from compass and energy measurements, take discrete values. In the experiment, there are 14 possible measures of distance (energy) and 16 possible measures of angle (compass). The processing of the sensor information in the DRAMA architecture is similar to that described in section 6.2. A detailed description of the sensor sensitivity, range and their binary transcription in the neural architecture is given in table 6.1. There are 93 units ( $1+8+8+8+9+8+14+31+6 = \text{Default-speed} + \text{IRs} + \text{Light-detectors (front + back)} + \text{Color-detector} + \text{Compass-quadrants} + \text{Energy-range} + \text{Radio-sensor} + \text{Motor-states}$ ) in the simulation. The encoding of the sensor and actuator information is described in detail in page 133. The same encoding is used for all experiments reported in this chapter. The short-term memory duration of unit activity is the same for all units and is equal to 600 processing cycles in both simulated and physical controllers.

The environment is a rectangular arena measuring 700 by 700 units, which contains nine objects, coloured patches, which can be distinguished by their different colours. The robots can perceive colours using the light detectors. Figure 8.1 shows a graphical

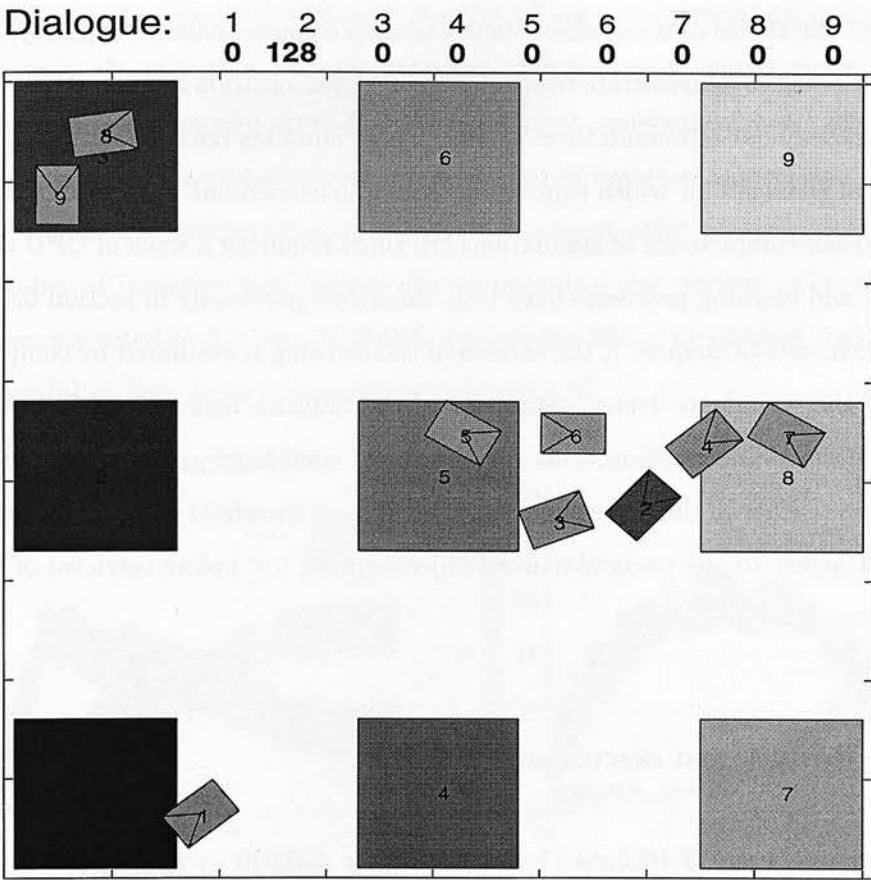


Figure 8.1: Graphical representation of the simulated environment for the experiments on learning of a group of robots.

representation of the simulated environment with the nine robots. The objects are represented as big rectangles of different colours. In the figure, we see the teacher robot (in dark grey) between five of the eight learner robots (in light grey), moving between objects 5 and 8. On top of the figure, the result of the robots' speaking is written for each of the nine robots. The teacher robot (2nd column) outputs '128'. This refers to the activation of the 8th radio unit in the robot's network, which is the radio encoding for the label of the object, across which the three robots are currently running. None of the learner robots is speaking, i.e. they all output '0', as they have not yet seen the object and thus have not yet associated the teacher's signal with the object's features. In Appendix A, we show eight snapshots of the same MATLAB simulation, which illustrate different phases of the teaching. An animated gif of the complete simulation can be seen from an http site: <http://www.dai.ed.ac.uk/daiddb/people/homes/audeb/simul2.html>

Note that the choice of using nine robots and nine coloured patches is purely arbitrary, our point being to demonstrate transmission of communication inside a group of agents, i.e. composed of at minimum three agents. These numbers refer in fact to the maximal number of elements for which simulations could be carried out in a reasonable amount of time, each complete set of simulations (10 runs) requiring a week of CPU time. The routines and learning processes have been described previously in section 6.1.2. As in the experiments of chapter 7, the success of the learning is evaluated by comparing the value of the confidence factor parameter of the network connections for correct and incorrect word-object connections. The level of confidence under which the learner becomes a teacher in the experiment is set by fixing a threshold value for the confidence factor attached to the particular word-object connection before retrieval of the word output can occur.

### 8.1.2 Results and discussion

We carry out a set of 10 runs (1 run lasting for 400,000 cycles) in which we study the speed of learning of a vocabulary of nine words by eight learner robots, given 1 teacher robot. The nine words refer to nine different coloured patch types, defined by nine different colours. The coloured patches are spread homogeneously in a square area as shown in figure 8.1. The learner robots can become teachers, that is they can emit signals for 'naming' the coloured patches, once they have reached a sufficient level of confidence in their word-colour associations. We ran 10 simulations, in which we varied the threshold between 10 and 100, which corresponds to considering the agent sufficiently confident in a particular word-colour association when it has observed this combination for at least 10, 20, 30, ...,100 times. This refers to the actual value of the confidence factors for these word-colour connections, as the confidence factors are increased by a value of 1 at each co-activation of the corresponding units. Because the agents can make incorrect associations, due to mismatched observations of teacher and learner agents, correct learning of the word-colour pairs is not immediate but results from the repetition of the teaching process until incorrect associations can be discarded from the correct ones by virtue of their relative frequency of occurrence. Therefore, choosing too low a threshold on confidence can allow the learner agent to become a

teacher before it has made a sufficient number of correct correlations compared to incorrect ones. In this case, the robot would emit incorrect signals given a colour perception, hence letting the learner agent following it make an incorrect association. This would increase the global amount of noise, i.e. of incorrect association episodes, and could lead to the failure of the experiment (i.e. unsuccessful learning for all agents) if the number of incorrect associations should overcome the number of correct ones. This is demonstrated by the results of the experiments where we observe that learning is unsuccessful as long as the threshold is lower than 50.

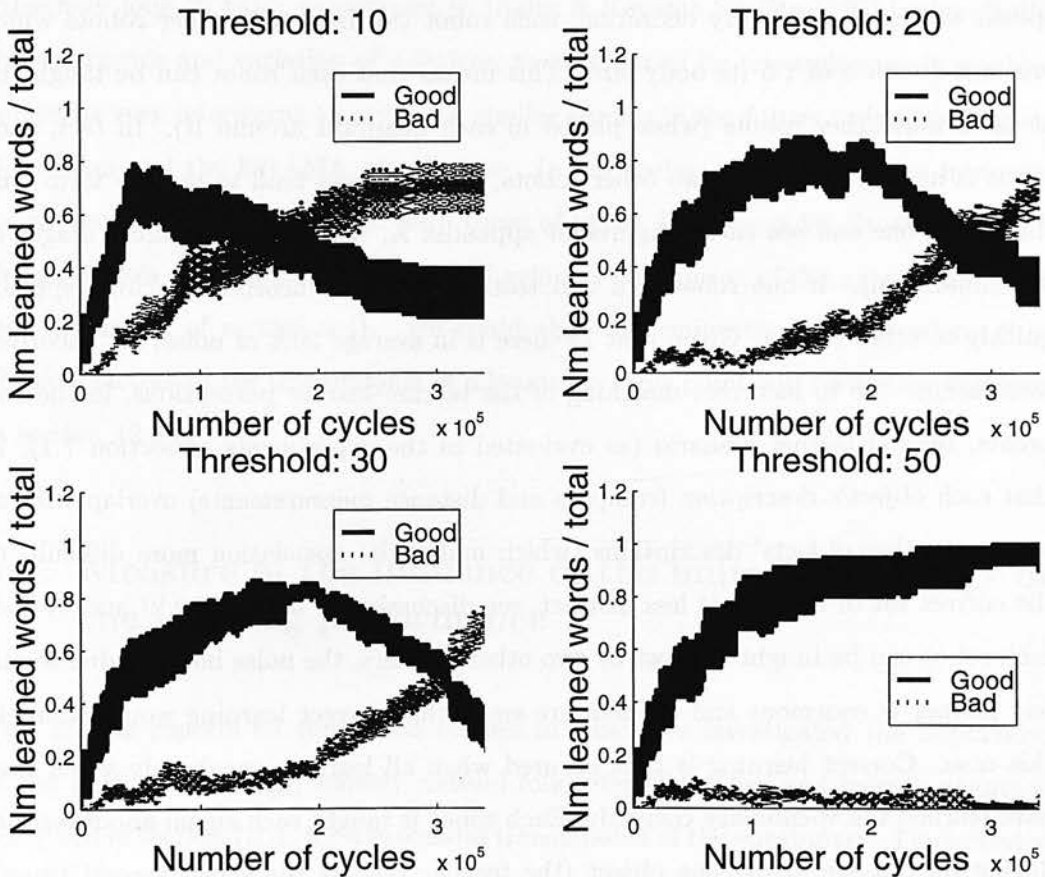


Figure 8.2: Learning curve (mean value for all agents) for four different thresholds (10,20,30 and 50) on the radio unit activation.

We show in figure 8.2 the learning curve for four different choices of minimal levels of confidence out of the ten investigated. Note that the curves for the threshold values greater than 50 have similar shape to that shown in figure 8.2. The simulation with threshold 50 was carried out for 20000 more cycles and it was shown that the curve

did not decrease after having reached the maximal value of 1. We observe that the number of incorrect correlations increases in the three first examples, so as to become more frequent than correct ones, leading finally to unsuccessful learning. The value of 50 for the threshold under which the learning is successful reflects the proportion of noise, i.e. incorrect associations due to incorrect matching of the teacher-learner perceptions, in the particular set-up. This value would then vary from one experiment to another. In the present case, the increase of the noise, which lead to unsuccessful learning, is due to incorrect teaching provided by the learner robots. The more robots speak, the more noise. By definition, each robot can hear only other robots which are in a distance of 1.5 its body size. This means that each robot can be taught by at most four other robots (when placed in each quadrant around it). In fact, each robot is usually taught by two other robots, as the robots tend to quickly form long chains (as one can see in the figures of appendix A, which show different stages of the simulation). If one robot is a bad teacher, then its incorrect teaching spreads quickly to other robots. Given that 1) there is in average 20% of noise, i.e. incorrect associations due to incorrect matching of the teacher-learner perceptions, in the two agents, teacher-learner scenario (as evaluated in the experiments of section 7.1), 2) that each object's description (compass and distance measurements) overlap with at least two other objects' descriptions (which makes the association more difficult, as the correct set of features is less distinct, see discussion of section 7.2.2) and 3) that each robot can be taught at most by two other learners, the noise increase due to one bad learner is enormous and chances are small that correct learning would result in this case. Correct learning is thus ensured when all learners speak only when they have learned the vocabulary correctly. Each robot is taught each signal about 5 times during its passage across one object (the teacher repeats the signal several times). Correct learning of the vocabulary requires at least 10 passages over each object (i.e. 10 different teaching episodes, in which the robot approaches the object from a different direction). Therefore, correct learning should occur after about  $10 \cdot 5 = 50$  associations, i.e. a value of 50 for the confidence factors associated with the signal-object features.

The simulations in which learning was unsuccessful were stopped while the curve of bad association was still increasing (see curves of thresholds 20 and 30 in figure 8.2. If we had let the simulation run longer, we could have checked whether the curve

would converge to the maximal value of 1. In this case, this would have meant that all robots had finally converged to a common definition of the vocabulary, while this definition would have been different from that initially taught by the teacher robot. The observation of a shift in the language definition, defined as the ‘emergence of a dialect’ have been made by [Arita & Koyama 98, Steels 97b] in their simulations of the development of language (see review of these works in section 2.3.1). We did not make this analysis, mainly because the aim of our experiment was to study correct transmission of a fixed pre-defined vocabulary (and also because these simulations were extremely long to run), as opposed to [Arita & Koyama 98, Steels 97b] who studied the emergence and variation of a lexicon as an effect of its transmission. It would of course be very interesting to carry out similar studies in the future, using the physical simulation and the DRAMA architecture. In particular, it would be very intriguing to compare the results obtained with those of [Arita & Koyama 98, Steels 97b], as in these people’s studies, no physical or behavioural description of the agents were given (see discussion of section 3.4). We could then determine the role played by these physical factors in the transmission of a language. This point will be further discussed in section 12.3.2.

## 8.2 Measure of the influence of the imitative strategy on the learning performance

This section reports on simulation studies in which we investigated the importance of the imitative strategy, namely mutual following of teacher and learner agents as described in section 7.1.1, for a successful transmission of the vocabulary. The imitative strategy allows the agents to share a common set of internal and external perceptions, as they perform the same movements, and, thus, travel over the same ground (thereby having similar relative measures of energy consumption, orientation, inclination), and as they share close physical positions and ‘look’ in the same direction. It is thus expected that, if the agents lack the capability of imitating/following each other, their perceptions at any point of time would seldom be similar, and consequently the chances for the learning to be successful would strongly decrease. We carry out simulations in which we compare the learning success of two learner robots, while only one of the

learners is capable of following the teacher robot.

### 8.2.1 Experimental set-up and procedure

For these studies we use the same simulated environment as described in section 8.1, namely a rectangular arena containing nine different objects (see figure 8.1), with three robots, 1 teacher and two learners. The learning task is the same as in the experiments reported in section 8.1, namely learning a vocabulary to label the nine objects and to describe the objects' location in terms of polar coordinates. We study the influence of the imitative strategy on the learning success by comparing the learning performance of the two learner agents, only one of which is capable of following another agent by means of phototaxis. The three agents are wandering randomly in the arena. Once the follower agent meets the teacher agent or the second learner agent, it begins to follow it. When the non-follower agent meets one of the two other agents, it simply avoids it as an obstacle and carries on its random wandering; that is, in the non-follower agent, the DRAMA connections between light sensor and motors have been set up to produce obstacle avoidance, while in the follower agent, the connections have been set up to produce phototaxis. The non-follower agent can learn, similarly to the other learner agent, that is, if it is close enough to the speaking agent, it can receive its signals and associate them with the sensor measurements it has made during a short time delay window before and after the signals' reception (plus or minus the short-term memory duration).

### 8.2.2 Results and discussion

We carried out a set of ten runs, where 1 run corresponds to 100,000 processing cycles. Figure 8.3 shows the learning progress of the two learner agents when learning three different types of vocabularies: 1) a vocabulary to describe the nine objects (coloured patches) of the environment, 2) a vocabulary to label scaled values of polar coordinates referring to the objects' locations and 3) a vocabulary for the four quadrants of a compass, which measures the robot's individual orientation. Learning each of these vocabularies corresponds to associating radio signals, i.e. the teacher's words, with respectively 1) nine different colour measurements (objects' features), 2) five differ-

Table 8.1: Comparison of learning success of follower and non-follower agent

	<b>objects</b>		<b>polar coord.</b>		<b>orientation</b>	
	Foll	Non-foll	Foll	Non-foll	Foll	Non-foll
Nm-learned-words/total-voc.	1.0	0.6	0.5	0.5	0.6	0.0

ent measurements along the energy scale and eight different measurements along the global compass scale (objects' polar coordinates), and 3) four different compass measurements. The Y-axis represent the ratio between correctly and incorrectly learned words. In table 8.1, we show the mean value of this ratio over the 10 runs for each experiment and for each learner robot.

Results show that the non-follower agent is less successful on average and slower at learning the vocabulary concerning the coloured patches and the polar coordinates and was always unsuccessful at learning the vocabulary concerning the orientation relative to a compass. These results imply that the ability for following improves the grounding of extero perceptions, as done when naming the coloured patches detected by different colour perceptions and when naming its position relative to global polar coordinates. But it is especially important for grounding proprio perceptions, as done when naming its relative orientation. Being close enough spatially is often sufficient for the agents to share a common context of external perceptions and then to successfully ground the vocabulary onto the same sensor perceptions. In the experiments, the non-follower agent learns the vocabulary concerning extero perceptions correctly because, when it receives the teacher's signal, it is often close enough to get a similar measure of polar coordinates (the spatial scaling of the environment it learns to name is wide enough to allow two agents to share the same set of coordinates at one time) and to make a measure of colour (detection of coloured patch) shortly before or after meeting the teacher. The follower agent is, however, faster and more successful at learning because of its constant spatial closeness to the teacher agent. By contrast, it is not sufficient for the agents to be spatially close to one another for them to share a common set of internal perceptions, which would allow them to successfully ground these proprio incorrectperceptions onto a common vocabulary. Imitating or replicating another agent's actions lets the imitative agent go through the same actuator states as the imitated agent. It also allows it to make other similar internal perceptions which

are a consequence of its actions, e.g. its orientation, inclination (see experiments reported in section 7.1.2), and relative energy consumption. In the experiments, the follower learner agent implicitly imitates or replicates the teacher's movements while following it, and consequently orients its body towards the same direction as the one pointed to by the teacher. In contrast, the non-follower agent, which tries to avoid the teacher agent, is not or seldom (and then only for a short period) oriented similarly to the teacher and is thus less likely to correctly associate the teacher's signals, which accounts for the failing of its learning observed in all 10 simulations.

### 8.3 Synthesis

This chapter reported on two sets of simulation studies. In the first set, we showed that the learning scenario used in the experiments of chapter 7 to transmit a vocabulary from one teacher robot to one learner robot scales up successfully to transmit a vocabulary among a group of robots. The scenario is based on an imitative unsupervised learning strategy. The success of the experiment depends (1) on the constraints imposed on the learner agent's behaviour, resulting from the learner's following behaviour, which reduce the amount of analysed data to a relevant subset and (2) on the agent's discriminating abilities.

An interesting aspect of this experiment was that a learner agent could become teacher in turn, when sufficiently confident in its word-observation pairing. Results showed that it was important to fix a sufficiently high threshold for the level of confidence under which the robot could teach to prevent an unsuccessful result. If the robot teaches incorrectly, the number of incorrect associations which have to be discarded increases, leading to an unsuccessful result, if this number becomes greater on average than the number of correct associations. The value of the threshold depends on the proportion of noise, i.e. uncorrect teaching episodes, for the teacher-learner teaching scenario, which is different for each physical set-up and thus has to be estimated for each experiment.

In the second set of simulation studies, we studied the importance of the imitative/following strategy for the success of the learning. We carried out simulation studies, in which we compared the learning success of agents with and without the ability of following. The results of these studies showed that agents lacking this capability would

be slower at learning or would simply fail. The capacity of following the other is particularly important when the word to be learned concerns the agent's proprio perceptions, e.g. orientation, as it allows the two agents to share similar internal perceptions, e.g. pointing in the same direction.

### Chapter 9

## Using the Robot's communicative ability

The first part of the chapter discusses the experimental setup, which consists of two robots in a shared environment. The robots are equipped with a simple communication system that allows them to exchange information. The experimental setup is described in detail in the appendix. The second part of the chapter discusses the experimental results. The results show that the robots are able to learn the meaning of a word by observing the other robot's actions. This is achieved by using a simple learning algorithm that updates the robot's internal representation of the word's meaning based on the observed actions. The experimental results show that the robots are able to learn the meaning of a word even when the other robot's actions are noisy. This is achieved by using a simple learning algorithm that updates the robot's internal representation of the word's meaning based on the observed actions. The experimental results show that the robots are able to learn the meaning of a word even when the other robot's actions are noisy. This is achieved by using a simple learning algorithm that updates the robot's internal representation of the word's meaning based on the observed actions.

The second part of the chapter discusses the experimental results. The results show that the robots are able to learn the meaning of a word by observing the other robot's actions. This is achieved by using a simple learning algorithm that updates the robot's internal representation of the word's meaning based on the observed actions. The experimental results show that the robots are able to learn the meaning of a word even when the other robot's actions are noisy. This is achieved by using a simple learning algorithm that updates the robot's internal representation of the word's meaning based on the observed actions.

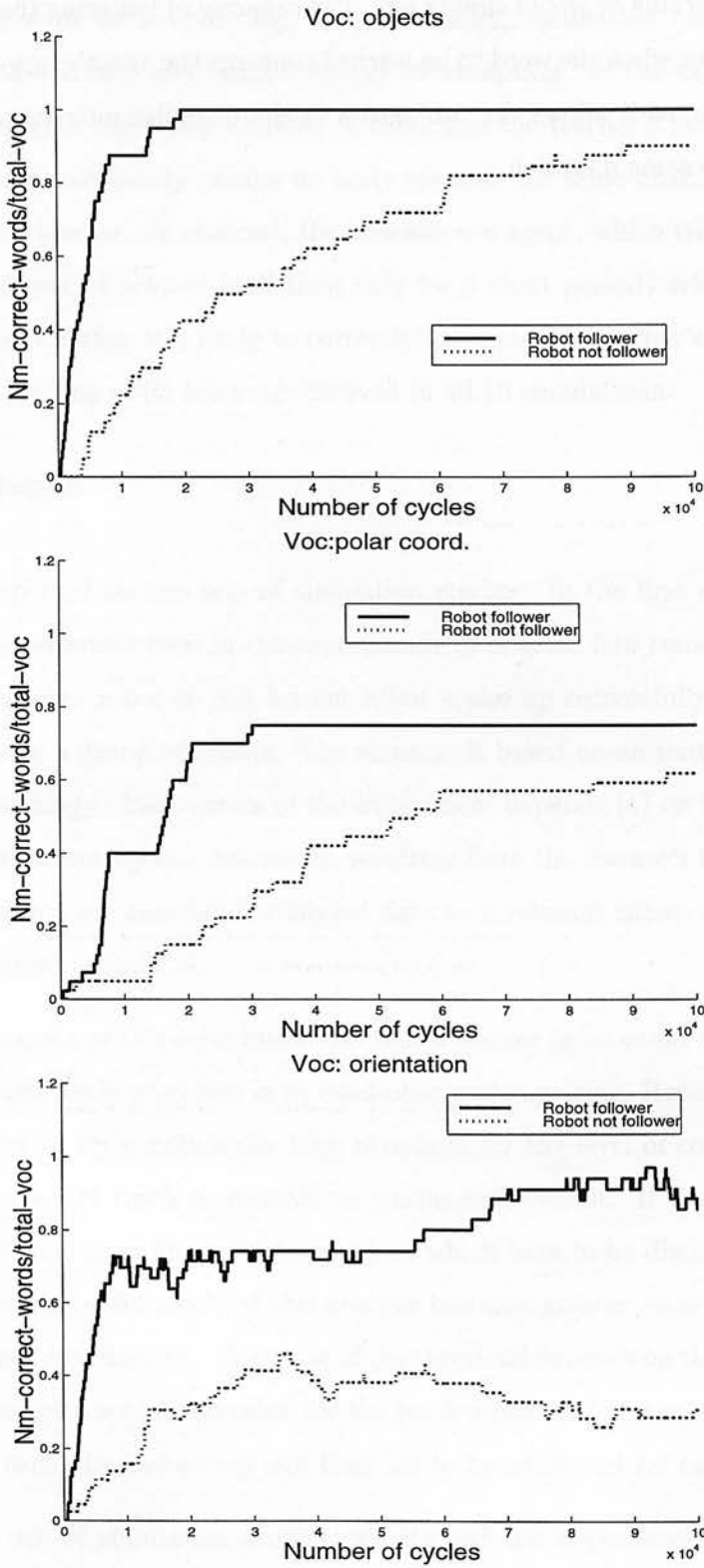


Figure 8.3: Learning curves for follower and non-follower learner agents in three different learning tasks: learning a vocabulary for objects, for scaling values of polar coordinates and for orientation relative to a compass.

## Chapter 9

# Using the robot's communicative ability

This chapter reports on three experimental case-studies which show different uses of the robot's communication ability as part of multiple robot and human-robot interactions. These experiments follow from the experiments reported in sections 7.1.2, 7.2.1 and 8.1, in which the robot was taught a vocabulary to describe its perceptions and actions. In the first section, we describe an experiment in which the robot uses its understanding of a vocabulary describing locations in the environment to find a second robot, when the latter is saying its location. This experiment uses the robot's knowledge of a vocabulary to distinguish between hilly and plane areas, which it had been taught in the experiment of section 7.1.2. In the second section, we report on an experiment in which a group of robots benefit from their common interpretation of a vocabulary as it speeds up their learning of new information. The robots' understanding of the vocabulary in this experiment results from the teaching experiment reported in section 8.1. In the third section, we present an experiment in which a human instructor uses the vocabulary that the robot can understand to describe its orientation in order to command the robot inside an unknown environment. This experiment follows from the experiment reported in section 7.2.1, in which the robot had been taught the vocabulary.

## 9.1 Two-robot collaboration

We report on a case-study experiment in which a robot benefits from understanding the vocabulary used by another robot. This experiment uses two autonomous robots, a 'child' and a 'mother' robot. Both terms (mother and child) are used in a metaphorical sense in order to characterise a situation where a) one robot is the model for another robot's behaviour, b) the teacher/mother robot knows the right words for useful places and situations in the environment, and c) the learner/child robot is supposed to have an instinct to stay close to the teacher robot, i.e. the learner can benefit from the knowledge of the teacher.

This experiment was reported in [Billard & Dautenhahn 97a]. Originally, the scenario of the experiment was composed of two parts, a teaching part and a testing part. The teaching part was reported in section 7.1.2. We describe here the testing part. To aid recollection of the teaching part, we first briefly sketch out the complete scenario.

In the first stage of the experiment (teaching stage), the child robot follows the mother robot, which travels randomly around the environment. While they wander, the mother describes its environment using different 'words' (emission of different radio signals) depending on whether they travel on the plane, up or down a hill. The child robot attaches a meaning to the mother's signals, by associating them with different perceptions of inclination. In the second stage, child and mother are separated. Then, because the child crucially needs its mother (following a predefined instinctive behaviour), it immediately begins to look for her. Unfortunately, because of sensory limitations, it cannot see her and thus begins a random search that may be long and may not succeed. Meanwhile, the mother is still naming the location where she is. The child, hearing and understanding the mother's messages, uses them as supplementary information for finding her. In the mother-child scenario, the ability to understand the language of the 'mother' is advantageous for the child because it gives supplementary information to find its mother.

### 9.1.1 Experimental procedure

In the experiment of section 7.1.2, the learner/child robot learned a 3-word vocabulary to distinguish between three areas of inclination: *plane - up - down*, which corresponded to moving on the plane, or up and down a hill. In addition, because a bright light was suspended on top of the hill, the learner robot had also associated an increase of light with climbing up and consecutively associated the word *up* with significant brightness and the word *down* with darker light measurements (see figure 7.3 right).

In the present experiment, we use the results of the experiments of section 7.1.2 for commanding the robot's movement. We use the learner/child robot's understanding of the vocabulary (namely its two associations 'radio signal plane = dark + plane inclination' and 'radio signal up = bright + up inclination') to direct the robot towards the bright or dark area of the arena, which correspond respectively to the plane and hilly areas. The robot is directed by the teacher/mother robot's signals 'plane' or 'up'. The mother robot sends the signal which corresponds to its own inclination and light perception, that is, it is telling its location. Note that there was a unique bright and hilly area and so there was no ambiguity on the signals. As the child robot is too far away to detect the mother robot through infra-red sensors, it uses its understanding of the mother robot's signal and its ability to detect light from farther away to direct itself towards the mother robot's location. We use two FisherTechnik robots, whose hardware has been described in the [Billard & Dautenhahn 97b]. In the following, we first describe the behavioural architecture used to direct the robot's behaviour and then we report on the results of the experiment.

### 9.1.2 The behavioural architecture

In order to allow switching between the robot's learning the vocabulary in the first phase (experiment of section 7.1.2) and then using it in the second phase (reported here), we defined behavioural processes in addition to the associative process, such as drives and retrieval mechanisms which discriminate between desired and real sensor states. Note that in this experiment we used a simpler associative architecture than DRAMA, which did not have event detector modules and had no record of time, i.e.

no recurrent structure and no time parameter (see section 7.1.2 for more explanations). Figure 9.1 shows a schematic representation of the complete architecture. The learning architecture is represented by three learning boxes, linking light, radio and inclination sensor systems. Note that the network is in fact a 3-D matrix and the learning process is a single routine. The division into three boxes in the picture was made to point out the bidirectionality of the association between each sensor system. The inputs of infra-red sensor and motor systems do not participate in the learning, hence they are not linked by a learning box. Explanation for the rest of the schema are given in the text below. The control architecture described here was implemented only in the learner/child robot. The behaviour of the teacher/mother robot was defined as a simple behaviour-based architecture, with two behaviours of *following* and *talking*.

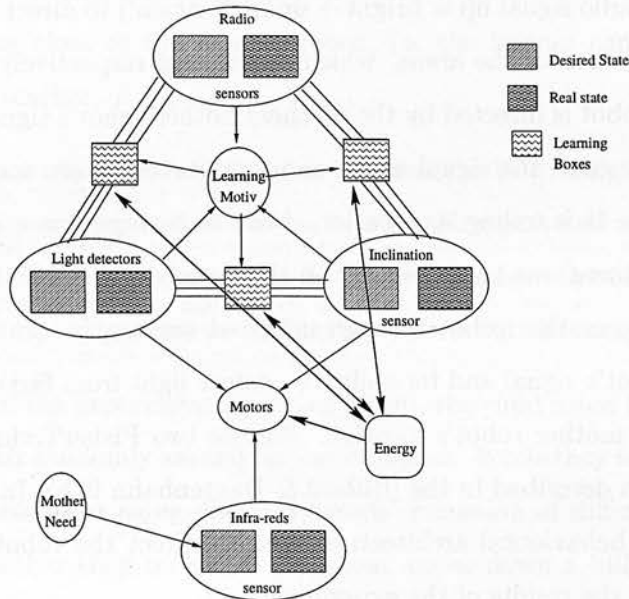


Figure 9.1: The behavioural architecture of the mother-child scenario.

The architecture's functioning is based on the conservation of an *energy* parameter  $E$ , which is a fixed quantity to be shared between two possible actions, activation of the motors and learning (i.e. update of the network connection parameters). The energy taken by each action is equal to the value of their corresponding motivational factor  $M$ , which is a function of the motivational factors associated with other sensor systems:  $E \geq M(\text{motors}) + M(\text{learning}) \simeq M(\text{IR}) + (M(\text{light}) + M(\text{inclination}) + M(\text{radio}))$ . The motivation for the activation of the motors is determined by the

motivation associated with the IR sensors, which corresponds to seeing the other robot. This is used to regulate the child robot's following of the mother robot: the farther away the mother, the bigger the motivation to move and the more energy given to the motors; thus the faster the robot runs. If the child does not detect the mother at all, i.e. minimum infra-red reflection, it runs at the maximum of its speed. The motivation for learning is determined by the motivation associated with light, inclination and radio sensors. Learning occurs when desired and real states of at least one of these three sensors do not match. The motivational factor of each sensor is proportional to the Hamming distance between the *desired state* and the *real state* of the sensor. The sensors desired states are produced by two built-in drives, the *mother need* and the *need for internal equilibrium*.

### Desired versus real sensory states

We distinguish between *real* and *desired* sensory states. The *desired* state is determined by a built-in behaviour, say an instinct. It represents what the agent wishes to sense but not necessarily what it really senses, which is the *real* sensory state. Comparison between desired and real sensory states is used as a motivation for acting. The agent acts (either moves or learns) in order to satisfy its desires if they do not match reality.

In our implementation, two built-in instincts are defined, the *mother need* and the *internal equilibrium need*. The mother need is a survival instinct, which corresponds to the wish to perceive a high frontal infra-red reflection:

$$M(IR) = \frac{|\text{desired state}(IR) - \text{real state}(IR)|}{250} = \frac{|250 - \text{measure}(IR)|}{250}$$

where 250 is equal to the maximal infra-red reflection, which corresponds to being very close to the other robot. This principle was successful because the mother robot was the only obstacle in the arena. The division by 250 allows to get normalised values.

The need for internal equilibrium provides the motivation for learning (and additionally for moving). It compares the value of current, i.e. real sensor input, to the predicted, i.e. retrieved, input for the light, inclination and radio sensor systems separately. By definition, the desired state of these systems is the retrieved one (updated at each processing cycle). Internal equilibrium is achieved when the real sensor states at both

ends of the associative network correspond to the predicted ones, i.e. desired and real states are equal. If this is not the case, a motivation for learning is activated. The motivation intensity is proportional to the normalised difference between the desired and real states, similarly to  $M(IR)$ , see above.

### Motivation versus Energy potential

The motivational factors compete for the distribution of energy,  $E$ , which has a fixed value to be shared between all possible actions of the robot, namely moving (activation of the two motors) and learning (updating of the structure of the learning architecture). The energy given to each action is proportional to the value of the corresponding motivational factor, which is determined by those of the sensors.

In our experiment, the desire for an infra-red detection of the mother produces a motivation for moving. Motors left and right are activated proportionally to the motivation. The robot runs in the direction where the infra-red sensor reading is the closest to the desired value, which corresponds to a high infra-red measurement.

As mentioned previously, the instinct for internal equilibrium creates a motivation for learning. In addition, it also provides a small motivation for moving. This was done, as it was considered that small movements in parallel to the update of the learning could be advantageous in some cases in order to cope with the experimental noise. Therefore, the motivational factor associated with the motor state is a function both of the motivational factor of the infra-red sensor (mother need) and of the motivational factor of the learning:  $M(motors) = M(IR) + \frac{1}{2} \cdot M(learning)$ .

Because the *mother need* related to the survival of the individual, its motivational influence on the motors' activation is privileged over the one of the *need for equilibrium* in the energy sharing process. The activation of the motors due to the *mother need* can consume up to a maximum of 80% of the energy available. The remaining energy (at minimum 20%) is divided up by the actions to satisfy the need for equilibrium, that is in first instance learning and in second instance motors' activation. Thus,  $E \geq M(IR) + 2 \cdot M(learning)$ , where  $M(IR) \leq 0.8 \cdot E$  and  $M(learning) \leq 0.2 \cdot E$ , and therefore  $M(motors) \leq 0.9 \cdot E$ .

This setup produced the behaviour designed for the mother-child scenario. When the child is moved away from the mother, the motivational factor produced by the 'mother need' activates the motors at the maximal speed, which corresponds to using 80% of the energy available. In the meanwhile, the mother sends radio signals corresponding to her position in the arena (on the hill or on the plane). The child decodes the mother's signals by retrieving the light and inclination measurements associated with the radio encoding of the signal. The retrieved measurements differ from that currently measured by the child, as it is not on the same location as the mother, therefore not satisfying the need for equilibrium and consequently creating a maximal (as all sensor states disagree) motivation for learning and consequently for moving (as the need for equilibrium increases both the learning and moving motivation factors). The energy consumed by the motors jumps to 90% of the energy available (10% more from  $M(\text{learning})$ ), so that about 10% of energy is left for learning. This was considered insufficient to activate an update of the associative network, as we had fixed a minimum threshold for the activation of the learning function to 15% of the energy. Consequently, the child does not update its knowledge of the vocabulary when it hears the mother's messages; this would in any case result in unsuccessful learning as the child's current perception of light and inclination are not similar to that of the mother. Instead, the child uses its understanding of the vocabulary to search for the mother, by driving its motors such as to go closer to the point which satisfies most its desired sensor state. That is, the motivational factors of right/left motors are proportional to the motivational factors of the right/left light sensors (similarly to infra-red sensors, see last paragraph). As a result, the robot search for light on its left, for instance, would produce a bigger activation on the right motor than on the left, consequently letting the robot turn toward the correct direction.

We programmed the robots in PDL, which is a language developed by Luc Steels at the Vrije Universiteit Brussel [Steels 92]. It is based on a parallel processing principle. Each process acts on global quantities that are updated only at the end of each cycle. The final value of the quantity is the result of the updates of each process. Consequently, the observable behaviour of the robot results from the combination of all processes.

The *following* behaviour of one robot by the other one was programmed as a com-

combination of ten processes acting on the motors and directed by the measurements of the bumpers and the infra-red sensor (see [Billard & Dautenhahn 97b] for the code). The *following* was bidirectional. Each robot was provided with a set of three infra-red sensors, on the back of the teacher and on the front of the learner, that enabled them to 'see' each other. The motors were given values, which control the motor current and hence the speed, proportional to the sensory measurement of the infra-red detectors. The closer the following robot, the slower it moves and reversely for the followed robot. The differences between the side detectors and the middle one used to control the two independent driving wheels. A set of six bumpers (frontal for the learner and in the back for the teacher) was used for close alignment. This resulted in a very efficient alignment and smooth following behaviour of one robot behind the other one.

### 9.1.3 Experimental set-up

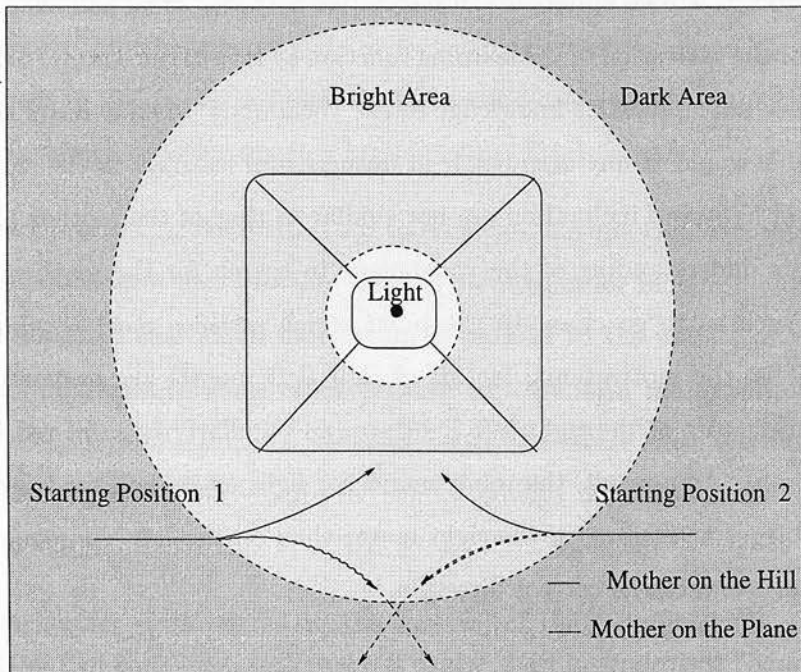


Figure 9.2: Schematic view of the 'Mother-Child' experiment.

We use the same set-up as for the first stage of the experiment, described in section 7.1.2. It consists of a rectangular arena, with in the middle a hill, on top of which is suspended a light bulb. Figure 9.2 shows a schematic view of the experiment with the hill and the light source on top of it. The light emitted by the light source is

directed downwards to the hill, forming a cone of light around the hill, brighter at the top and then decreasing progressively. The dark area corresponds to the part of the area outside the light cone. In this area, the robot does not receive direct light but only diffuse reflections from the walls. The light intensity received by the robot's two detectors is then approximatively the same.

This light intensity threshold is important because it determines the reactivity of the robot. The difference between desired and real states for each light detector determines the speed given to the motors. The difference in sensitivity between right and left detector determines consequently the direction of movement of the robot. Therefore, as long as the robot stays in the dark area, it can only move forward, because the difference in light intensity measured by its two detectors is too small to cause turning. However, once it has crossed the boundary between the two areas, it will turn either towards or away from the light source depending on the mother's messages.

#### 9.1.4 Results

The experiment begins with the child robot on the plane in starting position 1 (cf. figure 9.2) and the mother robot on the hill. Child and mother are separated by a distance bigger than 50cm, so that the child cannot detect the mother using its infrared sensors. As soon as the mother robot is activated (the robot is switched on using an external switch), it begins to emit the radio signal for 'hill'. As soon as the child robot is switched on, it begins to turn towards the hill. The scenario is then repeated by placing the child robot in the symmetrical position 2 (cf. figure 9.2) and so on for five different positions on each side of the hill. The situation is then reversed. The mother is put on the plane and the child placed again in one of the starting positions and so on. This was done in order to verify whether there was any bias in the robot's association of light-radio signals from any direction.

10 tests were thus done and recorded with a video camera. Tests showed that the experiment was successful, the robot's reactions were correct, but very sensitive to the difference between dark and bright areas. Outside the bright area, the robot keeps strictly moving forward, as explained previously. When entering the bright area only, the robot begins well to turn towards the right direction, that is either towards or away

from the hill, depending on whether the mother is on the hill or on the plane.

Once the region 'hill' is reached, the desire for light is satisfied. The robot slows down, however it does not stop until the desire for inclination and infra-reds are also satisfied. Thus, if it does not find the mother on the hill, it moves slowly around the hill, satisfying partially its desires for both light, upwards inclination and infra-red perception.

## 9.2 Collaboration among a group of robots

This experiment follows from the experiment reported in section 8.1, in which we studied transmission of a vocabulary from a teacher robot to eight learner robots. In this experiment, we study how the vocabulary learned in the first experiment can be used beneficially by the same group of nine agents. The experiment starts with all agents knowing a vocabulary of 31 words for describing the nine different colours of the coloured patches present in the environment and their locations in terms of 14 energy levels and 8 compass orientations. As the agents wander randomly around the environment, they learn the locations of the coloured patches by associating the sensor perception of the particular colour input with the coordinates of the agent's own position. When one agent has learned the location of a coloured patch, it can further transmit it via the communication channel (using the learned vocabulary) to all other agents. Each agent can thus learn the coloured patches' locations from listening to the speaker agent's talk and without actually coming across the particular location itself. This results in a speeding up of the learning of the whole population, which we study using two different scenarios for the information transmission. The transmission occurs as soon as one agent comes across a coloured patch location (one to many and long distance communication) or only when two agents come across one another (one to one and short-distance communication).

The experimental set-up was described previously in section 8.1. We describe, here, first the mechanisms used in the experiment to allow retrieval of the learned associations (understanding of the vocabulary) and then report on the results of the experiment.

### 9.2.1 Retrieval mechanism

A retrieval mechanism based on the notion of desired and real sensor state, similarly to section 9.1, was implemented to allow the robots to use their understanding of the vocabulary to learn new colour patches' locations from hearing another robot's saying so. The robots' predefined understanding of the vocabulary is set-up by fixing from the start their network's connections between the radio sensor and the colour, compass and energy sensor, such as to represent a vocabulary of 31 words to label 9 colours, 14 energy levels and 8 compass measurements. The robots communicate by exchanging information about objects' locations. One robot's message is composed of three radio signals (1 byte with 1 bit activated) for describing the object colour and its location in terms of compass and energy measurements. When one agent receives a radio signal sent by a second agent (which produces a pattern of activity in the radio sensor units), it decodes it by retrieving the correlated activity in units corresponding to the colour, energy and compass sensors. That is, the three radio signals retrieve a specific unit activity in each of the set of units attached to colour, compass and energy sensors. Learning of location of the coloured patches transmitted in this way results then from updating the connections between the coactivated units in the colour, compass and energy sensors (but not the radio sensor, see explanations below).

The sensor activity which is retrieved for the colour, energy and compass sensors is stored in a *desired vector state* (one for each sensor), which is different from the *real vector state*, which contains the current sensor measurement. Thus, the retrieved sensor activity does not affect directly the robot's behaviour, that is it does not participate in controlling the motors and radio activity (by retrieving the *real* unit activity in these sensors). However, in order to control the robots' speaking, i.e. to activate the radio emitter to transmit the objects' locations, in the simulations, a mechanism is implemented, which transfers the desired state into the current one. This transfer occurs either when the robot discovers an object or when it meets another robot and engages in an answer-question dialogue (see description of the results below). Learning of the objects' locations results then from association among desired sensor states and association among real sensor states. It is important to note that association among desired sensor units is done separately from association among real sensor units. Thus, in the

experiment, the hearer agent can either learn from the desired sensor units' activity (which correspond to the sensor perceptions 'transmitted' by the speaker robot) by associating mutually the active units in each sensor system or it can learn from the real sensor units' activity (its current sensor measurements), but it can not associate mutually its desired sensor state (speaker's measurement) with its real sensor state (its current measurement). By definition, radio measurements are never associated with other sensor measurements in this experiment. When they are produced by the speaker, they do not produce an input activity in the speaker's radio sensor but rather an output activity, which is then not forwarded to the DRAMA associative module (see section 4.3.2 for explanations); thus, no association occurs. When they are received by the learner, only the desired sensor states they produce are associated. Association between real sensor states was defined, so that all except the radio sensor state could be associated.

### 9.2.2 Results

We carried out 3 sets of simulation studies in which the nine robots use the vocabulary to transmit to each other the location (polar coordinates) of the nine objects (coloured patches) in their environment. In figure 9.3, we compare the results for three different learning scenarios. In the first case, the agents learn the locations of the coloured patches by making the associations when they travel across over a coloured patch; in this case, learning of the patches' locations results from each agent's individualistic search. In the second and third cases, the agents use their knowledge of the vocabulary to transmit to each other information about each patch location. In these two cases, learning results, then, from both individualistic search and from collaborative behaviour. An agent informs the other agent about a patch location by emitting a set of three signals, one corresponding to the colour type, and two corresponding to the 'words' for its location in terms of energy level and compass orientation. The other listener agents learn the new location by associating together the corresponding three sensor stimuli which have been activated by the reception of the signals (as explained in the previous subsection). In the second experiment, the agents transmit the location of the coloured patch as soon as they discover it, that is when they travel on it. The

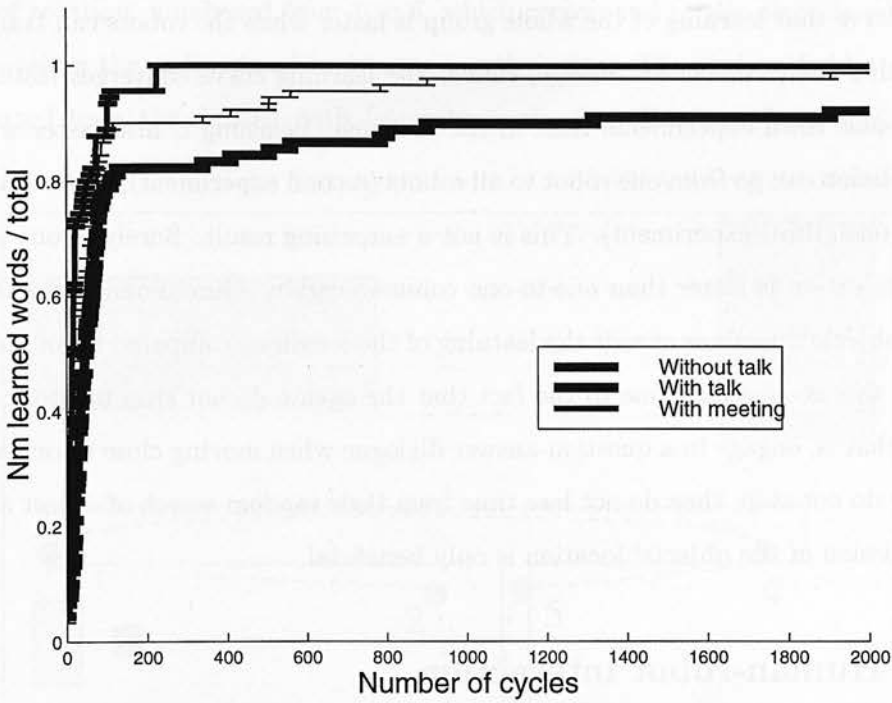


Figure 9.3: Speed of learning colour patches locations in three cases (mean value over all nine robots and all runs; error bar are standard deviations): as a result of each robot's individual exploration (-), by listening to another robot's speech which talks as soon as it finds a new location (-.-), through dialogue conversation when meeting another robot (....).

speaker robot's signal can be received by all robots in the whole arena (long distance communication). In the third experiment, the robots' transmission of information occur only when two robots meet (short distance communication). That is, when two robots are close enough to 'see' each other, they engage in a conversation; each robot speaks in turn, the robot with lowest numbering (robots are numbered as 1,2,..to 9) first, and asks the other for all coloured patches whose location it does not know yet. That is, one robot sends a signal for the colour type (question), which activates, in the receiver agent's 'brain', the corresponding patch location (if known). The retrieval process proceeds in two stages: first the corresponding sensor stimuli for compass and energy are retrieved and then by transitivity of the associations the corresponding radio signals are retrieved, which are further emitted by the robot as its 'answer' to the other robot's query. The dialogue ends as soon as the agents become separated by more than 20 cm. (e.g. when one agent turns in an opposite direction for avoiding an obstacle or another agent) or when they have enumerated all the objects.

We observe that learning of the whole group is faster when the robots can transmit to each other their current knowledge, that is the learning curve converges faster in the second and third experiments than in the first one. Learning is also faster when the transmission can go from one robot to all robots (second experiment), rather than from one to one (third experiment). This is not a surprising result. Surely, a one-to-many communication is faster than one-to-one communication. One-to-one communication of the objects' locations speeds the learning of the locations compared to an individual search; this is especially due to the fact that the agents do not stop to speak to each other, that is, engage in a question-answer dialogue when moving close to one another; as they do not stop, they do not lose time from their random search of object and thus transmission of the objects' location is only beneficial.

### 9.3 Human-robot interaction

This experiment follows the experiment of section 7.2.1, in which a human instructor taught a robot a four word vocabulary *North-South-West-East*<sup>1</sup> to distinguish between four orientations, relative to compass measurements, at 90 degrees from each other. The learning was done in a constrained environment, namely a rectangular arena consisting of four corridors at right angles to each other. The robot learned to associate four different radio signals sent by the instructor with each of the four orientations, i.e. compass measurement, it had when travelling in each of the four corridors.

In the experiment we report here the robot is taken out of the learning environment and allowed to wander freely inside the laboratory, which consists of a room of 20 by 6 meters with several pieces of furniture. By means of a portable radio transceiver (used for the teaching), the human instructor directs the robot along a fixed path which requires turning in each of the four compass directions learned previously in the cage. If the robot receives no signal from the human instructor, it simply goes straight forward, avoiding obstacles by means of its frontal infra-red detector and bumpers. Figure 9.4 shows a schema of the robot's path inside the laboratory. There are 6

---

<sup>1</sup> Note that we use the terms 'North/South/West/East' in order to give a simple image of the decomposition of the compass measurement into 4 quadrants. The magnetic field measured by the robot does not correspond to the earth's magnetic field, but to the magnetic disturbances created by the electronic machines in the laboratory, which luckily create a homogeneous field.

points of rotation, numbered from 1 to 6, which correspond to the place at which the robot receives the radio signal and consequently turns. The real path (dotted line) is distorted from the desired path (straight line). The drawing of the real path is schematic.

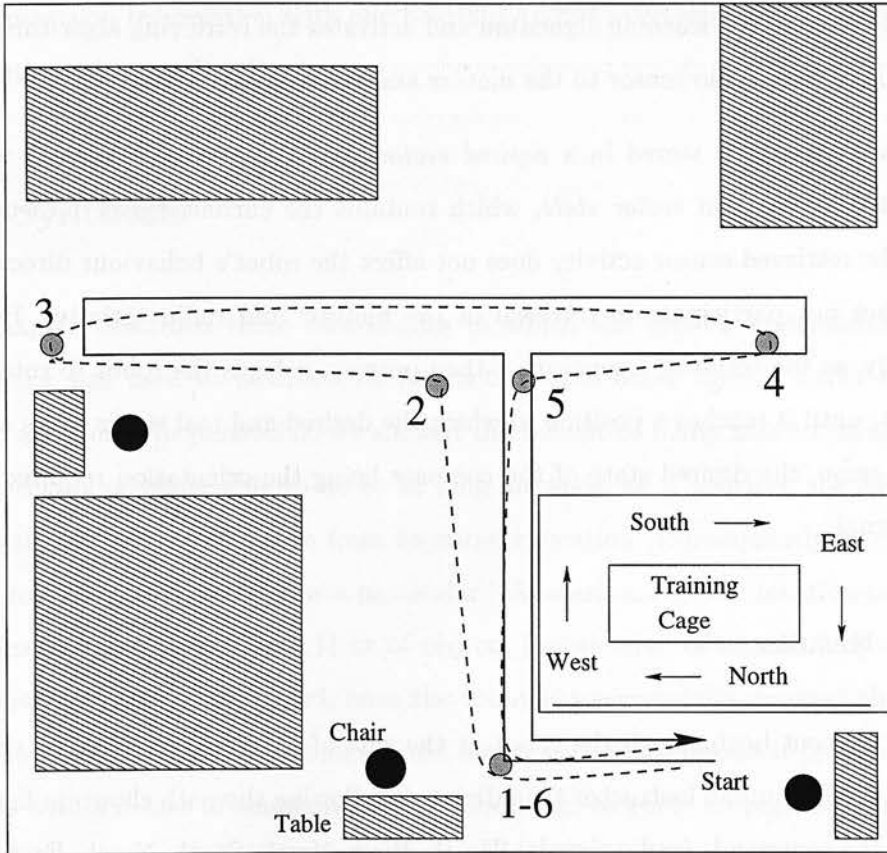


Figure 9.4: Schema of the robot's path inside the laboratory.

The experimental set-up was described previously in section 7.2.1. We describe, here, first the mechanism used in the experiment to allow retrieval of the learned associations (understanding of the vocabulary) and then report on the results of the experiment.

### 9.3.1 Retrieval mechanism

We implement the same retrieval mechanism, based on a comparison between desired and real state as described in section 9.1, to allow guiding of the robot's behaviour in this experiment. The same architecture (DRAMA training and retrieval processes) is used both for the teaching phase (reported in section 7.2.1) and the testing phase

(reported here). In order to distinguish between the teaching and ordering signals, we define a two bytes protocol, the first byte encoding the 'teaching/order' type of the signal and the second byte encoding the particular radio message. A predefined default connection in the network, which becomes active under the reception of the 'ordering' byte, switches off the learning algorithm and activates the retrieving algorithm for the connections from radio sensor to the motors and the compass sensor.

The novel activity is stored in a *desired vector state* (one for each sensor) which is different from the *real vector state*, which contains the current sensor measurement. Thus, the retrieved sensor activity does not affect the robot's behaviour directly, that is, it does not participate in retrieval of the motors' and radio activity. However, indirectly, as the result of a separate if-then process, it leads the robot to rotating on the spot, until it reaches a position in which the desired and real states of its compass are the same, the desired state of the compass being the orientation required by the radio signal.

### 9.3.2 Results

The experiment begins with the robot at the side of the Dodgem cage, as shown in figure 9.4. The human instructor then directs it following the path shown in figure 9.4, sending the commands (radio signals) 'North, West, North, South, North, East, South' (c.f. schema of figure 9.4) when the robot reached the position at which a turn in the corresponding direction was necessary. The guiding scenario was repeated ten times. The robot showed correct behaviour, that is it responded well to the orders when it received them<sup>2</sup> and was able to follow the whole path in each case. However, the path it followed was never the same, due to the fact that it was not always going straight, and it was shown to position itself to the required direction within 45 degree precision. The imperfect straight wandering is due firstly to imperfect gears and imprecise speed control (we do not use a proportional speed control system, which would make the robot's travelling smoother, but rather a two step system, i.e. half or full speed), which is unavoidable when using simple LEGO robots. The imprecise angle positioning is

---

<sup>2</sup> Correct radio transmission occurs in about 75% of the cases, that is if a signal is received then it is perfect, we could tell if the robot had received a signal by it emitting a sound.

due to the poor compass sensitivity, which can distinguish only between bearings of 22.5 degree spacing. As we teach to distinguish only between four main directions, it happens often that the robot associates two bearings with one radio signal. The robot can thus position itself toward the requested direction within at minimum 22.5 degree precision (association with one bearing) and at maximum 45 degrees precision (association with two bearings).

## 9.4 Synthesis

This chapter described three case-studies in which the robot's understanding of a vocabulary was used to facilitate its interaction with other agents, either other robots or a human. In particular, we showed the benefit of being able to communicate and especially of being able to do so at long distance, as it provides the agent with information it has not access to from its current location. Consequently, it can speed up the robot's search process for a particular information, e.g. the location of another robot (experiment of section 9.1) or of objects (experiment of section 9.2). Human-robot interactions are facilitated, once the robot has successfully mapped the human instructor's signals onto a set of significant sensor measurements, as it provides the instructor with a means of commanding the robot, e.g. to guide its path in an unknown environment (experiment of section 9.3).

An important aspect of these experiments lies in the fact that they show how easy it is to extract the information stored in the associative architecture (DRAMA). For the three experiments reported here, only simple mechanisms had to be added to the retrieval mechanisms defined in chapter 4 for the DRAMA network. In particular, we defined a general framework of retrieval mechanisms based on the notion of *desired state* and *real state* of the sensors and actuators of the robot. This was successfully used for on-line retrieval of the associations, which was further used to direct the robot's actions (sections 9.1 and 9.3) and the robot's learning (section 9.2).

Note that the experiments of sections 9.1 and 9.3, presented only qualitative, as opposed to quantitative, results, as these experiments were meant simply as an illustration of the use of the vocabulary. A quantitative evaluation of the DRAMA architecture's

retrieval performance is given in section 9.2 and in the experiments of chapter 11. Note also that the architecture's retrieval performance is evaluated continuously through all experiments reported in this thesis, as the robots' actuators (motors and radio emitter) activity is always controlled by retrieval of the motor outputs, given predefined sensor-actuator connectivity (or from learned connectivity for the radio emitter).

## Chapter 10

# Learning sequences of perceptions

This chapter describes an experiment in which a robot learns to extract spatial and temporal regularities in its perceptions (sensor measurements) when travelling in a highly regular environment. The experiment consists of letting two robots, a teacher robot and a learner robot, travel several times across a series of three corridors. The two robots follow closely each other, the teacher in front. The teacher robot emits one different radio signal, each time it turns from one corridor into another. The learner robot records the time series of its sensor measurements, namely the sequence of compass, light and radio measurements which results from the regularity of the two robots' travel in the corridors.

The experiment is carried out first in simulation and then in a physical environment. We first describe the experimental set-up and procedure and then report on the result of simulated and physical experiments. This work has been published in [Billard & Hayes 99].

### 10.1 Experimental set-up

Physical experiments are carried out in the Dodgem cage with the two LEGO robots, which were described in section 6.1.1. For these experiments, the rectangular arena is restricted to forming a series of three inter-connected corridors at right angles to each other. The corridors are delimited by walls on each side. An aluminium plate lies

in the middle of the second corridor. The same environmental set-up is reproduced in simulation. That is, the simulated environment conserves the proportionality of sizes between the robots, corridors and aluminium plate's dimensions of the physical environment. In addition, the simulated robots have a speed of movement equal to the mean one of the real robots (i.e. 1cm/sec). Consequently, simulated and real robots should perceive similar spatial and temporal regularities in their sensor measurements. In figure 10.1, we show a graphical representation of the simulated environment and a picture of the physical one. In the simulation, the teacher robot crosses over the aluminium plate in the second corridor, followed by the learner robot. On top of the picture, we show the dialogue of the robots. The teacher robot sends the signal '4' to label the object. The learner robot, which has not yet learned it, says nothing (output '0'). This image is a snapshot of one simulated run, taken during the first circling of the robots around the three corridors. In Appendix B, we show eight snapshots of the same simulation, which show the progress of the learner robot's learning. An animated gif of the complete simulation can be seen from an http site: <http://www.dai.ed.ac.uk/daiddb/people/homes/audeb/simull.html>

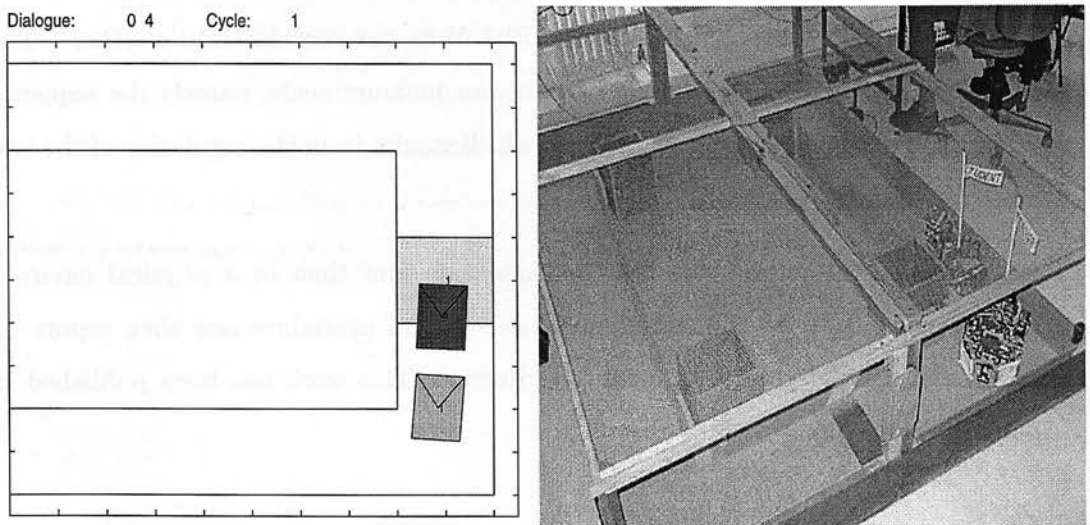


Figure 10.1: **From left to right:** a graphical representation of the simulated environment and a picture of the physical one.

Relevant sensors with which the robots are provided in this experiment, are a compass which measures bearings of 22.5 degrees, a pair of frontal light detectors for mutual following of the two robots by means of phototaxis (see section 7.1.1 for explanation of

the following routine), a pair of light detectors placed underneath their body to measure the difference of light reflectivity when travelling across the aluminium plate as opposed to travelling on the plane ground, and a radio transceiver for communication between the two robots. Teacher and learner robots are provided with the same sensor types (whose sensitivity might however be different, as two physical sensors are never the same). The simulated robots are provided with the same set of sensors as the physical robots. The robots are controlled by the DRAMA architecture. The processing of the sensor information in the DRAMA architecture is similar to that described in section 6.2. A detailed description of the sensor sensitivity, range and their binary transcription in the neural architecture is given in table 6.1. There are 43 units ( $2 + 1 + 6 + 16 + 10 + 8 = \text{Bumpers} + \text{IR} + \text{Light-detectors (front + below)} + \text{Compass-quadrants} + \text{Motor-states} + \text{Radio-sensor}$ ) in the network of the physical experiments. There are 53 units ( $1+8+8+8+3+16+3+6 = \text{Default-speed} + \text{IRs} + \text{Light-detectors (front + back)} + \text{Color-detector} + \text{Compass-quadrants} + \text{Radio-sensor} + \text{Motor-states}$ ) in the simulation. The encoding of the sensor and actuator information is described in detail in page 133. The short-term memory duration of unit activity is the same for all units and is equal to 600 processing cycles in both simulated and physical controllers.

## 10.2 The experimental procedure

The experiment consists of letting the two robots run along the corridors, starting from the bottom left entrance, the teacher in front of the learner. During a run, the robots travel the series of corridors ten times. The corridors walls in this experiment constrain the robots wandering to following one single path, which consequently creates temporal regularities in the robot's sensor perceptions. Given the ability of the DRAMA architecture for learning spatio-temporal regularities in the inputs (see chapter 5 for explanations), we expect that the learner robot will learn the time series of sensor measurement which it perceives during its travel along the three corridors.

While travelling along the corridors, the robots make different light and compass measurements; when crossing over the aluminium plate, the robots perceive an increase of light reflection in the light detectors, which they carry underneath their body. Because the corridors are placed at right angles to each other, travelling in each corridor

corresponds to measuring a compass value belonging to a different quadrant (which we call ‘South’, ‘West’ and ‘North’<sup>1</sup>). In addition, the learner robot receives three different radio signals while travelling in each of the three corridors. These signals are sent by the teacher robot; they represent labels for the two compass measurements made in the first and third corridors (signals ‘South’ and ‘North’) and for the increase of light, measured in the second corridor when the robot crosses the aluminium plate (signal ‘object’). Assuming that the robots travel in the corridors with the same average speed from one circling to the other, we expect the learner robot to perceive the following series of sensor stimuli when travelling along the three corridors: *Radio signal ‘South’, compass measurement ‘South’, compass measurement ‘West’, radio signal ‘Object’, measurement of light increase, radio signal ‘North’, compass measurement ‘North’.*

### 10.3 Results

Table 10.1: Comparison between results of simulations and physical experiments. See text for legend.

	<b>Simulations</b> Mean & Std	<b>Physic. exp.</b> Mean & Std
Ratio of cf	0.62 ± 0.10	0.68 ± 0.03
Nm words/total	1.0 ± 0.0	1.0 ± 0.0
Nm examples	583 ± 20	176 ± 15

Ten and five runs were carried out with the simulated and physical environments respectively. Each run started with the robots aligned one behind the other and placed at the bottom left entrance of the series of corridors. A run consisted of the robots travelling ten times along the corridors, each time in the same direction. The success of the learning was evaluated through reading the values of the connection parameters (confidence factor and time parameter), as explained in section 7.1.3. Remember that the confidence factor gives a measure of frequency of activation of the connection, while the time parameter records the time delay between consecutive activation of the units linked by the connection. In this experiment, we expect to observe significant

<sup>1</sup> Note that the robots measure a distorted component of the earth’s magnetic field, due to the noisy magnetic emissions of the laboratory machines; thus the labels South, West and North do not always correspond to their usual meaning.

values for the confidence factors of the connections linking the units of the compass and light sensors to the units of the radio sensor. These associations should be such that they represent the correlations between each radio signal and its meaning, in terms of compass and light measurement. These associations are similar to those studied in the experiments of chapters 7 and 8, in which the robot learned to correlate different radio signals with different sensor measurements and actuator states.

In addition, we also expect to observe associations between all the units of the compass and light sensors, as a result of their regular consecutive activation during the robot's travels (successive compass measurements: South-West-North and light increase after West measurement). The record of the correlations between these units will be demonstrated by finding significant values of confidence factors for the relevant connections, compared to other connections, and by observing a convergence of the time parameter values of these connections over the training period to constant values associated with the different temporal offsets between the different events.

Table 10.1 gives, for each set of experiments, simulated and physical, the mean value and standard deviation of (1) the ratio between the confidence factor (cf) values associated with the correct word and the maximal value of cf attached to all words for a given sensor measurement, i.e.  $cf(\text{correctcorrelations}) / (cf(\text{correctcorrelations}) + \max(cf(\text{incorrectcorrelations}))$ ); (2) the ratio between the number of words learned at the end of a run and the total vocabulary; (3) the number of teaching examples given in a run. We observe that in both simulated and physical experiments, the confidence factor ratio is greater than 0.5, which means that the correct correlations have been made more often on average than the incorrect ones, hence that learning is successful. In addition, learning is stable as the standard deviation for the ratio is small, while the value of the confidence factor remains greater than the threshold of 0.5.

Figure 10.2 top shows the variation of the time parameter values for each correct connection between radio units (standing for a signal) and corresponding sensor measurement (standing for an object) during a run. Data are calculated as the difference of time parameter  $tp$  values at each time step, i.e.  $tp(t) - tp(t - 1)$ . We show the mean value over all runs. Left and right figures show the results of simulation studies and physical experiments. We observe strong fluctuations of the parameter values at the

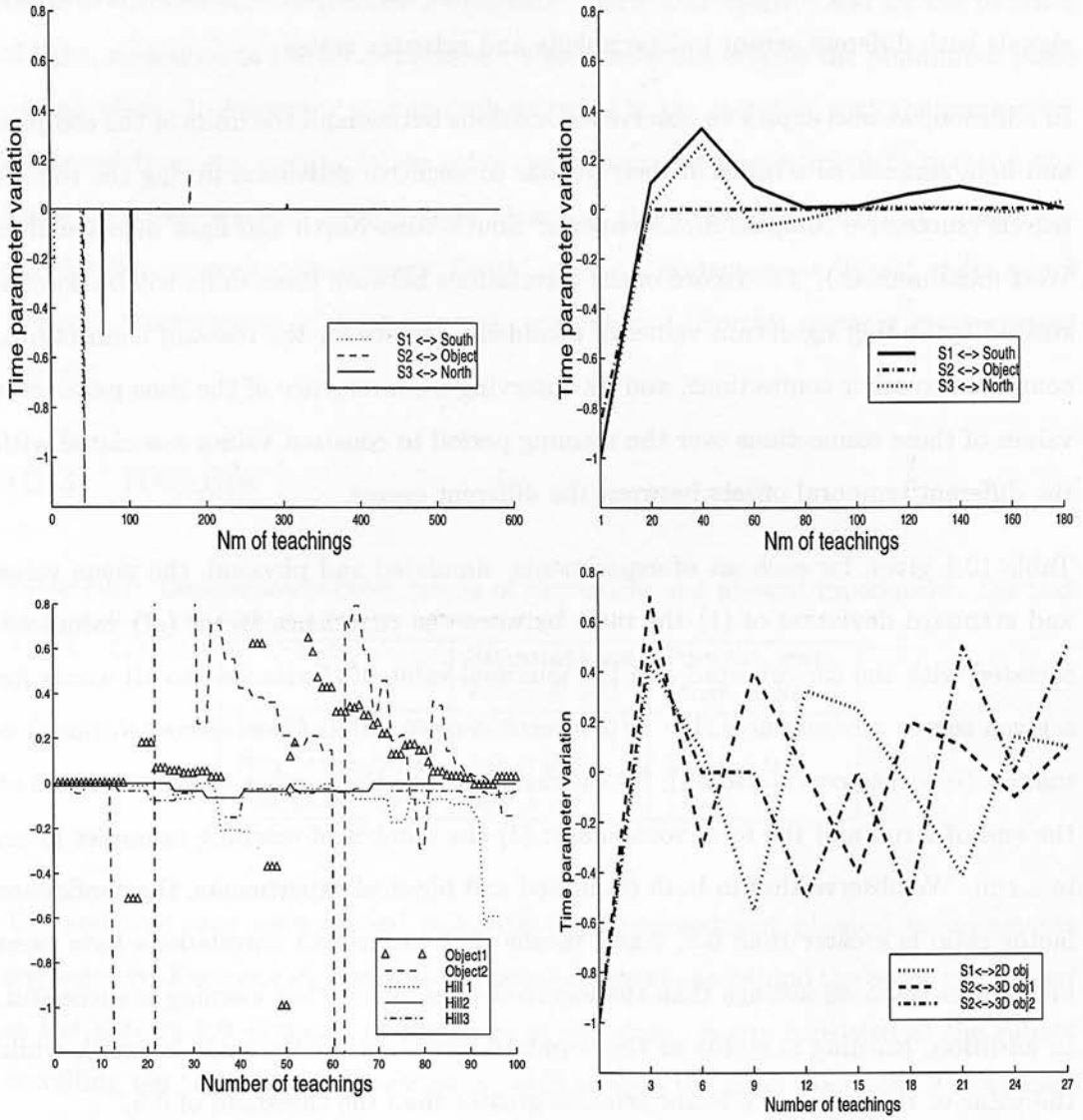


Figure 10.2: Variation of the time parameter values for each correct connection signal-object. Results of simulations and physical experiments (left and right columns respectively).

beginning of the run for the first third of the teaching (i.e., for about the first 100 and 60 teaching steps in simulations and physical experiments respectively). This corresponds to the learning phase; that is, the time during which the robot adjusts the time parameters to fit the training data. The curve then stabilises around the zero value, which means that the robot's network has converged to the correct time parameter values. Small fluctuations remain (at times 200 and 300 for the simulations and at times 140, 180 in the physical experiments), which are due to the imperfect travelling of the robots (simulated randomness in the simulator) and consequently the irregular travel timing. The consistency between the results of simulations and physical experiments demonstrate the good approximation of the noise made by the simulation.

In figure 10.2 bottom, we show the same curve for the results of simulations (left) and physical experiments (right) of section 7.2.2, to which we want to compare the result of the present experiment. In contrast to the top graphs, the bottom graphs (results of section 7.2.2) show stronger and continuous fluctuations during the teaching. In other words, the time parameters do not converge to fixed values in this experiment. This is no surprise as there was no pre-defined temporal regularities in the robots' behaviours or perceptions in this experiment. Recall that this experiment consisted of having teacher and learner robots wander freely in a rectangular arena, while following each other. The learning task was for the learner robot to associate 5 (simulations) and 3 (physical experiments) radio signals, emitted by the teacher, to the observation of 5 and 3 different objects respectively. For each run of this experiment, the robots started in a different position, randomly generated. In addition, in order to prevent any bias in the experiments, the robots' direction of travel was changed regularly during the run, so that the robot would, each time, follow a different path and would approach each object from different directions. Consequently, there was no temporal regularity in the robots' sensor measurements, that is in their sequential observation of objects, from one teaching episode to the next. It follows that the values of the time parameters in this experiment do not stabilise (as shown in figure 10.2 bottom) and that significant fluctuations (up to 10 times bigger than in experiment 2, outside values are out of the graphic) are measured all along the experiment.

### Retrieval of time series

Based on a quantitative and qualitative comparison between the connection parameter values at the end of each run, we assessed in the previous section the success of the learning. We observed that the robot had made the expected correlations between its different sensor measurements: it had made the correct correlations between radio signals and compass and light measurements, representing the ‘signal-meaning’ correlations; it had correctly recorded a temporal correlation between specific sensor measurements of compass and light sensors, that was demonstrated by the stabilization of the time parameter values in figure 10.2. It remains now to demonstrate that it has learned the correct sequence of stimuli, i.e., that it has learned the correct timing between each sensor measurement’s occurrence.

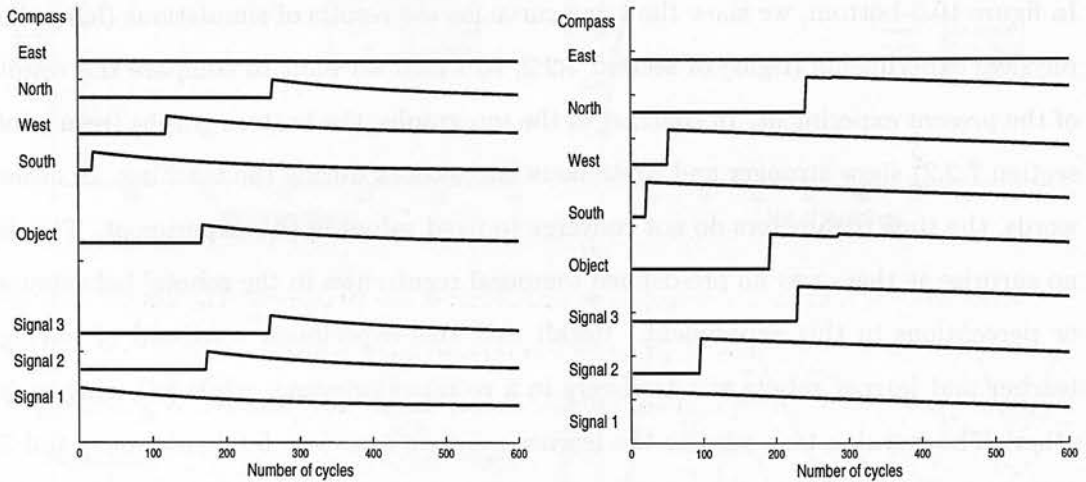


Figure 10.3: Retrieval of the sequence of sensor stimuli when presenting the radio signal for ‘South’ (signal 1) after learning of the time series. Results of simulations and physical experiments (left and right columns respectively). Y-axis shows the activity of the units corresponding to the three radio signals (signal 1-3), the light sensor (object) and compass sensor (segmented per quadrant).

In order to show that the robot has properly learned the correct sequence of stimuli, we run off-line a rehearsal of the DRAMA network, in order to produce the predicted sequence of sensor measurements, when taking the radio signal standing for ‘South’ as the starting activation (the signal ‘South’ is supposed to be the first sensor measurement the robot perceives when entering the first corridor). This amounts to retrieving for 600 cycles (this corresponds to the time needed by the robot to make one circle across the three corridors, about 10 minutes) all the network units’ outputs, starting

with all units' inputs and outputs set to zero apart from the input to the first radio sensor unit which is set to 1 (this unit corresponds to the signal 'South'). Rehearsal of the network was done, using the value of connection parameters with the highest ratio of word learning success obtained at the end of the run.

Results show that the starting activation of the radio unit 1, ('signal 1'), for 'South' is followed by a sequence of unit output activations in the radio, compass and light sensors. Figure 10.3 shows the sequence of sensor measurements, which results from the rehearsal process applied to the network, using parameters obtained in the simulation (figure left) and in the physical experiment (figure right). We observe that in both cases the radio signal for 'South' retrieves successively the compass value for South (1st corridor), the compass value for West (2nd corridor), the radio signal for 'Object' (signal 2), that is the patch (aluminium plate in the physical experiment) lying on the middle of the second corridor, the radio signal for 'North' (signal 3) and the corresponding value of compass. This demonstrates that the expected stimuli sequences (see the introduction of this chapter) have been correctly learned in both simulations and physical experiments. However, the exact time delay in terms of processing cycles between retrieval of each stimulus differs between simulation and physical experiments. Although the simulated set-up is quite similar to the experimental one (same proportion between corridor length and robots' size, same position and dimension for the patch), the two environments differ. The robots' wandering along the corridors in the real environment is very chaotic and varies significantly from one run to the next. In addition, the radio reception is not perfect in reality which results in significant variation in the time delay between reception of the signal and measure of the corresponding sensor stimuli<sup>2</sup>. These two facts account for the observed differences in time delay in the sequence rehearsal between simulated and physical experiments. Note that, as the time parameters represent the mean value of time delay between two units' coactivation in the network, the delay between consecutive pairs of unit activation in the rehearsal of the sequence does not represent the actual timing between the events in the physical experiment, but the time delay as perceived by the robot on average over all runs.

---

<sup>2</sup> The teacher sends the same signal about 10 times for a given stimulus in order to compensate for the loss in the reception; the variation in the time delay arises from the learner catching only the first or the latest signals.

## 10.4 Synthesis

This chapter reported on an experiment which demonstrated the ability of the DRAMA architecture for learning spatio-temporal regularities in an autonomous robot's sensor measurements. Experiments were done both in simulated and physical environments. Results of simulated and physical experiments were consistent in showing convergence and stability of the learning. This experiment was compared to the experiment of section 7.2.2, in which the time delay parameters did not converge to a stable value as there was no temporal regularities in the teaching occurrence in that experiment.

Correct learning of the timing of the perceptions' sequence, measured by the robot during its travelling along the three corridors, was demonstrated by running off-line a rehearsal of the sequence, given a starting sensor stimulus. Simulated and physical experiments were shown to retrieve the sequence in the correct order. However, different timings of perception occurrence was retrieved in each experiment, which showed that stronger variations of time delay were occurring in the physical experiment.

## Chapter 11

# Experiments with Robota, the doll robot

In this chapter, we report on another set of experiments, using a different robotic platform, namely a doll-shaped robot called **Robota**, for studying issues related to human-robot interaction. This new implementation of the DRAMA architecture aims 1) to verify the computational ability of the model for learning highly redundant combinations and sequences of inputs; 2) to see whether a mode of interaction based on imitation and communication could be acceptable to humans and whether it could be used to teach a robot; 3) to test whether using a robot which has a familiar and somewhat 'cute' appearance might prove appealing (rather than frightening) for people to interact with.

Robota is a robot, whose shape is similar to that of a doll, and which has the ability to learn, imitate and communicate. Using a simple phototaxis behaviour, the robot can imitate (mirror) the arm and head movements of a human demonstrator. We carry out experiments in which the robot is taught to perform different sequences of actions and to label these action sequences with different 'names'. In a second set of experiments, the robot is taught combinations of words, which form English proto-sentences, to describe its actions and perceptions of touch on different parts of its body. Experiments are carried out by different experimenters, other than the present author. Finally, we carry out tests with children of 5 and 6 years old, who teach the robot words to label different parts of its body and simple actions sequences. Note that the aim of these tests was not to provide a psychological study of the children's behaviour,

but rather to give a qualitative evaluation of the children' understanding and interest in the game we proposed.

This chapter is divided as follows. We first describe the doll robot's hardware and controller. The DRAMA control architecture is described in great detail in chapter 4 and 5, therefore we give here only a short summary of the particular implementation. In section 2, we report on the two experiments on teaching the robot sequences of actions and perceptions and on the tests carried out with children. We conclude this chapter with a brief discussion and summary of the experiments' results. This work has been published in [Billard *et al.* 98]

## 11.1 The doll robot

The hardware of the doll robot is made partly of plastic parts (arms, legs and head) which were taken out of a commercial doll and of LEGO pieces, which form the central part of the body. The robot controller is made of a micro-controller with 512k byte EPROM space and 128k byte Static RAM. The CPU is a Phillips 93C100 series 68000 compatible running at 30 MHz (same processor board as used with the LEGO vehicle, see section 6.1.1). Figure 11.1 shows pictures of the doll robot.

The robot has three motors, for moving each arm and the head separately. It has also a LEGO sound emitter, which is used to simulate the robot crying in the tests with children (see section 11.4). The robot is provided with five touch sensors (electrical switches), placed under the feet, inside the hands and the mouth, a tilt sensor which measures the vertical inclination of the body (it distinguishes between horizontal and vertical positions) and four infra-red (IR) detectors. Each infra-red detectors consists of an emitter and a receptor. Two of the IR receptors are placed on the robot's chest and measure the signals of the corresponding IR emitters which the demonstrator holds, one in each hand. The signal of each sensor is used to control each of the robot arms. When the demonstrator moves his/her left arm in front of the robot, the right detector on the robot is activated, which triggers the robot's right arm waving movements, i.e. lifting the arm up and then down with a fixed time interval between the two movements. The two other IR emitters are placed on the robot's ears, while the two

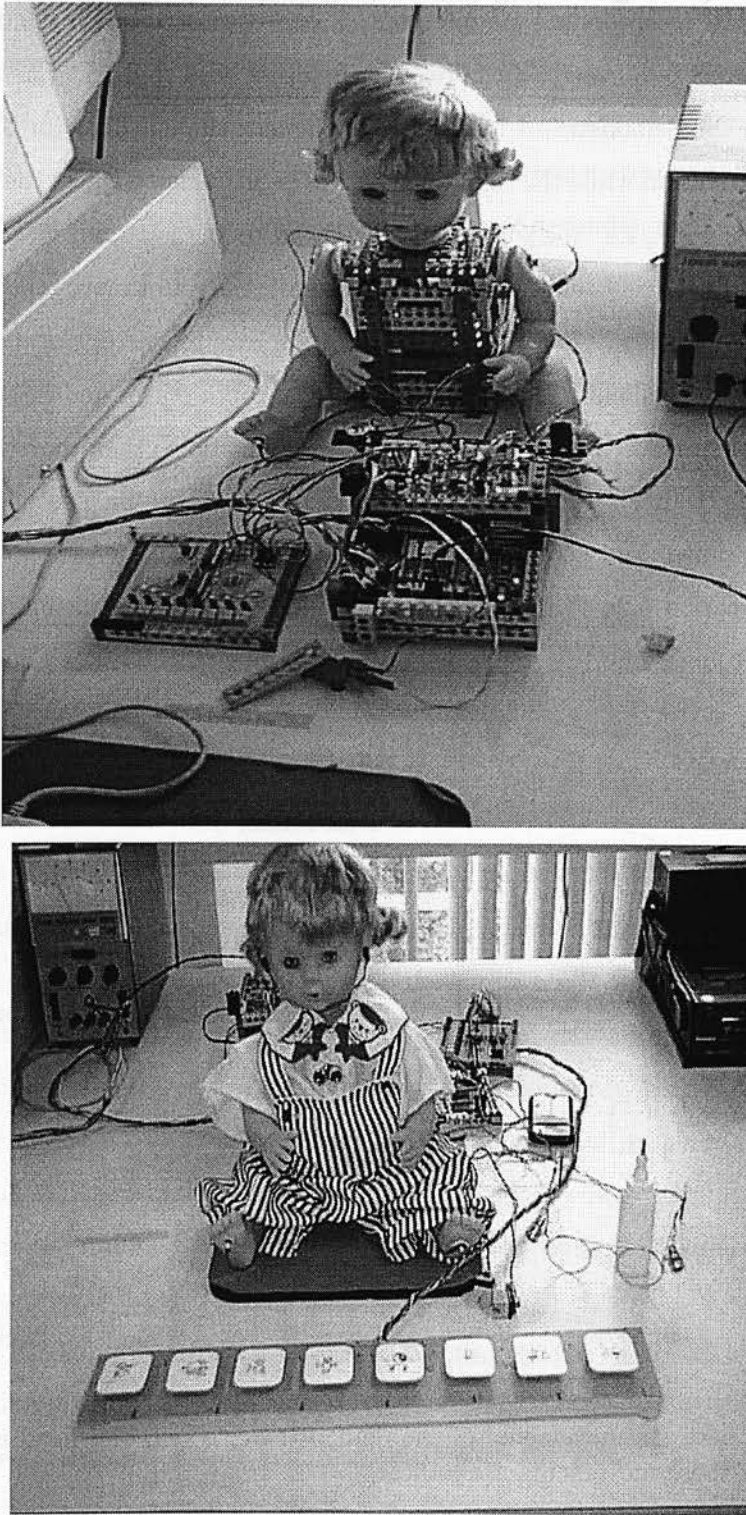


Figure 11.1: *Top:* The LEGO structure of the robot doll with the mini keyboard; *Bottom:* The doll dressed up, with, on the right side, the feeding bottle, the glasses and the infra-red hand sensors.

corresponding IR receptors are mounted onto a pair of glasses which the demonstrator wears. Phototaxis performed on the two IR signals is used to direct the robot's head. That is, when the demonstrator looks, e.g. to the left, the left detector on the glasses receives full activation while the right one receives none, which triggers the robot's head movement to the left, i.e. the robot turns the head to the left (and vice-versa for the right). After a fixed time delay (about half a second), the robot moves the head back to facing the demonstrator. As a result, the robot appears to mirror the demonstrator's arm and head movements. The code directing the robot's arm and head movements consists of if-then rules. For example, 'if receives right IR' - 'then move right arm up for 5 seconds' - 'wait for 10 seconds' - 'move right arm down for 5 seconds'. The rest of the robot's sensor and actuator processing is done by the DRAMA architecture as described in section 6.2. Figure 11.2 shows a schematic representation of the sensor positions on Robota's body.

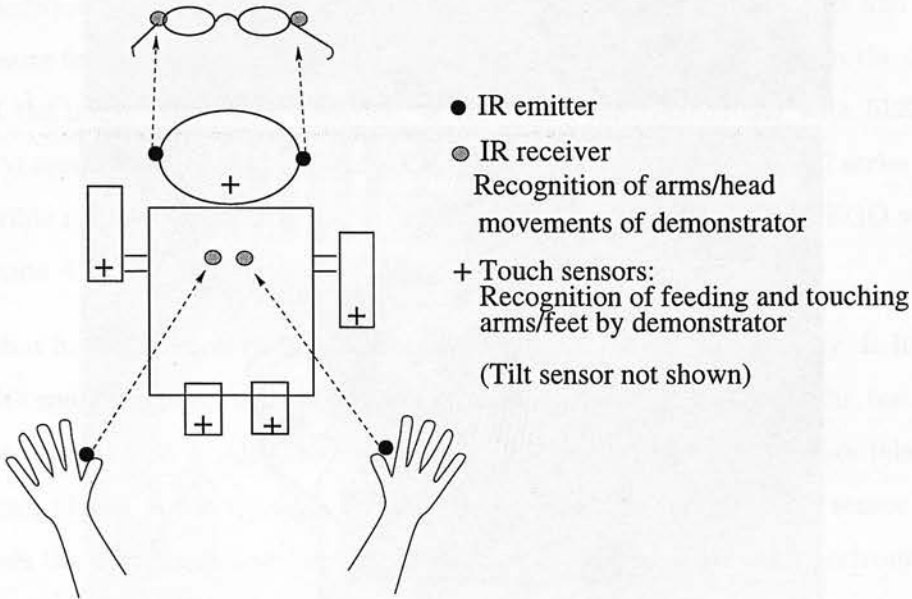


Figure 11.2: Schematic representation of the sensor positions on Robota's body. Correspondences between movements of the demonstrator's arm and head and Robota's recognition of these movements are indicated.

The robot is provided with a simple communication system, which consists of a keyboard and a loudspeaker (which is a commercial pocket recorder with 20 seconds recording time). We use, in fact, two keyboards containing 8 keys, a mini one, as shown

on figure 11.1 top (in front of the robot on the left), with a set of eight light bulbs on top of it and a big one as shown on figure 11.1 bottom. Big and small keyboards have the same functionality. The big one was used with the children, as it is easier to manipulate. The pocket recorder is used by the demonstrator to record sounds, spoken words or sentences (a sound slot lasts for 2 seconds), which correspond to the conceptual meaning the demonstrator attaches to each key of the keyboard. The demonstrator communicates with the robot by pressing the keys on the keyboard, each key representing a different word; the robot answers the demonstrator by reading back the sound slots of the corresponding words of the keyboard.

The structure of the robot (body, keyboards, glasses, sensors) were designed by the author. The robot's body was built by Auke Jan Ijspeert and the author. The sensors and keyboard circuits were built by the DAI electrical workshop.

## 11.2 The robot's controller

The robot is controlled by the DRAMA architecture, which was described in chapter 4. The architecture allows learning of spatial regularities and of time series. In the experiments of this chapter, we evaluate the network's ability for learning several time series of input patterns across multiple sensor-actuator modalities. In particular, the DRAMA architecture allows learning of overlapping sequences, that is sequences composed of one or more common patterns. This combinatorial property of the network is used in the experiments to teach the robot different sequences of actions (section 11.3.1), which are built up from the same four basic actions, and different word combinations (section 11.3.2), formed with the same basic eight words, to describe its actions with English proto-sentences. Proto means that the sentences are incomplete; the determinants are missing, e.g. the robot is taught sentences like *I touch left arm*, and the plural conjugations are not respected, e.g. the robot says *I touch left foot* for touching the two feet, see section 11.3.2. The sequential ordering of the words is respected.

The implementation of the architecture to control the robot learning is the same as in the experiments of chapters 7 to 10; see description of section 6.2. There are 18

units ( $6 + 8 + 4 =$  body switches + keyboard switches + head/arm actions) for the experiment 11.3.1 and 17 units ( $9 + 8 =$  encoding of actions and switches of table 11.2 + keyboard switches) for the experiment 11.3.2. Each unit is active when the switch or action it represents has been activated. In average, there are about 5 units active (e.g. 4 body actions and 1 keyboard switch or 4 keyboard switches and 1 action or 1 body switch) at a time during the experiments. The short-term memory duration of unit activity is the same for all units and is equal to 25 processing cycles in both simulated and physical controllers.

At each processing cycle, any change of activity in one sensor activates association of the novel sensor state with previously recorded changes of state in any other sensor systems. This results in a temporal mapping of different sensor states across all the robot's sensory modalities (keyboard switches, body switches, tilt sensor, infra-red sensors). Sensor states are associated to each other spatially through the update of the network connections' confidence parameters. The time delay between consecutive activation of two sensors is recorded in the connections' time parameters. It follows that the robot can learn time series of sensor activation, recording the real time delay between each sensor activation.

Retrieval of the associations is performed at each processing cycle. The output of the DRAMA architecture controls the flashing of the lights of the mini-keyboard, the reading of the sound recorder and the robot's arm and head movements. This is used to let the robot demonstrate its understanding of the language; that is, its correct association of different keyboard states with different body sensor states. After the robot has associated a sequence of movements with the activation of specific keys of the keyboard (section 11.3.1), pressing the key retrieves each motor movement in sequence, after the recorded time delay. That is, the robot repeats the dance pattern. Similarly, after the robot has associated a sequence of key activations with a specific body switch activation (section 11.3.2), pressing the body switch again retrieves each key activation in sequence, which is shown by lighting up the lights on top of each of the key (small keyboard) and by emitting the associated recorded sound.

The robot's imitative behaviour, which results from phototaxis on the infra-red sensor measurements, was implemented by predefining the network's connections between

infra-red sensors and corresponding actuators. Note that, in contrast to the experiments of chapters 7 to 10, here, the robot's movements are not controlled directly by the DRAMA output. The up-down and right-left movements are predefined as separate processes (as described in section 11.1), which are activated by the activation of the infra-red network units. Similarly, the random movements of the robot's head and arms in response to the child's touch and the crying behaviour, used in the experiments with children (see section 11.4), are also defined as separate rule-based processes, which are activated by the activation of the corresponding switch network units.

### 11.3 Experiments

The potential of the robot's learning controller (DRAMA architecture) at learning complex times series of sensor and actuator patterns was evaluated through two experiments, which we report in the following.

#### 11.3.1 Learning dance patterns

In this experiment, the robot was taught four to eight 'dance' patterns, where each dance pattern consists of a sequence of head and arm movements. We carried out five tests<sup>1</sup>, performed by three different persons. The results of each test is to be found in appendix C. The experimenter was free to choose the number of dance patterns and the number of movements of each dance it wanted to teach. As a result, none of the teaching was similar (there were dance patterns common to two teaching episodes but the complete sets of dance patterns were different). In addition, as each experimenter would move at different speed one could evaluate the performance of the DRAMA network to adapt to variation in the input timing. In figure 11.3 we show an example of the teaching of eight dance patterns.

In order to teach the robot, the demonstrator first performs herself the dance, moving sequentially her arms and head, which the robot immediately imitates in response to the infra-red reception of the sensors attached to the glasses of the demonstrator and

---

<sup>1</sup> In fact, we did a lot more tests, as the robot was presented at SIRS'98 and SAB'98 conferences. The robot was shown to learn correctly in all demonstrations done at these occasions. However, only the ten tests done in the laboratory were recorded.

the sensors which the demonstrator holds in her hands (see explanation of section 11.1). At the end of the dance, the demonstrator presses a key on the keyboard. The robot associates the pressing of the key (which activates the corresponding sensor unit) with the complete sequence of actions which it has memorised. It updates the parameters of the connections linking the sensor unit corresponding to the key to the actuator units which have been activated by the different actions of the dance (recall that each DRAMA unit keeps a memory of its activation for a fixed delay, see explanation in section 4.3.5). The demonstrator repeats the same process for each key, teaching a different dance pattern each time.

During the training, the demonstrator verifies if the robot has correctly learned the dance patterns, by pressing again the keys. This activates retrieval of the associated action sequence by reading backwards the DRAMA connections from the keyboard switches inputs to the robot's actuator outputs (as described in section 11.2). Note that because the time delay between each action is recorded in the network's time parameters, each action of the sequence is retrieved after the time delay as observed during the demonstration.

The robot learned correctly to perform all the different dances. Each dance pattern was learned after 1 to 3 trials, depending on the precision of the demonstrator's movements. The imitation of the head movements was tricky to obtain, as the reception of the infra-red sensor on the demonstrator's glasses was very imprecise. This is due principally to the fact that the infra-red beam is very narrow and that the heads of the robot and the instructor were not at the same height, hence the imprecise focus of the beam. It took between 10 to 15 minutes to teach the robot eight sequences. An example of teaching of the eight sequences of figure 11.3 has been recorded with a video and is stored in the DAI archives. In appendix C, we give the tables of results of all experimenters' teaching examples, with a list of the taught dance patterns and the time needed for the teaching.

Learning of the association results from a statistical process. Correct association between the key and the corresponding action is learned once this has been recorded more often than other combinations relative to a given threshold. Teaching errors, such as pressing the wrong key or showing the wrong actions, does not result in failure

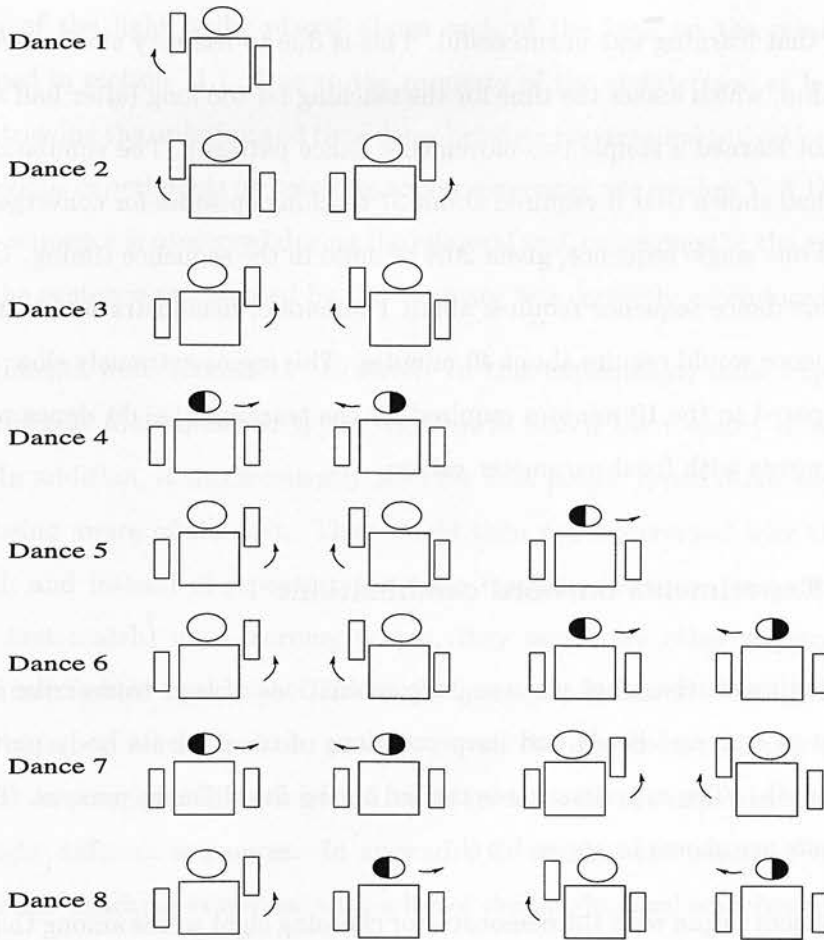


Figure 11.3: Examples of a teaching of eight dance patterns

of the learning but only delays its success. The correct sequence should be repeated a number of times until the incorrect associations are discarded. Note that in this experiment the success of the learning depends strongly on choosing correctly the value of the short term memory (memory of the unit activation), so that it is long enough to allow association of the complete sequence and short enough to distinguish between two sequences. This is especially tricky, when the timing performed by the different experimenter varies. However, all experimenters had no problem in adapting their teaching speed to match that of the robot. We chose a memory duration of about 30 seconds (25 cycles). In section 7.2.2, we reported on a quantitative evaluation of the influence of the memory parameter on the success of the experiment and in section 5.2.1 we proposed an algorithm for on-line tuning of this parameter together with the learning of the connection parameters. We tried to use the algorithm in this experiment, but

found out that learning was unsuccessful. This is due to the very slow convergence of the algorithm, which makes the time for the teaching far too long (after half an hour, it had still not learned a simple two-movements dance pattern). The simulations of section 5.2.1 had shown that it required about 37 teaching episodes for convergence of the learning of one single sequence, given 20% of noise in the sequence timing. Given that teaching one dance sequence requires about 1 minute of demonstration, learning each dance sequence would require about 40 minutes. This seems extremely slow, especially when compared to the 10 minutes required for the teaching of eight dance patterns in the experiments with fixed parameter values.

### 11.3.2 Experiments on word combinations

In this experiment, the robot was taught combinations of keys to describe its actions (movement of arm and head) and its perceptions of touch on its body parts (hands, feet and mouth). The experiment was carried out by five different persons. The results of these tests are shown in appendix C.

The experiment began with the demonstrator choosing eight words among the proposed eleven, which were *I, You, move, turn, touch, arm, foot, head, left, right, up/down*. The words were written on stickers, which the demonstrator put in front of each key of the keyboard, to help her to remember the meaning of each key during the training. The demonstrator then taught the robot to describe its actions (moving arms and head) and its perceptions of touch (pressing of the switches) on its feet, arms and mouth, by associating them with a combination of keys. These combinations of keys represented English proto-sentences. An example of teaching was *I move right arm, I turn head left, You touch mouth, You touch left/right foot* (see figure 11.4). In order to teach the robot, the demonstrator first pressed the relevant switch or activated the robot's arm or head by moving the corresponding sensors, and then pressed the corresponding keys on the keyboard, one after the other, so as to preserve the sequential order. During training, the demonstrator verified whether the robot learned the sequence, by pressing the switch again or activating the robot's arm or head. This prompted the robot to retrieve the key sequence, by reading backwards the DRAMA connections from the sensor-actuator inputs to the keyboard switches' inputs, which resulted in the sequential

activation of the light bulbs placed above each of the keys on the mini-keyboard, as explained in section 11.1. Due to the property of the architecture at learning and further retrieving the ordering and time delay between two sensors' activation (as shown in the previous experiments on teaching action sequences, see section 11.3.1), the order of the key sequence is conserved during the retrieval and, consequently, the grammatical order of the sentence represented by the sequence was correctly reproduced.

All experiments were successful. However, in this experiment, some experimenters found it difficult to adapt their typing rhythm to match the memory duration of 30 seconds. In addition, it was frequently the case that people typed other keys in error, without being aware of the fact. They would then not understand why the learning had failed, and instead of repeating the same teaching sequence (so as to erase the incorrect first match) until learning occurs, they would try other sequences, which would confuse further the robot. This resulted in a much slower learning rate than in the experiment on teaching dance patterns, even though the complexity of the task was similar in principle. It took from 5 to 20 minutes for the experimenters to teach four to eight different sequences. In appendix C, we give the tables of result of all experimenters' teaching examples, with a list of the taught word combinations and the time needed for the teaching.

Learning in this experiment results from the same principle as in the previous experiment on teaching dance patterns. These two experiments differ only in the direction of the association across the sensor space. In the dance experiment association was done from the keyboard units to the actuator (infra-red) units, while in the second experiment association was done from the actuator and body switch units to the keyboard units. Because actuator, keyboard and body switch information are processed similarly in the associative memory (training and retrieval processes are the same for all units, independently of the original type of the information they represent), correct learning and retrieval of the taught sequence in the second experiment results from the same property of the retrieval mechanism, as described for the first experiment.

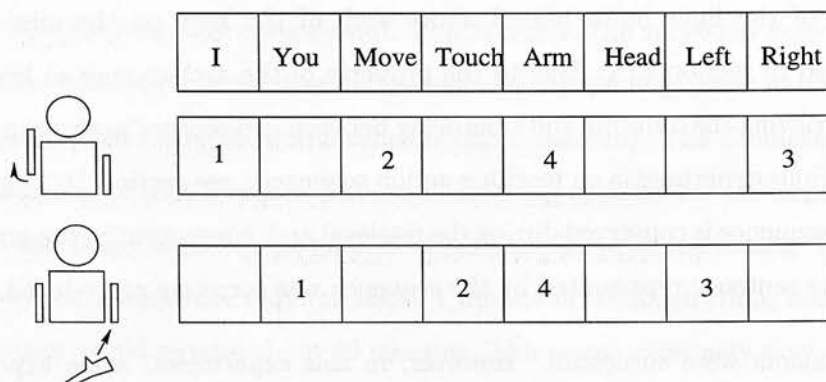


Figure 11.4: Examples of teaching of two word combinations for lifting up the robot's right arm and pressing the switch on the robot's left arm.

### Learning inference

An interesting result of this experiment was that, although the robot had been taught only complete sentences, it had learned implicitly the conceptual meaning of each of the words of the sentence. For instance, after having taught the two sentences *you touch left arm* and *I move right arm* for pressing the switch on the robot's left arm and activating the robot's movement of the right arm, pressing simultaneously the switch on the robot's left arm and moving its right arm results in the robot lighting up only the key corresponding to the word *arm* (note that if the demonstrator does each action separately, the robot correctly retrieves the complete sequence of words for each, as explained in the previous section). This results from the threshold-based retrieval mechanism, which requires that all activated units agree in their vote for another unit activation (see section 5.2 for explanation). In the previous example, the arm unit is the only unit for which both the switch and actuator units agree. Similarly the concepts of left or right can be retrieved if one moves the robot's left arm together with touching the switch on the left foot, after having taught the two sentences *I move left arm*, *you touch left foot*.

In table 11.1, we show the robot's learning development alongside the teaching. The robot is first taught the sentence for the action of moving its right arm (that is, the teacher moves her arm left up and down and the robot repeats the movement by lifting its arm right up and down; then the teacher types the sequences of keys corresponding

Table 11.1: Example of sentence teaching

Activated sensor	Taught sentence	Retrieved sentence
Move right arm	<i>I move right arm</i>	
Touch left arm	<i>You touch left arm</i>	
Move right arm + touch left arm		<i>arm</i>
Touch right foot	<i>You touch right foot</i>	
Touch left foot	<i>You touch left foot</i>	
Touch right foot + left feet		<i>You touch foot</i>
Touch mouth	<i>You touch head</i>	
Touch right arm		<i>You touch</i>

to the sentence). Following the same procedure, the teacher then teaches the sentence *You touch left arm*, after having touched the switch placed in the left hand of the robot. Next, when the teacher simultaneously lets the robot move the right arm and touches the robot's left arm, the robot retrieves the word *arm*, which is the only invariant for the two events. Note that, if the teacher activates each sensor (moving the right arm or touching the left arm) separately, the robot retrieves the complete sentence associated with each event. The teaching continues with the teacher teaching the robot the two sentences *You touch right foot*, *You touch left foot* for touching separately the switches placed on the robot's feet. Then, when the teacher presses simultaneously the two foot switches, the robot retrieves the sentence *You touch foot*, where the three keys standing for *You*, *touch* and *foot* are the only invariants for the three events, the words *left* and *right* appearing only for each event separately. Next, the teacher teaches the robot the sentence *You touch head* when inserting the baby bottle inside the robot's mouth, which activates the switch in the robot's mouth. When the teacher presses the switch on the robot's right arm (for which the robot has yet been taught no sentence), the robot's retrieves the sentence *You touch arm*. This results from the fact that the activation of each sensor (arm, foot, mouth) and action (moving arm and head) of the robot is encoded, so as to encapsulate some of the body symmetries. In table 11.2, we show the encoding. For instance, activation of the left foot corresponds to activating three units, which represent the fact that the switch is a sensor, placed on the foot on the left side of the body. Activation of the right arm movement activates three units, corresponding to the notion of actuator, arm and placed in the right side of the body. Similar decomposition of proprioception seems to exist in the human body [Gross 96], as we can distinguish between perceptions on right and left sides of the

body and between perceptions coming from upper or lower limbs (head, arms, legs). Proprioception of touch and of actions is also different. We can also perceive a change in our body inclination.

Table 11.2: Encoding of the doll's sensors and actuators.

Performed action	Actuator	Sensor	Arm	Foot	Head	Down	Left	Right
Move right arm	1	0	1	0	0	0	0	1
Move left arm	1	0	1	0	0	0	1	0
Move right head	1	0	0	0	1	0	0	1
Move left head	1	0	0	0	1	0	1	0
Touch right arm	0	1	0	0	1	0	0	1
Touch left arm	0	1	1	0	0	0	1	0
Touch right foot	0	1	0	1	0	0	0	1
Touch left foot	0	1	0	1	0	0	1	0
Touch mouth	0	1	0	0	1	0	0	0
Rock body	0	1	0	0	0	1	0	0

This decomposition of the robot's perception leads to the fact that, although, in our example of table 11.1, the robot has not yet been taught a sentence to describe the action of moving its right arm, the network units corresponding to this action have already been activated through previous teaching (e.g. the activation of the left arm and the touching of the right arm switch at steps 1 and 2) and, thus, the connections linking these units to the keyboard units have already been updated. By elimination across the so far taught combinations, one can infer why the activation of the three units for the right arm movement results in the activation of the three keyboard units, corresponding to the words *You*, *touch* and *arm*.

From the eight word combinations which the demonstrators were given, they were able to retrieve the concepts for *I move* associated with activation of the actuators, *you touch* for pressing of a switch, as *arm*, *foot*, *left* and *right*. In addition, the robot had learned implicitly the sequential order of word occurrence, by learning the time delay duration (short or long) before activation of the corresponding key unit. For instance, in the example of table 11.1, the robot inferred correctly the sentence *You touch arm*; that is, it retrieved the correct word combination, in the correct order, although it had not been taught this sequence previously. Note, finally, that the architecture is not limited to learning only eight dance patterns or combinations of eight words. Only the

hardware, that is the keyboard, was limited to contain only eight keys. There is no a priori limitation on the number of words or patterns it could learn; this depends on the number of inputs to the network, which fixes the maximal capacity of the network.

## 11.4 Tests with children

In the following section, we give a narrative account of the tests we carried out with children. It is by no means meant to be a psychological analysis of the children's behaviour, but rather it aims to give a qualitative assessment of the children's appreciation of the game. Through these tests, we wished to evaluate the potential of a robotic doll as a toy for children and in particular to evaluate the children's understanding and interest in the particular game we proposed.

The robot doll was tested with nine children of five years old (six boys and three girls<sup>2</sup>). The scenario of the game the children were asked to play consisted of the following:

1. The child is presented with the doll as if it were a young baby, who does not know how to speak yet. The task of the child is then to teach the doll to speak. The child is shown the keyboard; the meaning of each key is described by an icon on top of it. A word is prerecorded in the pocket recorder for each key: (from left to right) 'food', 'rocking', 'hello' (right arm up), 'Yeah' (two arms up), 'No' (side movements of the head), 'hand', 'left foot', 'right foot'. The child is explained that (s)he has to teach the doll the words on the keyboard.
2. The child is shown how (s)he can direct the movements of the doll's arms and head by moving his/her own arms and head, while holding the two infra-red emitters in each hand and wearing the glasses (the children liked very much wearing glasses!).
3. After five minutes of play, the doll begins to cry (emitting a sharp continuous sound). The child is told that the sound meant that the doll is crying and that it cries because it wants something. "Do you know what the baby-doll wants?" (All children responded immediately that "it wants to eat"). The child is then

---

<sup>2</sup> This unequal proportion of boys to girls was involuntary.

shown the baby feeding bottle and asked to feed the doll (which they did with great pleasure). After some seconds of feeding, the baby stops crying and the child is congratulated to have found out what the baby-doll wanted.

4. The child is then asked to teach the doll how to say that it wants to eat. The child is shown how to teach the doll using the keys on the keyboard (first feed, then press the first key on the left). The child is asked to feed the doll again, which prompts the doll to say 'food' (retrieving of the correct association between the mouth switch and the key, which activates reading the corresponding word on the recorder). The child is then congratulated because (s)he has managed to teach the doll.
5. The child is then asked to try to teach the doll another word of the keyboard. (Children spontaneously tried to teach the words for hand or left/right foot.) The child is asked to teach the doll to say 'hello'. For this, (s)he must first wave the right arm, which prompts the doll to wave its left arm, and then press the corresponding key on the keyboard. The child is then shown that if he/she presses the key 'hello' again, the doll waves its arm and then say the word 'hello', showing that it has well learned the correspondence between the key, waving the left arm and saying hello. The child teaches the robot further to wave both arms and say 'yeah' and to shake the head and say 'no'. (The experience had to be repeated two or three times before the child would understand that the robot was actually learning the relationship between actions and saying the words).
6. After five minutes of play, the doll begins to cry again and says 'food'. The child feeds the doll and the doll stops crying.
7. The child carry on teaching other words on the keyboard.
8. The doll begins to cry again. (All children immediately said that the doll wanted to eat and then tried to feed it. However, the doll did not stop crying.) The child is then told that the doll probably wants to be rocked. The child is shown how to gently rock the doll. After a couple of rocking movements, the doll stops crying. The child is then asked to teach the doll to say the word 'rocking', by first rocking the doll and then pressing the corresponding key.

9. After five more minutes, the doll begins to cry again and (usually) says 'food'.

The child feeds the doll and the doll stops crying. End of the game.

At the end of the game, all children were asked the following questions: Did you like the game? To which they all, apart from one, answered a definitive 'yes'. The child who did not like it explained to his father that he did not like it because he had difficulty understanding the words the doll was saying and pressing the switches on the doll's body (critiques related to the hardware rather than to the game itself), he however very much liked feeding the doll and having it moving when pressing the keys, as it reminded him of his 'Tamagochi' (Japanese children's game; the child was Japanese). Do you have a doll at home? Only three of them (boys) said they did not have one. Is this doll different from another doll? They all said no (probably thinking of the shape of the doll, see answer to next question). Do you like this doll? They all said 'yes'. Is this doll better than other dolls? All children, apart from the child who did not like the game, said 'yes'. Why? Answers were, e.g., 'because it can do things', 'because it moves/speaks'. Is the doll a boy or a girl? 'a girl' (all). The doll has no name, which name would you give to it: they gave names of some existing dolls or of some of their own toys.

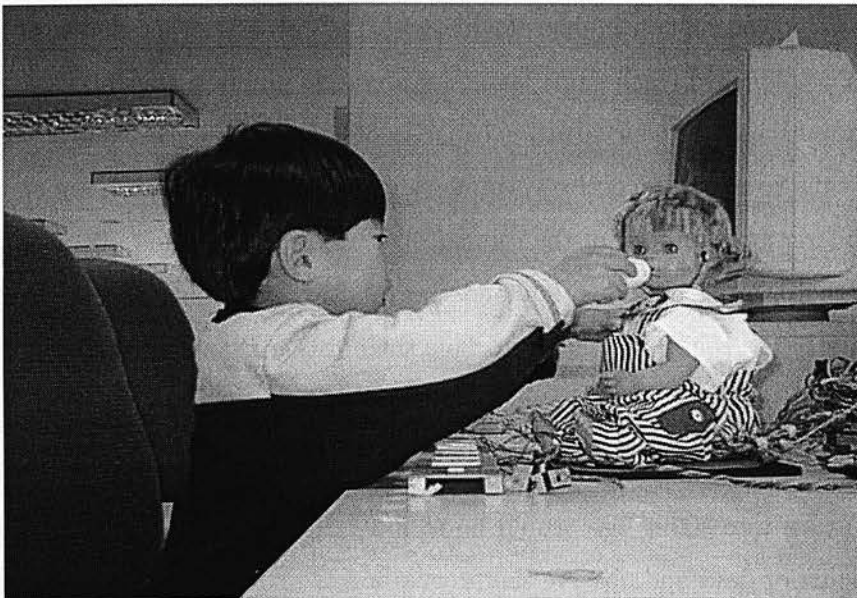


Figure 11.5: Kyo Takiguchi 'feeding' Robota

The children took a real pleasure in the fact that the doll was responding to their touching her. Little behaviours such as small random movements of the head and arms had been implemented as a reaction to the child touching the switches on the doll's hands, feet and mouth. The robot would then appear to sometimes wave the head from left to right when the child tried to feed it, as if it did not want to be fed. The children liked also the fact of being able to command the robot's arms and head movements, although they took less pleasure in this than in touching the robot. One reason is probably that they do not control well enough their own movements yet and thus had problems to direct correctly the robot's movements. For the robot to respond correctly to head and arms' movements, you had to make very precise movements (lift the arms straight and in a narrow region around the middle of the robot's body, and small slow sideways movements of the head). The children had problems to achieve such precision. It was not clear whether all children had really understood the learning process behind the robot's speaking, that is the relationship between their touching the robot, pressing the keys, and the robot's consequent speaking, especially as some children would press the wrong key, which would result in the robot speaking a different word than the expected one. These observations suggest that the game reached the limits of competence of so young children, as it required from them to concentrate for a long period of time on a repetitive task. However, we should mention that most of the children were keen on continuing playing with the doll after they had been told that the game was finished (each game lasted for about 20 minutes; this limit was chosen in order not to exceed the concentration capacity of young children). None of the children had had time to teach the entire vocabulary during the game, nor to understand fully all the different capabilities of the robot (especially to understand all the correlations between their acting on the robot and the robot's movements and speaking). The complexity of the robot seemed to intrigue them rather than to intimidate them.

It would now be interesting to carry out further test with older children and again with young children, but in using a simplified program (no keyboard, teaching only dance patterns). An interesting test would be to have the robot staying in the nursery for several hours or days and observe the progression of understanding and interest of the children over this particular game.

## 11.5 Discussion

Experiments reported in section 11.3 showed that the system was able to learn complicated action sequences and to distinguish between them by labelling them differently. Important was that the sequences could be made of the same set of actions, ordered differently, or of a subset of these actions, ordered similarly or differently than they are in the longer sequences. This combinatorial property of the system was exploited further to teach the robot to associate combinations of words, which formed English proto-sentences, to describe its actions and perceptions of touch on different parts of its body, e.g. *I move left arm, you touch right foot*. By the exclusion property of the threshold-based retrieval mechanism (and because we chose an encoding or the sensor measurements which encapsulated specific features represented by the taught concepts), the concepts of *left, right, I move, you touch, arm, foot* could be extracted from teaching complete sentences only. Moreover, the robot would also learn the sequential ordering of words' appearance and would reproduce the sentence with the same ordering as in the teaching.

This second experiment demonstrated that the learning architecture could allow learning of a synthetic 'proto-language' which shares some properties with natural language: 1) each word (key in our experiment) can carry a specific meaning (e.g. *arm, head, right, left, etc.*); 2) words can be combined and the combination can be given a different meaning while not losing the meaning of each word taken separately (e.g. *You touch right arm, I move head right*); 3) different combinations of the same words can be given different meanings, the meaning of each combination being determined by the order of appearance of each word in the combination (e.g. different combinations of the same basic actions make a different dance pattern<sup>3</sup>); 4) the conceptual meaning of each word can be learned implicitly by only presenting them as part of complete sentences, which can then be used to infer new word combinations; 5) precedence between words' appearance in the combination is learned and can be used to infer the correct order when constructing a new word combination.

---

<sup>3</sup> This was not shown with the combinations of words, as we could not imagine two plausible combinations using the same words, given the 11 words we had chosen; however, the principle behind learning the dance sequences and the combination of words being the same, the observation made for the dance experiments is also valid for the word combinations.

Note that only simple sentence examples were used so far, in which the words could easily be tied to the taught concepts. Moreover, the 'language' the robot was to learn was regular; that is, the robot's learning task was to recognise temporal regularities in the words' ordering across the taught sentences and to correlate the words' usage with its sensors and actuators' activity. As such, these experiments were a first step towards demonstrating the validity of the system (the learning architecture and the imitative strategy) for teaching a robot a symbolic communication system, such as a regular language. However, it remains to be shown how the system could scale up to learning a complete language with grammar structure and irregularities. So far, the architecture functioning is based on recognising only regularities. It is therefore unclear whether it could account also for exception rules. Moreover, more studies of the network capacity for learning sequences should be done to determine the model's ability for handling complex recursions and combinations as found in natural language. In page 239 we carry further this discussion.

Experiments done with children of 5 years old showed the potential of the system as a game for children. The children enjoyed playing with the robot because they could interact with it in different ways. The robot would respond to the children touching specific parts of its body, by making small movements or little noises. It would mimic the child's head and arm movements. It would speak words which are part of the child's every-day vocabulary (e.g. *food, hello, no*). Imitation is a game that young children like to play with each other and their parents, thus it was easy for them to understand that they could interact with the robot in this way.

Communication and imitation were the principal means of interaction between the child and the robot. They were easy to understand for the child as they are social interactions, which he has already mastered. More generally, communication and imitation would also be important means of interaction between humans and robots. In section 1.1.4 and 1.1.3, we mentioned how verbal communication and imitation could be used by the demonstrator to drive the robot's attention to the task she is demonstrating. The experiments we reported here showed how, while imitating the movements of a human demonstrator, a robot could be taught to perform and then label sequences of movements. In our system, learning, communication and imitation behaviours result

from the same processes, namely training and retrieval algorithms of the DRAMA architecture. This results in a computationally fast and cheap system, which allowed its implementation in a computationally limited hardware system. From a robotic point of view, these are important characteristics which makes the system particularly relevant for further implementation for real time control and learning in more complex robotic platforms (that is robots with finer sensor capabilities and more complex actuators, which require more complex computation and are thus very sensitive to the speed of computation). This will be discussed further in chapter 12.

## 11.6 Synthesis

This chapter reported on experiments, in which a mobile robot is taught by a human instructor to perform sequences of actions. Teaching is based on an imitation game, in which the robot mirrors the arm and head movements of the teacher. The instructor teaches the robot to distinguish between action sequences by assigning them a different label. The learning capability of the robot is provided by the DRAMA architecture. The experiments were aimed at evaluating the architecture ability to learn sequences with common subpatterns, as demonstrated in simulation in section 5.2. The ability of the model to learn sequences and combinations of inputs was tested further in an experiment in which the robot was taught English proto-sentences to describe its actions and perceptions. For these experiments, we built and used a robot whose shape and features look similar to that of a familiar doll. We tested the reaction of children to the robot by having them playing a simplified teaching game.

Results demonstrated the validity of the set-up, namely the learning architecture and the imitative strategy, to teaching a robot complex sequences and combinations of sensor and actuator inputs. In particular, tests with children suggest that the imitative and communicative behaviours of Robota makes it an interesting toy for children. Moreover, if Robota was provided with more complex actuator capabilities, it would also make an interesting robotic platform for research on human-robot interaction and especially on robot learning from demonstration and by imitation.

The first experiment was designed to test the robot's ability to learn a simple task. The robot was trained to move a block from one position to another. The training process involved showing the robot a sequence of actions that would move the block to the target position. The robot was then tested on a new sequence of actions to see if it could learn the task. The results showed that the robot was able to learn the task after a few trials.

The second experiment was designed to test the robot's ability to learn a more complex task. The robot was trained to move a block from one position to another, but the target position was now a function of the starting position. The training process involved showing the robot a sequence of actions that would move the block to the target position for a given starting position. The robot was then tested on a new sequence of actions to see if it could learn the task. The results showed that the robot was able to learn the task after a few trials.

The third experiment was designed to test the robot's ability to learn a task that required it to generalize from a limited set of training examples. The robot was trained to move a block from one position to another, but the target position was now a function of the starting position. The training process involved showing the robot a sequence of actions that would move the block to the target position for a given starting position. The robot was then tested on a new sequence of actions to see if it could learn the task. The results showed that the robot was able to learn the task after a few trials.

The fourth experiment was designed to test the robot's ability to learn a task that required it to generalize from a limited set of training examples. The robot was trained to move a block from one position to another, but the target position was now a function of the starting position. The training process involved showing the robot a sequence of actions that would move the block to the target position for a given starting position. The robot was then tested on a new sequence of actions to see if it could learn the task. The results showed that the robot was able to learn the task after a few trials.

The fifth experiment was designed to test the robot's ability to learn a task that required it to generalize from a limited set of training examples. The robot was trained to move a block from one position to another, but the target position was now a function of the starting position. The training process involved showing the robot a sequence of actions that would move the block to the target position for a given starting position. The robot was then tested on a new sequence of actions to see if it could learn the task. The results showed that the robot was able to learn the task after a few trials.

The sixth experiment was designed to test the robot's ability to learn a task that required it to generalize from a limited set of training examples. The robot was trained to move a block from one position to another, but the target position was now a function of the starting position. The training process involved showing the robot a sequence of actions that would move the block to the target position for a given starting position. The robot was then tested on a new sequence of actions to see if it could learn the task. The results showed that the robot was able to learn the task after a few trials.

The seventh experiment was designed to test the robot's ability to learn a task that required it to generalize from a limited set of training examples. The robot was trained to move a block from one position to another, but the target position was now a function of the starting position. The training process involved showing the robot a sequence of actions that would move the block to the target position for a given starting position. The robot was then tested on a new sequence of actions to see if it could learn the task. The results showed that the robot was able to learn the task after a few trials.

# Chapter 12

## Discussion

This chapter discusses the results of the work described in the previous chapters of this thesis in comparison with other work in the area described earlier in chapter 2. The first section discusses the theoretical performance of the DRAMA architecture, which was presented in chapters 4 and 5. The second section discusses the experimental performance of the DRAMA architecture, as demonstrated by the robotic experiments reported in chapters 6 to 11. The last section presents a number of research directions which follow from the work of this thesis.

### 12.1 DRAMA's theoretical performance

An important part of this thesis (chapters 4 and 5) was used to describe the DRAMA (Dynamical Recurrent Associative Memory Architecture) architecture, which we developed to allow learning of spatial and temporal regularities by an autonomous robot. The model consists of a fully recurrent neural network without hidden units, which uses Hebbian update rules. Similarly to time delay networks, it uses two weight parameters for each connection, to record separately the time delay and the frequency of co-occurrence of two input patterns. The DRAMA network differs from other structurally or functionally similar ANN models in two main aspects: 1) In contrast to other time-delay recurrent neural networks, it is based on an unsupervised learning algorithm, which uses Hebbian rules. 2) In contrast to other associative memory models, such as Hebbian networks, the connections of the network are associated with two parameters (instead of one) in order to keep a separate record of the spatial and tem-

poral structure of the input patterns. In particular, the temporal parameter allows it to record the real time of occurrence of the pattern.

In chapter 5, we analysed theoretically and through numerical simulations the properties of the model. The model was shown to cope (that is, the capacity remains maximal) with up to 30% of noise, where the noise corresponded to a 30% likelihood of spurious unit activation when presenting the training patterns and a 30% variation of time delay between consecutive activation of the input and output units of the pattern. The model was shown to be able to learn time series of inputs, while the series can overlap on one or more inputs (that is, it can learn several time series, where the series can have several similar inputs but with a different ordering of occurrence).

Training and retrieval algorithms are of one-time-step length, which makes the model computationally inexpensive and fast to run and, therefore, allows its implementation for on-line learning of a computationally limited robot. The advantages of the model in terms of fast and easy computation will be further discussed in the following sections. In the following, we discuss the general properties of the model by comparison to associative memory and recurrent neural networks, with which it shares several similar properties.

### 12.1.1 The model as an associative memory

The DRAMA architecture has several characteristics in common with associative memory models, such as Hebbian networks, as it uses a similar training algorithm (Hebbian rules) and a similar retrieval algorithm (threshold-based). Similarly to Hebbian networks which use a binary encoding for the patterns, the capacity of the DRAMA model decreases when trained with patterns whose encoding overlaps, that is, patterns which have common units active. In chapter 4, we described the Willshaw network which is a bidirectional associative memory based on Hebbian rules. The development of the DRAMA architecture was influenced by our study of this model. The DRAMA architecture has some similarities with the Willshaw network. The retrieval rule which is based on a threshold strategy where the thresholds act both on the connection weight strength and on the number of active input units is similar to that used in the Willshaw net (they are however different as the connection parameters have not the same

function or use in the two models). The structure (in DRAMA there are asymmetrical connection parameters and self-connections on the units) and the training rules of the two networks are however different. The DRAMA architecture differs from other Hebbian networks mainly in its recurrent structure (self-connections on the units and bidirectional asymmetric connections between the units), while other models use unidirectional or symmetric connections. The recurrent connections introduce a short-term memory of the units' activity, which allows association of temporally delayed unit activations, while the time delay between the two units' activation is unspecified, but remains within the margin of the short-term memory duration. By transitivity of the associations, time series of unit activations can be learned. Retrieval of the associations is such that each unit activates its correlated unit only when the correct time delay has passed. This property of DRAMA to introduce explicitly the time into one of the connection parameters (the time parameter) is what distinguishes it most significantly from other models of associative memory. Associative memory models that can learn sequences of patterns do exist ([Hattori & Hagiwara 96], [Kolen & Pollack 91], [Rinkus 95], [Schwenker *et al.* 96]). However, in these models, the time delay between each pattern occurrence is fixed and is equal to one processing cycle (it has therefore no intrinsic relationship with the real time of pattern occurrence). That is, the patterns of the series are presented sequentially to the net, without delay, and are retrieved similarly.

Note, finally, that the DRAMA architecture, when used in the robotic experiments, functions as an autoassociative memory on the robot's sensor-actuator state. However, in contrast to other autoassociative networks such as Hopfield nets ([Hopfield & Tank 86]) or Boltzmann Machines ([Hinton & Sejnowski 86]), patterns are not stored as minima of an energy function. Rather, because of the condition of orthogonality on the patterns, learning performs a sort of Principal Components Analysis (a well known capability of networks using Hebbian rules ([Fyfe 95], [Oja 82],[Oja *et al.* 95])), where the patterns form a basis of the input space.

### 12.1.2 The model as a recurrent neural network

The structure of the DRAMA network is a fully recurrent network, without hidden units. It is similar in function to time delay recurrent neural networks. However, it differs from these models by the fact that it uses an unsupervised training algorithm, based on Hebbian rules, while the other models use a supervised training algorithm, such as the backpropagation algorithm and other derivatives [Pearlmutter 95]. The advantage of using Hebbian rules is that training of the network requires only one-time-step for each processing cycle, which allows it to process the information (e.g. sensor information of the robot) in real time, while backpropagation needs several time steps of computation (usually of order 100 to 1000) between each information processing cycle. The drawback of using an unsupervised Hebbian learning algorithm is that it cannot be used to train a network with hidden units, that is, with intermediary units between input and output units, whose values are unknown. The algorithms developed to train RNNs with hidden units are the Boltzmann machine learning procedure [Hinton & Sejnowski 86], Backpropagation [Pineda 87] and other similar procedures, which all require several time steps of computation for each information processing cycle (which prevents on-line learning in computationally limited robotic systems like ours). It would, however, be interesting to investigate how hidden units could improve the network performance. "Hidden units makes it possible for the network to discover and exploit regularities of the task at hand such as symmetries or replicated structures" [Pearlmutter 95]. Therefore, using hidden units might improve the network's ability for discriminating between redundant or overlapping patterns and consequently increase the network's capacity, that is the number of patterns the network can store, by allowing more overlap between the patterns before the network fails to distinguish between them.

One may question why we did not use one of the previously developed models of recurrent neural networks (e.g. Elman net [Elman 90], Jordan net [Jordan 86] or dynamical recurrent net [Giles *et al.* 94]). There are several reasons for that. The first one is that, because of hardware limitations, we were restricted to defining a system that would use only integers (no floating points) and that should be computationally fast (because of the limited on-board processing power). Recurrent neural networks using

the backpropagation algorithm, as developed first by [Pineda 87], and other extensions [Pearlmutter 95] had to be eliminated because of their long time of computation (multiple training steps) and their complex computation (calculating derivatives). Associative Hebbian networks were very attractive because of their simplicity. One may argue in favour of buying a more powerful hardware system. Apart from the financial aspect, there is a motivation to try to do the best with what is at our disposal. We are not using expensive robots nor expensive sensors (no camera, laser, etc.), but we developed a system that is capable of more complex cognition than the simple behaviours of obstacle avoidance, wall following, etc.

### 12.1.3 Limitations of the model

There are a number of parameters which need to be fixed in the system, which are the short-term memory (decay parameter), the threshold  $T$  and the error margin  $e$ . The limitations of the model resulting from the arbitrary setting of those parameters were discussed in chapter 5, in which an algorithm was presented for on-line tuning of these parameters. The algorithm for tuning  $T$  and  $e$  has been used in the physical experiments (chapters 7 to 11). The decay parameter was however fixed in those experiments, because the algorithm was too slow to converge (as mentioned in chapter 11). For analysis of the effect of this parameter on the success of the robot's learning, see experiments of section 7.2.2.

An important aspect of DRAMA is its use of time parameters in addition to the confidence factors (which are similar to the usual synaptic weight of most Hebbian networks) in order to explicitly encapsulate the temporal structure of the patterns, on which it is trained. The first advantage of this encoding was that it made the model easy to analyse; the temporal structure of the learned pattern was easily retrievable as it was recorded in the parameter. In addition, it made the model capable of fast computation (see section 5.1.3); this was an important requirement for the robotic implementation of the model. A disadvantage of introducing a second parameter is that it increases the stack memory. Also, introducing explicitly the time is somewhat against the biological approach of connectionism, in which the temporal structure of a pattern is implicitly represented through the dynamics of the connectivity of

dynamic neurons (e.g. leaky-integrators). However, it is to be shown whether one of the biological and non-biological models is more efficient than the other. More neurons might be needed to represent the temporal structure in a biological type of model than in the DRAMA network. It is thus likely that a biological model would not have a better space efficiency. The processing time of the temporal information would also probably not be faster in a biological model than in DRAMA, as both would rely on the delayed firing rate of the neurons for the time processing.

We mentioned previously the DRAMA architecture's poor space efficiency, due to the constraints on the patterns' encoding. This was not a disadvantage in the particular robotic experiments we used it for. More important for us was the time efficiency and the capacity at learning complex time series. However, it might be a disadvantage for application using sensors with high sensitivity, such as a camera, which would require a large number of units to represent the sensor information (one unit per pixel or set of pixels). In this case, it might be relevant to pre-process the data using another ANN architecture, such as a topographical map or a feed forward NN, for a preliminary classification of the data<sup>1</sup>. Then, the reduced amount of information could be used by the DRAMA network for higher level classification. Such an approach has been followed by Tani [Tani 96], who uses a combination of a Hopfield associative memory network and a RNN. The robot's camera information is processed by the Hopfield net which determines categories of visual inputs. The recurrent neural network is trained on the output of the Hopfield net and on the simultaneous motor state of the robot. The net learns sequences of visual perception and action of the robot, while the robot travels in a circling corridor. Because J. Tani used backpropagation algorithm to train the RNN, he could not process the information on-line. It would be interesting to carry out the same work using the DRAMA architecture, instead of the RNN with backpropagation, and to determine whether it would allow to carry out successfully and on-line the same computation.

---

<sup>1</sup> Note that topographical maps are particularly relevant for sensors with high resolution as the classification often relies on finding topological invariance in the input.

## 12.2 DRAMA's experimental performance

In chapters 6 to 11, we reported on the implementation of the DRAMA architecture in a number of robotic experiments, carried out in physical and simulated environments. In these experiments, DRAMA was used to control both the robot's behaviour (retrieval of sensor-actuator associations in the network) and its learning (update of the associations). The experiments concerned teaching a robot a synthetic proto-language. The teaching was based on an unsupervised strategy, using an imitative scenario. Teaching was provided either by another robot (chapters 7, 8 and 10) or by a human instructor (chapters 7 and 11). The imitative scenario in the teacher-learner robot experiments, consisted of the learner robot following and, thus, replicating the movements of the teacher robot. In the human-robot experiment of chapter 11, the doll robot mirrors the demonstrator's arm and head movements.

In this section, we summarise the results of each of the robotic experiments of chapters 7 to 11 (chapter 6 gave an overview of the experiments of these chapters), and discuss these in comparison with related works in robotics on robot learning of spatio-temporal regularities and related works in computational linguistics, which study symbol grounding of an agent's perceptions.

### 12.2.1 Grounding of a robot's perceptions and actions

Chapter 7 presented experiments in which the robot was taught a vocabulary to label its actions and perception of inclination, orientation, light and touch. The learning task consisted of associating different radio signals with different combinations of inclination, compass, light and touch sensor measurements and different motor states. Learning resulted from a statistical process of elimination, provided by the DRAMA architecture, where incorrect signal-measurement associations were discarded, based on their frequency of occurrence.

#### Evaluation of the learning parameters

The difficulty of the task lay in choosing correctly the value of the learner agent's short-term memory, as a function of the dispersion and featural description of the objects

in the environment, so as to keep the number of incorrect associations low compared to the number of correct associations. The longer the memory, the more objects are remembered and associated with the same radio signals, thus the more difficult to extract the relevant associations. On the other hand, the memory should be long enough to allow the agent to make the relevant observations for the teaching. When teaching is given by the teacher robot, which is followed by the learner robot, the minimal memory duration corresponds to the time needed by the learner robot to reach the teacher's previous position at the time of the teaching episode (signal emission), from which it can make the same observations. In addition, the more features two objects have in common, i.e. the bigger the overlap between the teaching patterns, the less good the learning (since the DRAMA capacity decreases in the face of overlapping patterns). In section 7.2.2, we studied, through simulated and physical experiments, the success of the learning, while varying the short-term memory, the objects' dispersion and feature description. These studies validated the above mentioned expectations.

The retrieval threshold factors  $T$  and  $e$ , which appear in the unit activation function (given by equation 4.2, page 68) also determine the success of the learning, by fixing the tradeoff between considering two units' recorded co-activation as spurious or relevant. The threshold  $T$  determines this tradeoff as a function of the frequency of co-activation of these two units (given by the *confidence factor* parameter of the connection linking the two units). The more often the units have been co-activated, the more likely it is that this co-activation is not random. The threshold  $e$  discards correlation due to spurious unit activity by evaluating the average time delay of two units' co-activation and considering as irrelevant the associations which show too large a variation of time delay, relative to an error fixed by  $e$ , around the training time.

In section 5.1.4, we presented an algorithm to determine on-line the values of these thresholds, i.e. to calculate these values at each time step. The idea was to use the information on the ratio of spurious/relevant unit activations, reflected by the current values of the network connection parameters. The calculation was based on the assumption that incorrectly updated connections should have the lowest confidence factor values (since they are not frequently updated) and the biggest variations of time parameter (since there is no regularity in the time delay of co-activation of the

units). The thresholds' values were calculated based on a Gaussian estimation of the distribution of the parameter values of correct and incorrect connections. Through numerical simulations, we showed that the capacity of the network was significantly improved when calculating the threshold values at each processing cycle, than when using fixed value thresholds. Further, in section 5.2.1, we presented an algorithm for on-line tuning of the short-term memory value and tested it in simulation for learning a sequence of nine patterns. The algorithm was shown to successfully converge up to a proportion of 20% of noise, i.e. the likelihood of a spurious unit activation, in the input. However, as the time needed for convergence was relatively long, the algorithm was found to be too slow to be used reliably in physical robotic experiments. It would now be very interesting to carry on with this investigation and to design an algorithm for dynamical update of the DRAMA learning parameter to be used in robotics experiments.

Note finally, that other factors also influence the success of an experiment, e.g. the agent's sensor capabilities (range and sensitivity) and behaviour control (non homogeneous travelling). Further implementation of the model in different robotic set-ups, in particular in robots with more degrees of freedom and greater sensor sensitivity would allow us to determine the real influence of these hardware characteristics on the success of the learning. Note, however, that the fact that we implemented the architecture in three different robotic set-ups (FischerTechnik vehicles, Lego vehicles and a doll robot), using different sensors and applied in different environments, showed that the success of the learning is not dependent on a particular type of hardware. However, it might be improved by using finer sensor capabilities, which would give more information to distinguish between the objects of the teaching (lowering the overlap between teaching patterns and thus improving the network capacity), and better actuator capacities, which would make the robot follow the other robot (chapters 7 and 8) more smoothly and thus be less liable to make incorrect measurements.

A last remark concerns the fact that the learning method we proposed is bottom-up, starting from a fixed segmentation process of the information to an associative learning process. It would also be interesting to investigate a bottom-up-bottom mechanism, as proposed e.g. by [Grossberg & Merrill 92] and discussed by [Mozer 93], where feedback

from the associative memory can activate a tuning mechanism of the threshold parameters of the event recognition modules. Such a method was used by Tani [Tani 96] in an experiment on robot learning of sequences for action-perception. His model is composed of a Hopfield Associative memory, for classifying inputs from a CCD camera, and an RNN, for learning sequences of action-perception (camera features). The two networks are trained simultaneously, while each network's training algorithm is based on a minimisation of the prediction error of the two networks. We compare further our work with his in page 12.2.4.

### **Simulation versus physical experiments**

Experiments of sections 7.1.3, 7.2.2 and of chapter 10 were carried out both in simulated and physical environments. Simulation studies were carried out in order to determine the feasibility of the experiments by studying their successes and failures in a more reliable environment. The main advantage of simulations over physical experiments is that they are repeatable, faster (simulating a 1 hour experiment takes about 5 minutes) and do not suffer unexpected hardware breakdown. The disadvantages in terms of model faithfulness are, of course, well known (for a more complete discussion of this see [Torrance 92a]). Physical experiments were carried out because it was our original aim to have a real robot capable of learning a synthetic language and of learning sequences of perception-action.

In section 7.2.2 and chapter 10, simulation studies were carried out before physical experiments in order to demonstrate and test the stability and success of the learning in the particular set-ups. However, as mentioned in section 7.2.2, physical and simulated worlds differ in many aspects. For instance in the simulation a poor account is given of the physics of the sensors and the world features (a simple field of view is defined for the lightening whose intensity is invariant over time, the inclination of the hills is perfect at all points, etc.). Because of this, the results of simulations and physical experiments could only be compared qualitatively. In all experiments, physical and simulated experiments were consistent in showing successful learning in each experiment. In particular, physical and simulated studies of section 7.2.2 pointed out the same effect of the short-term memory, object dispersion and featural description

on the learning success.

Recently, there has been a number of works, in which people first developed the robot controller in simulation and then tested it into real robots, e.g. ([Lee *et al.* 96], [Martinoli & Mondada 95], [Nolfi 97a], [Miglino *et al.* 95], [Wilson *et al.* 97]). Particularly handy for this kind of experiments is the Khepera robot [Mondada *et al.* 93], which has been developed together with a simulator. Because of its small size, the Khepera requires the building of a specific environment for the experiments. This environment is often simpler than, e.g., an office environment, which makes it easier to simulate. The robot is often working alone [Miglino *et al.* 95, Lee *et al.* 96] or together with other Khepera robots [Martinoli & Mondada 95], which again reduces the uncertainty of the task at hand (in comparison to an inhabited office environment). The remaining differences between simulated and physical worlds in these experiments (similarly to ours) are related to the physics of the robot's perceptions (the robot's real sensors are bound to give imperfect measurements, and physical light and infra-red emissions are difficult to simulate) and the robot's actions (friction effects of the motors on the ground are often not simulated and the acceleration and speed of the motors in the physical world is often variable due to mechanical defects of the gears). It would then be interesting to test how the robot's controller developed in the simulation can cope with the physical noise. In this case, it is advantageous to develop an adaptive controller, provided with learning mechanisms to better cope with large variations in the environmental dynamics. Connectionist models have been shown by these other authors to be good models of robot controllers for this purpose. Our work validated these results, in the sense that we used a connectionist architecture (DRAMA), which was shown to successfully cope with the noise of the physical experiments.

### **Comparison with work on robot grounding of perceptions and actions**

Similar experiments on grounding radio signals in a robot's sensor capabilities were carried out previously by Yanco and Stein [Yanco & Stein 93] and Steels and Vogt [Steels & Vogt 97], who used respectively reinforcement learning and evolutionary techniques (see section 2.3.1 for a more detailed description of these experiments). Their experiments showed that a vocabulary of 5 and 3 words was learned after 900 and 60

training examples, respectively. Our method seems then faster at learning a vocabulary of large or the same size. In the experiment of section 7.1.3, we reported on an experiment in which the robot learned a vocabulary of 8 words to describe four actions and four perceptions of orientation. The robot learned the vocabulary in less than 30 teaching examples, which correspond to about 15 to 30 minutes of physical experiment.

In addition, our learning method is more general than the above mentioned methods, as it makes no restriction on the type of perception the robots could talk about. In Yanco & Stein's work [Yanco & Stein 93], the vocabulary consisted only of the robot's actions because the learning algorithm was based on an action-selection mechanisms. In Steels & Vogt [Steels & Vogt 97], the vocabulary concerned only the robots' external perceptions as these were the only perceptions they could share. By contrast, the mutual following strategy we use in our work allows the two agents to share a common context, namely sharing external (facing the same direction) and internal perceptions (performing the same movement, travelling the same distance and on the same ground). When aligning itself behind the teacher, the learner agent naturally points in the direction the teacher 'looks' at. While we avoid the problem of the mirror effect<sup>2</sup> occurring in pointing, we however introduce another problem lying in the fact that the learner cannot see over the teacher's body. In our experiments this is partly solved by the two agents moving and the learner eventually reaching the teacher's previous position. This obstruction problem also occurs in Steels & Vogt's pointing model [Steels & Vogt 97], as in their experiments the two robots spoke also about objects to their sides, which one of the agents could not see. In addition, because the learning mechanism we use is based simply on mutual associations between inputs from any sensor or actuator systems of the agent, the vocabulary can potentially concern any proprio and extero perceptions of the agent. In this thesis, we presented experiments where the robot was taught a vocabulary about its external perceptions of objects (section 7.2.2 and chapter 8) and internal perceptions of inclination, orientation, touch

---

<sup>2</sup> In Steels and Vogt's experiments [Steels & Vogt 97], the teacher robot points 'verbally' by indicating the quadrant in which to look at relative to itself; however, there is a left right symmetry between the two agents in their definition of quadrants for all objects lying between them; this effect was, however, first not considered by the authors and did explain why the agents could never agree on more than a third of the vocabulary (only a third of the same 'quadrant' space can be seen by both agents); on the suggestion of the present author, they did take this effect into account for later experiments and were able to improve the agents' learning performance to a maximum of 90% of success [Vogt 98].

and movements (sections 7.1.2 and 7.2.1 and chapters 10 and 11).

On the other hand, a disadvantage of our following strategy over a more precise pointing mechanism is that we are restricted to speaking only about static events or events that last long enough for the learner to eventually see them. In particular, a 'finger' pointing mechanism, which is more accurate than our 'body' pointing mechanism, would be more appropriate in an environment where features cannot be spatially separated (as in the eyes and face example which we described in section 3.4). Scasselatti et al. [Marjanovic *et al.* 96, Scasselatti 98] investigate a complex cognitive mechanism, implemented on a humanoid robot, which allows the robot to direct its gaze in the direction at which a human demonstrator is pointing.

### 12.2.2 Grounding of perceptions of a group of robots

In chapter 8, we showed, through simulation studies, that the learning strategy scaled up successfully to teaching a group of robots a vocabulary about objects' locations. In particular, we carried out studies in which we demonstrated the importance of the following strategy for good transmission of the vocabulary.

These experiments can be compared to computational linguistics studies, which investigated the development and evolution of a language among a group of artificial agents, e.g. [Arita & Koyama 98, Kirby & Hurford 97b, MacLennan 91, Di Paulo 98, Steels 96b] (see section 2.3.1 for an overview of research directions in this area). Similarly to these studies, the learning strategy of our experiment is based on unsupervised associative learning, where correct categorisation of sensor perceptions into concepts (words of the vocabulary) results from a process of statistical elimination among all possible meaning-object pairs, where the most likely pairs, i.e. the most frequently observed, are chosen. In agreement with these experiments, we showed that the population of agents would converge successfully to the same language. However, our study differs from those works, as we investigate the correct transmission of a single predefined vocabulary by one teacher agent to a group of learner agents, while these other works study the emergence of a shared vocabulary between agents which have none to start with. But, since the process underlying the success of their and our experiments is the same, namely statistical elimination of incorrect pairings, we obtain

similar results, which we however interpret differently. For instance, in our simulation, the learner agents were allowed to teach the vocabulary when confident enough in their word-meaning pairings. We varied the level of confidence and showed that, when this was not sufficiently high, the experiment would fail, that is, the agents would not converge to the same definition of the vocabulary. The experiment fails because the agents make on average more incorrect associations than correct ones. We stopped the simulations, when more than fifty percent of the agents were unsuccessful in their learning. But, if we had carried the study further, we would have been able to define whether all agents would converge to another definition of the vocabulary or whether they would continuously switch from one definition to another, as a result of too many misinterpretations of the vocabulary. We did not study this aspect as it was not our primary interest, and also because the simulations were computationally heavy and slow (as we simulate the physical interactions of the agents). However, this would be an interesting extension of the present work. In the above mentioned Computational Linguistics studies, people investigated especially this aspect of the simulation. They showed how a language can evolve, that is, change its definition, to produce 'dialects', when different populations of agents with different languages are put in contact [Steels 97b] or when erroneous transmission of the language across generations of agents occurs as a result of misinterpretation of the pairings [Arita & Koyama 98].

An important difference between our study and the Computational Linguistics' ones lies in the fact that we simulate the behaviours of the agents. That is, we give a physical description of the agents, in providing them with a body, sensors and actuators, and of their interactions, in defining behaviours for the agents and giving a temporal and spatial account of the agents' actions. As we mentioned in section 3.4, other computational studies of the development and evolution of communication gave only a poor account, or none at all, of the physical features of the agents and their environment. However, although these studies used disembodied a-behavioural agents, their results have sometimes been interpreted (with exaggeration in our opinion) in terms of the agents' behavioural skills. E.g., in their simulation study, Oliphant and Batali [Oliphant & Batali 97] show that imitation is not a factor enhancing the evolution of communication. This result seems to contradict ours, as, in our simulations, the ability of following and so imitating another robot's movement

is advantageous. The problem lies in our different definition of imitation or imitative behaviour. The agents' imitative capability in Oliphant and Batali's work is not a behavioural skill, which would involve at minimum a definition of the agent's actuators and the agent's observation of the second agent's actions and of the mapping mechanism which interprets the observed actions into the agent's own set of actions [Dautenhahn 95, Demiris & Hayes 96, Meltzoff 96]. The same critique applies to Schmajuk and Zanutto's [Schmajuk & Zanutto 97] neural model of conditional learning, in which imitation consists only of feeding the observer agent's network with the data given to the observed agent's network. Again, in this work, no description is given of the type and functioning of the mechanism, resulting in the data transfer. In Oliphant and Batali's work, imitation is for an agent to match its statistical function of word-meaning associations, a purely cognitive functionality, with the one of a second agent; in other words, one agent learns the language by strictly reproducing each word-pair association as produced by the other agent. The authors conclude that this 'imitative' learning procedure is not advantageous for the development of communication as it only allows for copying a currently existing communicative system and does not allow for its modification and further improvement. These authors' work and their results relative to the influence of imitative behaviour on the development of communication is thus not comparable to ours as we are not approaching the problem from the same angle. In our model, imitative behaviour is not the learning mechanism per se, but acts alongside associative learning to constrain association on relevant data.

### 12.2.3 Using the robot's communicative ability

In chapter 9, we presented three case study experiments, which showed how the robot's understanding of a vocabulary to describe its actions or perceptions can be used 1) by the robot to speed up its search process for another robot, which is speaking its location, 2) by a group of robots to improve their learning performance, 3) by a human to direct the robot's path across an office environment.

Research in collaborative robotics investigates the advantage of using explicit communication to improve a group of robots' performance (see section 2.2.1 for a description of some of these works and their results). Similarly to these works, we showed through

simulation studies (experiment of section 8.2) how explicit symbolic communication could be used by a group of robots to transmit information about objects' location in the environment, which resulted in speeding up the robots' learning of these locations, compared to doing so through a random search. In particular, long distance communication was advantageous in our experiment compared to short distance communication, as it allowed more agents to get access faster to the information (a protocol for the communication prevented clashes in the transmission across multiple agents).

The experiment of section 9.1 showed another case study of the usefulness of communication for the interaction of two autonomous robots. In this experiment, a 'child' robot finds its 'mother' robot, while it can not see it, by successfully interpreting the mother's signals which indicate its location and consequently directing itself towards the mother's location. The experiment was meant as an illustration of the use of symbolic communication to improve the performance of a robot, whose other sensors have poor sensitivity. This result correlates with those of Mataric [Mataric 97b] and Parker [Parker 94a]. Independently, they showed how the use of explicit communication to transmit information, to which the robots do not have access (due to limitation in the sensor competence), could enhance the performance of a group of robots engaged in collaborative tasks (see also section 2.2.1).

The importance of communication in human-robot interaction, for either commanding or teaching the robot has been evaluated by numerous works in robotics (see section 2.2.1 for an overview). The recent development of speech processors, which are algorithms and hardware capable of understanding and producing natural language, has enabled them to be used to communicate with a robot for different applications, A number of works report on teaching a robot names for locations in an office environment, where the communication protocol is either built-in [Asoh *et al.* 97, Tatsuno 96], or is learned by the robot as part of the landmark learning task [Henis & Levinson 95, Torrance 92b]. Similarly, in section 9.3, we showed an example of use of the robot's understanding of a vocabulary to command it. In this experiment, the robot's understanding of a 4-word vocabulary, which labels four compass directions, is used by a human instructor to direct the robot's path in an office environment, by having the robot turning towards the compass direction labelled by the instructor's signal.

Our specific interest behind running these experiments was to show how easy it was to use the information which had been stored previously in the DRAMA architecture. Each of the three experiments was run after a first experimental stage, during which the robot had been taught the vocabulary. Retrieval of the stored word-meaning associations, namely association of radio signals with combinations of sensor measurements and actuator states, required the implementation of simple mechanisms aside from the original retrieval mechanism of the DRAMA network (described in chapter 4). Moreover, retrieval was done in real time.

In the experiment of section 9.3, there was also a specific interest in having the robot first learning the mapping radio signal-compass measurement, as it was difficult to set this one by hand. The magnetic field sensed by the robot was different when the robot was running, due to magnetic effects of the motors and of the electrical connection to the Dodgem cage, than when it was unpowered (which corresponded to what we could measure with a potentiometer).

#### 12.2.4 Grounding of a robot's sequences of actions and perceptions

Chapters 10 and 11 reported on experiments in which the robot associated consecutively series of inputs, delayed in time, which led it to learning sequences of perceptions (chapters 10 and section 11.3.1) and actions (section 11.3.2). These experiments were meant to demonstrate the DRAMA architecture ability for learning spatio-temporal regularities in multi-modal inputs of a robot. In chapter 10, the robot learned the timing and ordering of a sequence of compass, radio and light measurements, resulting from its regular wandering in a series of three corridors (which accounted for the regularity of compass and light measurement) and from the regular radio signal transmissions of the teacher robot which it followed along the corridors. In chapter 11, we presented experiments with *Robota*, the doll robot, which associated series of actions with different sensor activations (activation of the keys of the keyboard); each key represented labels for each of the series of actions, or, in other word, each key represented a different starting state for retrieval of the sensor-actuator sequence. In section 11.3.2, *Robota* associated consecutive activation of keys of the keyboard with activation of other sensors (placed on parts of its body) and actuators. This was used to teach the

robot to label its perception of touch on its body and its actions using English proto-sentences, represented by the particular ordering of the keys in the sequence, where each key represented a different word.

The experiments of chapter 11 demonstrated the ability of the network for learning several time series with overlapping states. It also showed the ability of the network to deal with variable time delay of occurrence between two inputs; since the experiments were done by different demonstrators, it was seldom the case that two demonstrators will produce the taught sequence with the same frequency or even that the same demonstrator would repeat the sequence with the same time rate.

There has been a number of studies in robotics in which the robot learned spatio-temporal regularities in its sensor inputs. These works used, for instance, a two-layer topographical map to store separately the spatial and temporal regularities (in each layer) of the robot's visual information (CCD camera and infra-red sensors) [Gaussier *et al.* 98b, Harris & Recce 97, Nehmzow 98], or a recurrent neural network to predict a robot's perception-action sequences, when travelling in a corridor [Tani 96]. Our experiment, using the DRAMA architecture, has two advantages compared to these works: 1) learning and retrieval can be performed on-line, by contrast to [Nehmzow 98, Tani 96] works where it was done off-line, 2) learning concerns several sensor and actuator modalities, as opposed to all four above mentioned works, which consider only association from the robot's sensor perception to the robot's actions.

In our experiments, learning of several time series was used by the robot to reproduce the series, one series at a time, when prompted to do so, by presenting it the first state of the series, e.g. activating the key in the experiment of section 11.3.1. In the works of [Tani 96] and [Gaussier *et al.* 98a], retrieval of the learned sequences of inputs was used to determine the robot's path towards a given goal, by choosing the shortest one among the possible set of retrieved sequences. Given the property of the DRAMA architecture at learning time series with similar states, it is expected that the network could be used for learning different paths followed by the robot in its wandering, where the paths could have common branching points. Then, retrieval of the learned sequence could be used to determine the robot's path, similarly to Tani and Gaussier's experiments.

### 12.2.5 Imitative behaviour

As we mentioned in section 3.1 (p. 39), we used a broad definition of the term imitation throughout the thesis, that is, 'a behaviour skill, which leads to some sort of similarity in behaviour among two or more individuals'. The following behaviour, which we use in the experiments of chapters 7 to 10, is only an implicit means of producing imitative behaviour, as when the learner agent follows the teacher agent, it implicitly replicates the teacher's movements in the 2-D plane. In the experiments of chapter 11, the robot's imitative behaviour consisted of mirroring the arm and head movements of a human demonstrator. The mechanism behind the imitative process was very simple and was based on phototaxis of infra-red detectors. The robot and demonstrator were linked through a wiring of infra-red sensors, each of them carrying one part of an infra-red emitter-detector pair. Each movement of the demonstrator (and thus of one infra-red sensor) was detected by the robot's correlated infra-red receptor and indicated to the robot which movement to replicate.

Therefore, the mechanisms at the basis of our robots' imitative behaviour are by no means a model of similar mechanisms in animals. As mentioned in section 3.3, imitation requires complex cognitive capabilities in order to identify the movements to imitate and to transfer the observation of a movement to its own actuator system, so as to replicate the movement. Models of imitative behaviour, which address some of these issues, have been implemented recently in robotic experiments ([Demiris *et al.* 97], [Scassellati 98], [Kuniyoshi & Inoue 94] and [Cooke *et al.* 97]), in which a humanoid robot imitates the arm and head movements of a human demonstrator. The robot's imitative behaviour, in these works, results from complex algorithms which create a mapping between the robot's visual stimuli (recognition of the movements of the different limbs of the demonstrator) and the robot's motor commands. These works are impressive by the finesse of their result. In all these works, the robot's imitation of the movement is smooth, that is it replicates the observed movement continuously and in the three dimensions.

We kept the mechanisms allowing imitative behaviour simple on purpose, in order not to overload the computation of the robot (as mentioned earlier, keeping the computa-

tion fast and inexpensive was an important criterion to have the robot functioning in real time). Note that, the infra-red strategy used with the doll robot could possibly scale up to a humanoid robot, where all body joints of human and robot would be linked. This decomposition of human body movements through emissions of infra-red sensors is a method used frequently by scientists for analysing people's gait [Lakany 97]. It would then be interesting to use this infra-red set-up to have a human instructor teaching a humanoid robot complex sequences of movements, while the robot would mirror the human movements. Of course, in the case of actuators with several degrees of freedom, the movement replication mechanism should be more complex than the simple photo-taxis mechanism we used with the doll robot, so as to take into account the mechanical constraints resulting from the complex physical structure of the robot.

## 12.3 Overall synthesis and outlook

This thesis addressed, on the one hand, the rather general problem of robot learning in a dynamic environment, while developing the DRAMA architecture. On the other hand, it focussed on a more specific problem, that is grounding symbolic representation in an autonomous robot, for the implementation of DRAMA in robotic experiments. There are now several research directions, which follow from the work presented here. We sketch some of these in the following first subsection. Most of the ideas presented here have been mentioned earlier in this thesis. We here put these together, so as to give a general overview of the potential extensions of this work. Finally, we conclude this section with a broad discussion on the issue of following a cognitivist versus a behaviour-based approach for studying the development of a language. This relates to the approach and hypotheses, which underlie the work of this thesis. We here contrast our approach to that generally taken by works in this area.

### 12.3.1 Future work and perspectives

This thesis presented a novel connectionist model, DRAMA, which allows learning of spatio-temporal regularities and of time series. Important characteristics of the model were stressed, such as 1) fast computation for the model to react in real-time, which was shown by the model's implementation for on-line and on-board learning of

a computationally limited robot; 2) robustness and adaptability in the face of varying environmental constraints, which was demonstrated by the successful implementation of the model in robotic learning tasks, involving a proportion of 30% of noisy/incorrect data; 3) as little built-in knowledge as possible to keep the system unspecific to a particular type of implementation (task, agent or environment), which was verified by the implementation of the model in different types of robotic agents (simulated and real vehicles and a doll-shaped robot), used in different environments and for different learning tasks.

### **Further implementations of the model**

The DRAMA ability for dynamic learning of spatio-temporal regularities and of time series is very general. Moreover, its unspecificity to a particular experimental set-up makes it relevant for a wide range of other robotic (and perhaps non-robotics) tasks. As we discussed in section 1.1.1 and 3.4, a robot is embedded in dynamic interactions with its environment. Learning from these requires the ability for correlating sensor and actuator information spatially — in order to categorise its perceptions — and temporally — in order to record the history of its perceptions and actions, which can then be used for predicting its perception and planning its actions. In chapter 10, we presented an experiment in which the robot learned time series of perception when travelling in a series of corridors. This experiment was a first step towards more complex experiments, involving learning of several perception-action series. For instance, we suggested, in page 234, the re-implementation of the experiments done by [Gaussier *et al.* 98a, Tani 96] for learning the robot's different paths across a complex environment. The DRAMA ability to learn several time series with common patterns, demonstrated by the experiments of chapter 11, should allow the model to cope with the high redundancy in the perception-action sequences, involved in the above mentioned tasks. In addition, the DRAMA ability to make spatial correlations across all the sensors of the robot, was shown to allow learning of landmarks, that is recognition of objects and learning of their locations. This suggests yet another kind of robotic learning task in which to implement the model.

Finally, another interesting implementation of the model would be the learning of

complex motor skills, that is of long and complex action sequences. In particular, we think of experiments on robot learning by imitation, whereby the robot first observes and then replicates the movement performed by a demonstrator. The experiments we carried out with the doll robot in chapter 11, in which the robot learned several action sequences by replicating those shown by a human demonstrator, was a first step in this direction. In section 12.2.5, we discuss scaling up of this experiment to teaching a robot with more degrees of freedom, such as a humanoid robot.

In the previous discussion, we mentioned that there might be limitations in the complexity of the tasks the model could learn. For instance, it is not clear whether the lack of hidden units limits the model's computational abilities for spatial categorisation of the input information. Also, the fact that the network takes only integer inputs, presented in a discrete time scale, prevents it from learning continuous spatial and temporal correlations in the inputs. However, hardware constraints in most robotic experiments result in a discrete scaling of the sensor information. Finally, the necessity of keeping the overlap between the patterns' encoding minimal (with ideally orthogonal encoding for all the patterns) might limit the use of the model in experiments requiring a high space efficiency for storage of a large number of patterns. Further implementations of the model will help in defining the real limitations of the model, which could then lead to further development and improvement of the model's functioning. We address some of these in the following.

### **Further analysis and development of the model**

There are a number of theoretical points relative to the architecture's functioning, which would be interesting to investigate. For instance, because the network takes a finite number of states, its dynamics could be studied by comparison to a Finite State Machine or to Hidden Markov Models (as mentioned in section 5.2.2). The network's ability for learning time series, that is the number of time series it can learn before unsuccessful retrieval of one series, should be calculated. So far, we gave only the network's ability for making bi-directional associations (see pp. 85, 91). Finally, it would be interesting to further develop the algorithm for dynamic tuning of the retrieval thresholds and the short-term memory parameter, which we showed to have

a significant influence on the success of the learning.

**Learning of a language** An important part of this thesis described experiments in which a robot learned a symbolic representation of its perceptions, following the teaching given by another agent, a human instructor or another robot. We showed that the robot could be taught a lexicon, i.e. a set of labels, to describe several of its internal and external perceptions, such as its body inclination and orientation, its measure of energy consumption (as a measure of distance travelled between two points), its different actions, as well as its observation of features belonging to objects in the environment. Further, we exploited the DRAMA ability at learning time series for teaching the robot combinations of the terms of the lexicon, so as to form English proto-sentences.

The success of these experiments opens the way to further implementation of the model in experiments on teaching a robot a complete language, which includes a syntax and other grammatical rules. The fact that the DRAMA architecture is comparable in function to a Hidden Markov model or other recurrent neural networks, which are models currently used in techniques of Natural Language Processing, suggests that the model could scale up successfully to learning a regular language. Further experiments in this direction will allow us to determine the model's capability and limitations (e.g. the need for hidden units) for performing such a learning task. Note that learning of irregular languages, such as human languages, which requires the understanding of ambiguity and the integration of exceptions to grammatical rules, are still a long way from the very simple model we have proposed so far.

Moreover, the experiments demonstrated the importance and utility of the imitative strategy as a means for setting up a shared set of perceptions between the communicative agents. This suggests to continue developing more complex imitative scenarios, which would then allow transmitting more complex forms of communication. This could, for instance, be the development of the robot's capability for imitating facial expressions (as currently investigated by [Breazeal & Scassellati 98]), used then for transmitting symbolic expressions to label human states of emotions. The robot's ability at replicating complex action sequences (as suggested earlier in this section) and

at learning long perception-action sequences could be used to teach the robot more complex conceptual notions for describing its behaviour. Behaviour relates here to sequences of perception-action, embedded in a particular context (involving long-term goals and some history of motivations behind performing the actions). In this case, the robot's internal state would be more complex than simply the robot's sensor and actuator state (e.g., it would also include the value of different motivational factors) and, therefore, higher-level conceptual information (e.g., concerning goals, rewards, etc.) could be transmitted through the language.

**Last thought** We hope that 1) this work will lead to the development of autonomous, social robots which are learning and communicating in real-world scenarios, using a rich sensor-motor repertoire of action and interaction with the environment, and 2) that along these studies we can contribute to imitation and language research with a particular focus on the role of the concrete embodiment of a robot. We extend this last thought in the following and final subsection.

### 12.3.2 Cognitivist versus behaviour-based approach

We built our approach on the assumption that grounding of communication creates constraints not only on the cognitive capabilities of the agent but also and especially on its behavioural capacities. We distinguished between behavioural and cognitive capabilities, where the former do not result only from the outcome of the agent's cognitive processes, but from the interaction of these processes' outputs, i.e. the agent's actions, with the dynamics of the agent's environment (see sections 1.1.1). We defined key features for the learning capacities, such as a selective mechanism for the discretisation of sensor perception, a short-term memory of perceptual events and associative capacities, and provided these in our experiments by using the DRAMA architecture. In chapter 3, we pointed out the importance of a basic behavioural social relationship between the two communicative agents to act as an attentional mechanism for eliminating irrelevant information, which could not always be discarded by means of combinatorial analysis only. We then provided our agents with a basic ability of imitation, namely mutual following of teacher and learner robots in chapters 7 and 8, and

mirror imitation of arm and head movements of a human demonstrator in chapter 11. This movement imitation scenario creates a spatial and temporal binding between the teacher and learner agents, which allows them to share a similar set of perceptions. However, although we used a simple protocol, namely phototaxis, for achieving this basic ‘social’ interaction, this proved to be a powerful ‘external’ (behavioural instead of cognitive) attentional mechanism.

We followed a behaviour-oriented (rather than a pure cognitivist) approach to the problem of grounding communication, by investigating the influence of social and behavioural aspects on the development of communication. Our approach differs from previous studies of the influence of sociality on the development of communication (e.g. [Dunbar 93, Parisi & Floreano 92, Di Paulo 98, Worden 98]), as we give a complete spatial and temporal description of our agents’ behaviour. This allowed us to point out the influence on the success of the learning of environmental factors (section 7.2.2), such as the featural description and relative dispersion of the objects relevant for the teaching in the environment, and of two parameters of the learning architecture (short-term memory duration and long-term memory capacity). In addition, we showed the importance of having a spatial and temporal synchronisation between the two communicative agents to allow sharing of a common perceptual context. The role of context is a key issue in sociality and social understanding. In particular, studies of (first or second) language acquisition demonstrated the importance of a meaningful social and linguistic context for inferring the correct meaning of spoken words or gestural signals ([Harley 94]). Our robotic implementation of the symbol grounding problem confirmed these ideas and showed that a simple movement imitation strategy is an interesting scenario for the transmission of a language, as it is an easy means for getting the agents to share a common context of perceptions. Nevertheless, however similar the two agents’ perceptions could be, they would never be exactly the same and therefore the agent’s learning capacities should be sufficiently complex to compensate for these differences. Not only should it be able to associate temporally delayed patterns, but this under a great amount of noisy and spurious data, two performance criteria which have been shown to be satisfied by the DRAMA learning architecture.

Whether used for grooming [Dunbar 93] or for transmitting information, communic-

ation is an interactive process between the two communicative agents and as such is a social interaction. Communication does not exist without the physical means of its production and reception. Whatever the level of interpretation chosen for the communicative signals, it is about the physical perception the communicative agents have of their world. As Steven Harnad puts it [Harnad 98]: “even at such abstract cognitive heights, [referring to] the highest level of abstraction of natural language when our interactions with objects are based only on the interactions between names and descriptions, [...] embodiment is never escaped, for the power of names and propositions is completely parasitic on the meanings of those names, and those must all eventually be grounded in the sensorimotor interactions with the kinds of objects they designate, and the sensorimotor invariants on the basis of which the names are assigned”. This requires very complex cognitive processes of segmentation of sensor information, sequence processing and spatial and temporal mapping. A model of the evolution of communication in terms of the agents’ cognitive capabilities should then encapsulate a description of **all** the required cognitive functions, as if we consider the progression of brain evolution (e.g. from fish to apes and humans), the ‘brain’ of an individual animal has always evolved as a whole, as a complete organ; although different areas have differentiated and specialised, e.g. when comparing ‘primitive’ with ‘advanced’ vertebrates (compare reptiles and mammals), different parts of the body plan of a species and therefore different parts of the nervous system have not evolved independently from each other and from the rest of the body. However, the computational models of the evolution of communication produced so far have generally deliberately left aside most of the so called ‘low-level’ cognitive functions, such as sensory perception, episodic memory, and focus of attention.

Note, finally, that the evolution of the brain functionality cannot be separated from the agent’s behaviour, as these cognitive processes have been shaped by the constraints created by the agent’s interactions with its environment. The body of an animal is a functionally and physiologically well integrated system. Thus, perception, action, communication and cognition are intrinsically interrelated in an embodied system. Our perception of the world is linked to our way of interpreting the world and talking about events in and actions upon this world. Such a system, or behavioral account of cognitive processes, is closely related to the *enactivist* position [Maturana 88, Varela *et al.* 93].

Similar positions are discussed in the area of embodied artificial intelligence (EAI), see [Prem 98] for a discussion of implications of embodiment for cognitive theories (and section 1.1). As Erich Prem points out, cognitive science and linguistics research has strongly focused on formal aspects and neglected, e.g. the notion of time, situatedness and interaction dynamics in animal cognition. "Getting the interaction dynamics right" is, according to Prem, a key principle of EAI research. This statement and the position to which it relates is confirmed by the findings of our experiments on robot communication grounded in social interaction dynamics. This brings us, then, to conclude in suggesting that a behaviour-oriented approach, as e.g. an enactivist position, might be more appropriate than a pure cognitivist one for describing the cognitive processes involved in solving the symbol grounding problem.



## Chapter 13

# Conclusion

This thesis has been concerned with extending our understanding of the problem of transmitting a symbolic and conceptual representation of the world to robotic agents, while grounding these representations in the robot's own sensori-motor perceptions. The main contribution of this work has been to integrate three different research directions in AI, which so far have been independently investigated. These were the design of connectionist models for learning in autonomous robots, the study of the use of social skills, such as imitation and communication, for robot teaching by demonstration, and the development of agents capable of understanding a language. The aim of the thesis was to build autonomous robots capable of learning a synthetic language.

There were three stages of development of this work. We first determined a number of key features for the cognitive and behavioural skills of the robot, which would allow the robot to learn a language taught by another agent, a robot or a human. In line with studies of Linguistics, we required for the agent to possess associative capabilities for spatio-temporal association across multiple sensor-actuator modalities. Inspired psychological studies of the development of language in children, we required behavioural skills for the robot to coordinate its actions with that of a second agent. This behavioural co-ordination is achieved, in our experiments, through movement imitation. In a second stage, we developed a connectionist architecture, DRAMA (Dynamical Recurrent Associative Memory Architecture), which allows learning of spatial regularities and time series. By fixing parts of the DRAMA network connectivity, we implemented phototaxis behaviour, which resulted in the robot's imitating behaviour,

i.e. following or mirroring the movements of a human or of a robot demonstrator. In a third stage, we validated the architecture in several robotic experiments for teaching a robot a synthetic proto-language. The complexity of the experiments was gradually increased, such as to verify different aspects of the system. We used three different robotic systems, two types of wheeled-based robots and a doll-shaped robot, in different environments, and thus demonstrated the generality of the system relative to the experimental set-up. We evaluated the DRAMA architecture's ability for learning spatio-temporal regularities across the sensor-actuator space of the robot, by teaching the robot a vocabulary to label both its proprio- and exteroceptions. We then validated the imitative scenario, as a means for transmitting a language to heterogeneous agents, by having the robot taught by a human and by a morphologically different robot. Further, we showed that the system scales up successfully to grounding a vocabulary among a group of robots. We demonstrated the architecture's ability for learning time series, by having the robot learning sequences of actions, dance patterns, and of perceptions, while travelling in a maze. We then used this architecture's property for teaching the robot combinations of words, leading to its learning of English proto-sentences to describe its actions and perceptions. Finally, we tested the importance of communicative and imitative skills for a robot in interaction with humans, by having children playing with the doll robot. The game consisted of teaching the robot a simple vocabulary to describe its body parts. The interest which the children showed during the game suggests that such a robotic system might be an interesting platform for diverse entertainment applications.

## 13.1 Contributions

The contributions of this work to related works in AI are the following:

**Learning robots:** There has been a number of works using connectionist models to allow learning of an autonomous robot. However, the current state of the art does not provide a model which allows both complex learning capability, such as learning of time series, and on-line processing of the learning. We presented a novel connectionist architecture, DRAMA, which allows learning of spatio-

temporal regularities and time series, and which is sufficiently computationally fast and inexpensive to perform in real-time. We demonstrated this property, by implementing the architecture for on-line learning and control of computationally limited autonomous robots.

**Robot learning by demonstration:** Recent work in robot learning by demonstration investigated learning by imitation as a means for teaching a robot complex motor skills. Following this line of research, we showed that imitation can also be used to teach a robot a language to describe both its proprio- and exteroceptions. We implemented the robot's learning and imitative behaviours, using a single connectionist architecture, DRAMA. We validated the system through experiments, in which a robot was taught by a human demonstrator to perform and label sequences of actions.

**Robot grounding of a language:** There has been a few works, which investigated experimentally a robot learning of a vocabulary, where the vocabulary concerned a specific and limited set of the robot's perceptions. Our work made a major step forward from those previous studies. We developed a system capable of learning a vocabulary, but also of learning combinations of words of the vocabulary, which form English proto-sentences, to label combinations and sequences of the robot's perceptions. We carried out extensive studies, which demonstrated the validity of the system in several robotic set-ups, in two agents, human-robot and robot-robot, and in multi-agent systems, for teaching a range of different concepts. These experiments and their results suggest the application of our model to experiments on robot learning of a complete regular language, composed of a large lexicon and of syntactical rules for combining the terms of the lexicon.

**Situated embodied robots:** Recent trends in robotics emphasize the importance of notions of situatedness and embodiment for the design of robotic agents. This work participated in this line of research by addressing the *symbol grounding problem* from a behaviouristic by opposition to a pure cognitivist point of view, for which the notions of embodiment and situatedness are key issues for symbolic cognition. This approach of ours brings a novel contribution to research in Computational Linguistics and related areas, which have for most ignored this

behavioural aspect in their simulation of the language development.

We view this thesis as a first step towards the development of robots with high level cognition, which would allow complex interactions with a human. It is our hope that parts of this work, such as the DRAMA architecture and the imitative scenario of the experiments, will be reused by others in more complex experiments. There are now several directions of research, which would be interesting to investigate. Some of these have been described in section 12.3.1 of this thesis.

# Bibliography

- [AAS 83] AAS. Communication and behavior in whales. Washington DC, Selected Symposium, 1983. on-line references <http://www.dai.ed.ac.uk/daidb/people/homes/rbf/WHALES/warinentry.html>.
- [Aitchison 96] J. Aitchison. *The Seeds of Speech: Language Origin and Evolution*. Cambridge University Press, 1996.
- [Anderson 95] J.A. Anderson. Associative networks. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*. MIT Press, 1995.
- [Angeline *et al.* 94] P. Angeline, G. Saunders, and J. Pollack. An evolutionary algorithm that constructs recurrent networks. *IEEE Trans. on Neural Networks*, 5:54–65, 1994.
- [Arbib 95] M. A. Arbib. *The handbook of brain theory and neural networks*. A Bradford book. Cambridge, Mass. London : MIT Press, 1995.
- [Arita & Koyama 98] T. Arita and Y. Koyama. Evolution of linguistic diversity in a simple communication system. *Artificial Life*, 4:1:9–17, 1998.
- [Arnaud 97] P. Arnaud. Collective locomotion. In *Proceedings of the Second German Workshop on Artificial Life, GWAL'97, Dortmund*, pages 33–42, 1997.
- [Asada *et al.* 97] M. Asada, E. Uchibe, and K. Hosoda. Cooperative behavior acquisition for mobile robots in dynamically changing real worlds via vision-based reinforcement learning and development. In *Proceedings of First International Workshop on Cooperative Distributed Vision*, pp.127-142, 1997.
- [Asoh *et al.* 97] H. Asoh, S. Hayamizu, I. Hara, Y. Motomura, S. Akaho, and T. Matsui. Socially embedded learning of office-conversant robot jijo-2. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence, IJCAI'97*, 1997.

- [Bakker & Kuniyoshi 96] P. Bakker and Y. Kuniyoshi. Robot see, robot do : An overview of robot imitation. In *AISB Workshop on Learning in Robots and Animals, Brighton, UK*, April 1996.
- [Bandura 61] A. et al. Bandura. Transmission of aggression through imitation of aggressive models. *Journal of Abnormal and Social Psychology*, 63, pages 575–82, 1961.
- [Barnes & Gray 91] D. Barnes and J. Gray. Behaviour synthesis for co-operant mobile robot control. In *Int. Conf. on Control*, pages 1135–140, 1991.
- [Bay 95] J. S. Bay. Design of the army-ant cooperative lifting robot. *IEEE Robotics and Automation Magazine*, pages 36–43, March 1995.
- [Beckers et al. 94] R. Beckers, O. E. Holland, and J. L. Deneubourg. From local actions to global tasks: stigmergy and collective robotics. In R. A. Brooks and P. Maes, editors, *Artificial Life IV, Proc. of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, pages 181–189, 1994.
- [Beer & Gallagher 92] R.D. Beer and J.C. Gallagher. Evolving dynamical neural networks for adaptive behavior. *Adaptive Behavior*, 1(1):91–122, 1992.
- [Beer 95] R. D. Beer. On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3(4):469–509, 1995.
- [Bengio 95] Y. et al Bengio. Lerc: a hybrid nn/hmm for on-line handwriting recognition. *Neural Computation*, 7:5:1289–1303, 1995.
- [Bengio 96a] Y. Bengio. Markovian models for sequential data. Technical Report 1049, 1996.
- [Bengio 96b] Y. Bengio. *Neural networks for Speech and Sequence Processing*. International Thompson Computer Press, 1996.
- [Beny 88] G. Beny. The concept of cellular robotic system. In *IEEE international symposium on intelligent control*, pages 57–62, 1988.
- [Berthouze & Itakura 98] L. Berthouze and S. Itakura. On why should a robot recognise itself. In *3rd International Symposium on Artificial Life and Robotics, Beppu, Jan 19–21.*, pages 678–681, 1998.

- [Berthouze & Kuniyoshi 98] L. Berthouze and Y. Kuniyoshi. Emergence and categorisation of coordinated visual behavior through embodied interaction. *Autonomous Robots and Machine Learning*, 5 and 31:369–379 and 187–200, 1998.
- [Bickerton 90] D. Bickerton. *Language & Human Behaviour*. UCL Press, 1990.
- [Billard & Dautenhahn 97a] A. Billard and K. Dautenhahn. Grounding communication in situated, social robots. In *Proceedings of TIMR 97, Towards Intelligent Mobile Robots Conference, Manchester*. Tech. Rep. Series, Dept. of Computer Science, Manchester University, ISSN 1361 - 6161. Report number UMCS-97-9-1, 1997. Longer version is published as technical report of the dept. of Artificial Intelligence, University of Edinburgh: Tech No 45.
- [Billard & Dautenhahn 97b] A. Billard and K. Dautenhahn. Grounding communication in situated, social robots, 1997.
- [Billard & Dautenhahn 98a] A. Billard and K. Dautenhahn. Experiments in social robotics: grounding and use of communication in autonomous agents. *accepted for publication in Adaptive Behaviour Journal, special issue on simulation of social agents*, 1998.
- [Billard & Dautenhahn 98b] A. Billard and K. Dautenhahn. Grounding communication in autonomous robots: an experimental study. *Robotics and Autonomous Systems, Special Issue on Scientific methods in mobile robotics*, 24:1-2, 1998.
- [Billard & Hayes 97a] A. Billard and G. Hayes. Learning to communicate through imitation in autonomous robots. In Wulfram Gerstner et al., editor, *Proceedings of ICANN97, 7th International Conference on Artificial Neural Networks*, pages 763–768, 1997.
- [Billard & Hayes 97b] A. Billard and G. Hayes. Robot's first steps, robot's first words ... In Richard Shillcock A. Sorace, Caroline Heycock, editor, *Proceedings of the GALA '97 Conference on Language Acquisition : Edinburgh*. Human Communication Research Centre, University of Edinburgh, 1997.
- [Billard & Hayes 97c] A. Billard and G. Hayes. Transmitting communication skills through imitation in autonomous robots. In J. Demiris A. Birk, editor, *Proceedings of EWLR97, Sixth European Workshop on Learning Robots, Brighton*, July 1997.

- [Billard & Hayes 98] A. Billard and G. Hayes. Transmitting communication skills through imitation in autonomous robots. In A. Birk and J. Demiris, editors, *Learning Robots: A Multi-Perspective Exploration*. LNAI Series, Springer-Verlag, 1998.
- [Billard & Hayes 99] A. Billard and G. Hayes. Drama, a connectionist architecture for control and learning in autonomous robots. *Adaptive Behaviour Journal*, Vol. 7:1, 1999.
- [Billard 96] A. Billard. Allo kazam, do you follow me? learning to speak through imitation for social robots. Unpublished M.Sc. thesis, Technical Report 43, Dept. of AI, University of Edinburgh, October 1996.
- [Billard *et al.* 98] A. Billard, K. Dautenhahn, and G. Hayes. Experiments on human-robot communication with robota, an imitative learning and communicating doll robot. In K. Dautenhahn B. Edmonds, editor, *Proceedings of Socially Situated Intelligence Workshop, Zurich CH, as part of the Fifth Conference of the Society of Adaptive Behaviour*. Centre for Policy Modelling technical report series: No. CPM-98-38, 1998.
- [Blumberg *et al.* 96] B. Blumberg, P. Todd, and P. Maes. No bad dogs: Ethological lessons for learning. In *From Animals To Animats 4, Proceedings of the Third International Conference on the Simulation of Adaptive Behavior*. The MIT Press/Bradford Books, 1996.
- [Breazeal & Scassellati 98] C. Breazeal and B. Scassellati. Infant-like social interactions between a robot and a human caretaker. *Submitted to a special issue of Adaptive Behavior on Simulation Models of Social Agents*, 1998.
- [Bremner & Gavin 88] J. Bremner and Gavin. *Infancy*. Oxford : Basil Blackwell, 1988.
- [Bremner & Gavin 97] J. Bremner and Gavin. *Infant development : recent advances*. Hove : Psychology Press, 1997.
- [Bridle 90] J. Bridle. Alphanets: a recurrent neural network architecture with a hidden markov model interpretation. *Speech communication*, vol. 9, no.1, pages 83-92, 1990.
- [Brooks & Stein 94] R.A. Brooks and L.A. Stein. Building brains for bodies. *Autonomous Robots*, 1:1:7-25, November 1994.
- [Brooks 89] R. A. Brooks. How to build complete creatures rather than isolated cognitive simulators. *Architectures for Intelligence*, pages 225-239, Fall 1989.

- [Brooks 91a] R. A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.
- [Brooks 91b] R.A. Brooks. Intelligence without reason. In *IJCAI-91, International Joint Conference on Artificial Intelligence, Sydney, Australia*, pages 659–595, 1991.
- [Brooks 97] R.A. Brooks. From earwigs to humans. *Robotics and Autonomous Systems*, 20:2–4:291–304, June 1997.
- [Brooks et al. 98a] R. Brooks, C. Breazeal, M. Marjanovic, B. Scassellati, and M. Williamson. The cog project: Building a humanoid robot. In *LLNAI series*. Springer-Verlag, 1998. to appear.
- [Brooks et al. 98b] R.A. Brooks, C. Breazeal (Ferrell), R. Irie, C. C. Kemp, M. Marjanovic, B. Scassellati, and M. Williamson. Alternative essences of intelligence: lessons from embodied ai. In *AAAI*, 1998.
- [Bruner & Watson 83] J. S. Bruner and R. Watson. *Child's talk : learning to use language*. Oxford : Oxford University Press, 1983.
- [Buckingham & Willshaw 92] J T Buckingham and D J Willshaw. Performance characteristics of the associative net. *Network*, 3:407–414, 1992.
- [Byrne & Whiten 88] R. W. Byrne and A. Whiten. *Machiavellian intelligence : social expertise and the evolution of intellect in monkeys, apes, and humans*. Oxford science publications. Oxford : Clarendon Press, 1988.
- [Canamero & Van deVelde 97] D. Canamero and W. Van de Velde. Socially emotional: Using emotions to ground social interaction. In K. Dautehhahn, editor, *Socially Intelligent Agents, Papers from the 1997 AAAI Fall Symposium*, pages 10–15. Technical Report FS-97-02. Menlo Park, CA: The AAAI Press, 1997.
- [Chauvin & Rumelhart 95] Y. Chauvin and D. E. Rumelhart. *Back-Propagation: Theory, Architecture and Application*. Lawrence Erlbaum Associates, 1995.
- [Chomsky 68] N. Chomsky. *Language and Mind*. Linguistics and philosophy. Harcourt Brace Jovanovich, Inc., 1968.
- [Colquhoun 96] A. Colquhoun. Protocol for communication of mini module radio transceiver. Memo of the DAI technical workshop at the University of Edinburgh, 1996.
- [Cooke et al. 97] S. R. Cooke, B. Kitts, R. Sekuler, and M. J. Mataric. Delayed and real-time imitation of complex visual gestures. In *Proceedings of the International Conference*

*on Vision, Recognition, Action: Neural Models of Mind and Machine*, Boston University, May 28-31 1997.

- [Crangle & Suppes 94] C. Crangle and P. Suppes. *Language and learning for robots*. CSLI Publication, 1994.
- [Dautenhahn 95] K. Dautenhahn. Getting to know each other – artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems*, 16:333–356, 1995.
- [Dautenhahn 96] K. Dautenhahn. Embodiment in animals and artifacts. In *Working Notes AAI 96 Symposium on Embodied Action and Cognition*. AAI Press, Technical Report FS-96-02, 1996.
- [Dautenhahn 97a] K. Dautenhahn. I could be you - the phenomenological dimension of social understanding. *Cybernetics and Systems, Special Issue on Epistemological Aspects of Embodied AI*, 28(5):417–453, 1997.
- [Dautenhahn 97b] K. Dautenhahn. Investigations into internal and external aspects of dynamic agent-environment couplings. In *Proc. of Dynamics, Synergetics, Autonomous Agents Conference, March 2-5, Gstaad, Switzerland*, 1997.
- [Dautenhahn 98] K. Dautenhahn. Grounding agent sociality: The social world is its own best model. In P. Petta J. Muller, editor, *From Agent Theory to Agent Implementation, a symposium at the 14th European Meeting on Cybernetics and Systems Research*, 1998.
- [Davis 73] J. M. Davis. Imitation: a review and critique. In P. Klopfer, editor, *Perspectives in Ethology*, volume 1, pages 43–72. Plenum, New York, 1973.
- [Day & Davenport 93] S. P. Day and M. R. Davenport. Continuous-time temporal back-propagation with adaptive time delay. *IEEE transaction on neural networks*, 4:348–354, 1993.
- [Dayan & Willshaw 91] P. Dayan and D J Willshaw. Optimising synaptic learning rule in linear associative memories. *Biological Cybernetics*, 65:253–265, 1991.
- [Deacon 92] T.W. Deacon. Brain-language coevolution. In J.A. Hawkins and M. Gell-Mann, editors, *The Evolution of Human Language*, SFI Studies in the Sciences of Complexity, pages 49–83. Addison-Wesley, 1992.
- [Deacon 97] T. Deacon. *Symbolic Species: the Co-Evolution of Language and the Human Brain*. General Science, 1997.

- [deBoer 97] B. de Boer. Generating vowel systems in a population of agents. In P. Husbands and I. Harvey, editors, *Proceedings of the Fourth European Conference on Artificial Life, ECAL97*. MIT Press, 1997.
- [Demiris & Hayes 96] J. Demiris and G. Hayes. Imitative learning mechanisms in robots and humans. In *published in the Proceedings of the 5th European Workshop on Learning Robots, Bari, Italy*, pages 9–16. also as Research Paper No 814 at the dept. of Artificial Intelligence at the University of Edinburgh, 1996.
- [Demiris & Mataric 98] J. Demiris and M.J. Mataric. Perceptuo-motor primitives in imitation. In *Working Notes, Autonomous Agents'98 Workshop on Agents In Interaction - Acquiring Competence Through Imitation, Minneapolis/St. Paul*, 1998.
- [Demiris 98] J. Demiris. *Movement imitation mechanisms in robots and humans*. Unpublished PhD thesis, Dept. of Artificial Intelligence, University of Edinburgh, December 1998.
- [Demiris et al. 97] J. Demiris, S. Rougeaux, G. M. Hayes, L. Berthouze, and Y. Kuniyoshi. Deferred imitation of human head movements by an active stereo vision head. In *Proceedings of the 6th IEEE International Workshop on Robot Human Communication*, pages 88–93. IEEE Press, Sendai, Japan, Sept. 1997.
- [Di Paulo 98] E. Di Paulo. An investigation into the evolution of communication. *Adaptive Behavior*, 6:2, 1998.
- [Dunbar 93] R.I.M. Dunbar. Coevolution of neocortical size, group size and language in humans. *Behavioural and Brain Sciences*, 16, pages 681–735, 1993.
- [Edmonds 97] B. Edmonds. Modelling socially intelligent agents. In *Proceedings of 'Modelling Socially Intelligent Agents in Organisations' workshop at the AAAI Fall Symposium on Socially Intelligent Agents, Cambridge, MA, November*. also as CPM Report No.: 97-26, Centre for Policy Modelling, Manchester Metropolitan University, 1997.
- [Elman 90] J.L. Elman. Finding structure in time. *Cognitive science*, 14, pages 179–211, 1990.
- [etal. 95] P. Tino et al. Finite state machines and recurrent neural networks - automata and dynamical systems approaches. *Neural Networks, special volume on Tem-*

*poral Dynamics and Time-Varying Pattern Recognition*, 1995.

- [Floreano *et al.* 98] D. Floreano, S. Nolfi, and F. Mondada. Competitive co-evolutionary robotics: From theory to practice. In *From Animals to Animats 5, Proceedings of the fourth International Conference on the Simulation of Adaptive Behavior*, pages 515–525. The MIT Press/Bradford Books, 1998.
- [Fujita & Kitano 98] M. Fujita and H. Kitano. Development of an autonomous quadruped robot for robot entertainment. *Autonomous Robots*, 5, pages 7–18, 1998.
- [Fukuda & Kawachi 93] T. Fukuda and Y. Kawachi. Cellular robotics. *Springer-Verlag*, pages 745–782, 1993.
- [Fyfe & Baddeley 95] C. Fyfe and R. Baddeley. Nonlinear data structure extraction using simple hebbian networks. *Biological Cybernetics*, 726, pages 533–541, 1995.
- [Fyfe 95] C. Fyfe. Information theory and hebbian learning. Tech. Report of Paisley University, Scotland, U.K., 1995.
- [Galef 88] B. G. Galef. Imitation in animals: history, definition and interpretation of data from the psychological laboratory. In T.R. Zentall and Jr. Cds. B.G. Galef, editors, *Social Learning: Psychological and Biological Perspectives*. Lawrence Erlbaum Associates, 1988.
- [Garton 92] A. F. Garton. Social interaction and the development of language and cognition. In *Essays in developmental psychology*. Laurence Erlbaum Associates, USA, 1992.
- [Gaussier & Zrehen 95] P. Gaussier and S Zrehen. Perac - a neural control architecture to control artificial animals. *Robotics and Autonomous Systems*, Vol.16, No.2-4, pages 291–320, 1995.
- [Gaussier *et al.* 98a] P. Gaussier, S. Lepretre, C Joulain, A Revel, J P Banquet, and M Quoy. Animal and robot learning: Experiments and models about visual navigation. In *Proceedings of the 7th European Workshop on Learning Robots, Edinburgh*. Tech. papers of the dept. of Artificial Intelligence at the University of Edinburgh, 1998.
- [Gaussier *et al.* 98b] P. Gaussier, S. Moga, J.P. Banquet, and M. Quoy. From perception-action loop to imitation processes: a bottom-up approach of learning by imitation. *Applied Artificial Intelligence*, 7:1, 1998.

- [Gaussier *et al.* 98c] P. Gaussier, A. Revel, C. Joulain, and S. Zrehen. Living in a partially structured environment: How to bypass the limitations of classical reinforcement techniques. *Robotics and Autonomous System*, 1998. to appear.
- [Giles 92] C.L. et al. Giles. Extracting and learning an unknown grammar with recurrent neural networks. *Advances in Neural Information Processing Systems*, 4, 1992.
- [Giles *et al.* 94] C.G. Giles, G.M. Kuhn, and R.J. Williams. Dynamic recurrent neural networks: Theory and applications. *IEEE Transactions on Neural Networks, Special Issue*, 52, 1994.
- [Girolami & Fyfe 97] M. Girolami and C. Fyfe. A temporal model of linear anti-hebbian learning. *Neural Processing Letters*, 4:3:1-10, 1997.
- [Goody 95] E. N. Goody. *Social intelligence and interaction : expressions and implications of the social bias in human intelligence*. Cambridge University Press, 1995.
- [Goody 97] E. N. Goody. Social intelligence and language: Another rubicon. In R. W. Byrne and A. Whiten, editors, *Machiavellian intelligence II : extensions and evaluations*, pages 365-397. Oxford : Clarendon Press, 1997.
- [Graham & Willshaw 95] P. B. Graham and D. J. Willshaw. Improving recall in a sparse autoassociative memory. *Biological Cybernetics*, 72, pages 337-346, 1995.
- [Graham & Willshaw 96] B. Graham and D. Willshaw. Information efficiency of the associative net at arbitrary coding rates. In *Proceedings of ICANN96, International Conference on Artificial Neural Networks*, LLNAI, pages 35-40. Springer-Verlag, 1996.
- [Graham & Willshaw 97] B. P. Graham and D. J. Willshaw. Capacity and information efficiency of the associative net. *Network: Comput. Neural Syst.*, 8:35-54, 1997.
- [Gross 96] R. Gross. *Psychology: The science of Mind and Behaviour*. Hodder & Stoughton Publishers, Bath Press, Somerset (U.K.), 1996.
- [Grossberg & Merrill 92] S. Grossberg and J. W. L. Merrill. A neural network model of adaptively reinforcement learning and hippocampal dynamics. *Cognitive Brain Research* 1, pages 3-38, 1992.

- [Hallam & Malcolm 94] JC Hallam and CA Malcolm. Behaviour: Perception, action and intelligence - the view from situated robotics. In *Proceedings Royal Society London*, volume 349, pages 29–42, 1994.
- [Hallam *et al.* 92] J. Hallam, T. Smithers, and U. Nehmzow. Real useful robots. *IEEE Robotics and Automation*, 1992.
- [Harley 94] T.A. Harley. *The Psychology of Language: from data to theory*. Erlbaum UK, Taylor & Francis Publishers, 1994.
- [Harnad 90] S. Harnad. The symbol grounding problem. *Physica D* 42, pages 335–346, 1990.
- [Harnad 93] S. Harnad. Grounding symbols in the analog world with neural nets. *Think, Special Issue on Connectionism versus Symbolism*, 2(1):12–78, 1993.
- [Harnad 98] S. Harnad. Beyond object constancy. *Seminar on Self-Learning Robots II: Bio-Robotic, Digest 98/248: 2/1-2/3*, 1998. Commentary on Pfeifer's (1998) 'Embodied Cognition: Dynamic and Information Theoretic Implications of Embodiment'.
- [Harris & Recce 97] K. D. Harris and M. Recce. Absolute localisation for a mobile robot using place cells. *Robotics And Autonomous Systems*, 22:3-4:393–406, 1997.
- [Hattori & Hagiwara 96] M. Hattori and M. Hagiwara. Episodic associative memories. *Neurocomputing, Vol.12, No.1*, pages 1–18, 1996.
- [Hayes & Demiris 94] G.M. Hayes and J. Demiris. A robot controller using learning by imitation. In *Proceedings of the 2nd International Symposium on Intelligent Robotic Systems, Grenoble, France*, pages 198–204, July 1994.
- [Hayes 84] P. J. Hayes. What the frame problem is and isn't. In Z. W. Plyshyn, editor, *The robot's dilemma: the Frame Problem In Artificial Intelligence*. Ablex, 1984.
- [Haykin 94] S. Haykin. *Neural Networks: a comprehensive foundation*. Prentice Hall International Editions, 1994.
- [Hebb 88] D.O. Hebb. The organisation of behaviour. *Neurocomputing: foundation of research, MIT Press, Cambridge,,* 1988.
- [Hemelrijk 95] C. K. Hemelrijk. A critical evaluation of the social intelligence hypothesis. Technical report No. IFI-AI-95.11, Dept. of Computer Science University of Zurich, 1995.

- [Henis & Levinson 95] E. Henis and S. Levinson. Language as part of sensorimotor behavior. In *Proceedings of 'Embodied Language and Action', AAAI Fall Symposium Tech. Report FS-95-05*, pages 48–53. AAAI Press, Menlo Park, CA, USA, 1995.
- [Hertz *et al.* 91] J. Hertz, A. Krogh, and R. Palmer. *An Introduction to the Theory of Neural Computation*. Addison-Wesley, 1991.
- [Heyes 96] C. M. Heyes. *Social learning in animals : the roots of culture*. San Diego, Academic Press, 1996.
- [Hinton & Anderson 89] G. E. Hinton and J. A. Anderson. *Parallel Models of Associative Memory*. Erlbaum, Hillsdale, 1989.
- [Hinton & Sejnowski 86] G.E. Hinton and T.J. Sejnowski. Learning and relearning in boltzmann machines. *Parallel Distributed Processing*, 1:7:282–317, 1986.
- [Holland 75] J. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.
- [Hopfield & Tank 86] J.J Hopfield and D.W. Tank. Computing with neural circuits: A model. *Science* 233, pages 625–633, 1986.
- [Humphrey 88] N. K. Humphrey. The social function of intellect. In R. W. Byrne and A. Whiten, editors, *Machiavellian intelligence*, pages 13–27. Oxford : Clarendon Press, 1988. Reprint of original version.
- [Hyvarinen & Oja 98] A. Hyvarinen and E. Oja. Independent component analysis by general nonlinear hebbian-like learning rules. *Signal Processing*, 1998.
- [Ijspeert *et al.* 98] A.J. Ijspeert, J. Hallam, and D. Willshaw. From lampreys to salamanders: evolving neural controllers for swimming and walking. In R. Pfeifer, B. Blumberg, J.-A. Meyer, and S.W. Wilson, editors, *From Animals to Animats, Proceedings of the Fifth International Conference of The Society for Adaptive Behavior (SAB98)*, pages 390–399. MIT Press, 1998.
- [Jackendoff 93] R. Jackendoff. *Patterns in the mind*. Harvester Wheatsheaf, 1993.
- [Johnson & Bay 95] P. J. Johnson and J. S. Bay. Distributed control of simulated autonomous mobile robot collectives in payload transportation. *Autonomous Robots*, 2:43–63, 1995.
- [Jordan 86] M.I. Jordan. Attractor dynamics and parallelism in a connectionist sequential machine. In Hillsdale Amherst. Lawrence Erlbaum, editor, *Proceedings of the*

*eight annual conference of the cognitive science society*, pages 531–546, 1986.

- [Kaski *et al.* 98] S. Kaski, J. Kangas, and T. Kohonen. Bibliography of self-organising map (som): Papers: 1981–1997. *Neural Computing Surveys*, 1:102–350, 1998.
- [Kawamura 63] S. Kawamura. The process of sub-culture propagation among japanese macaques. *Primates Social Behaviour*, pages 82–90, 1963.
- [Kirby & Hurford 97a] S. Kirby and J. Hurford. The evolution of incremental learning. In Richard Shillcock A. Sorace, Caroline Heycock, editor, *Proceedings of the GALA '97 Conference on Language Acquisition : Edinburgh 4-6 April 1997*. Human Communication Research Centre, University of Edinburgh, 1997.
- [Kirby & Hurford 97b] S. Kirby and J. Hurford. Learning, culture and evolution in the origin of linguistic constraints. In P. Husbands and I. Harvey, editors, *Proceedings of the Fourth European Conference on Artificial Life, ECAL97*. MIT Press, 1997.
- [Kirby 98] S. Kirby. Syntax without natural selection: How compositionality emerges from vocabulary in a population of learners. In Knight *et al.*, editor, *Proceedings of the Second International Conference on the Evolution of Language, London, April 1998*.
- [Kitano 98] H. Kitano, editor. *RoboCup-97: robot soccer World Cup I*. Lecture notes in computer science; Lecture notes in artificial intelligence v.1395. Berlin London : Springer, 1998.
- [Klingspor *et al.* 97] V. Klingspor, J. Demiris, and M. Kaiser. Human-robot-communication and machine learning. *Applied Artificial Intelligence Journal*, 11, pages 719–746, 1997.
- [Kodjabachian & Meyer 98] J. Kodjabachian and J.-A. Meyer. Evolution and development of neural networks controlling locomotion, gradient-following, and obstacle-avoidance in artificial insects. *to appear in IEEE Trans. on Neural Networks*, 1998.
- [Kohonen 89] T. Kohonen. *Self-organisation and associative memory*. Springer-Verlag, Berlin, 3rd edition, 1989.
- [Kolen & Pollack 91] J. F. Kolen and J. B. Pollack. Multiassociative memory. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, pages 7–10, August 1991.

- [Kolen 94] J. F. Kolen. Fools' gold: Extracting finite state machines from recurrent network dynamics. *Advances in Neural Information Processing Systems 6, San Francisco, CA: Morgan Kaufman, 1994.*
- [Koza 92] J. Koza. *Genetic Programming*. MIT Press, 1992.
- [Kuniyoshi & Inoue 94] M.I. Kuniyoshi and I. Inoue. Learning by watching: Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation, vol.10, no.6*, pages 799–822, 1994.
- [Kuniyoshi 94] M.I. Kuniyoshi. The science of imitation -towards physically and socially grounded intelligence. In *RWC Joint Symposium, 1994.*
- [Kuniyoshi 95] M.I. Kuniyoshi. Behavior matching by observation for multi-robot cooperation. In *International Symposium of Robotics Research, ISRR'95, 1995.*
- [Lakany 97] GM Lakany, H; Hayes. An algorithm for recognising walkers. In *Proceedings of the 1st International Conference on Audio- and Video-based Person Authentication, Crans-Montana, Switzerland, March 1997.*
- [Lee et al. 96] Wei-Po Lee, J. Hallam, and H. H. Lund. A hybrid gp/ga approach for co-evolving controllers and robot bodies to achieve fitness-specified tasks. In *Proceedings of IEEE 3rd International Conference on Evolutionary Computation*. IEEE Press, 1996.
- [Lin et al. 92] D. T. Lin, J. E. Dayhoff, and P.A. Ligomenides. Adaptive time delay neural network for temporal correlation and prediction. In *SPIE intelligent robots and computer vision XI: Biologica, neural net, and 3-D methods, Boston MA*, volume 1826, pages 170–181, 1992.
- [Lin et al. 93] D. T. Lin, P.A. Ligomenides, and J. E. Dayhoff. Learning spatio-temporal topology using an adaptive time-delay neural network. In *Proceedings of World congress on neural networks, Portland, OR*, volume 1, pages 291–294, 1993.
- [Lock 78] E.D. Lock. *Action, Gesture and Symbols: the emergence of language*. Academic Press, Harcourt Brace Jovanovich, 1978.
- [Lueg & Pfeifer 97] C. Lueg and R. Pfeifer. Cognition, situatedness, and situated design. In *Proceedings of the Second International Conference on Cognitive Technology (CT 97) Aizu, Japan, August 25 - 28 1997.*

- [MacLennan 91] B. MacLennan. Synthetic ethology: an approach to the study of communication. In J. D. Farmer C. G. Langton, C. Taylor and S. Rasmussen, editors, *Proceedings of the Second Artificial Life Workshop*, pages 631–658, 1991.
- [Maes 95] P. Maes. Artificial life meets entertainment: Interacting with lifelike autonomous agents. *Special Issue on New Horizons of Commercial and Industrial AI*, 38, No. 11:108–114, November 1995.
- [Marjanovic et al. 96] M. Marjanovic, B. Scassellati, and M. Williamson. Self-taught visually-guided pointing for a humanoid robot. In *From Animals to Animats 4, Proceedings of the fourth International Conference on the Simulation of Adaptive Behavior*. The MIT Press/Bradford Books, 1996.
- [Martinoli & Mondada 95] A. Martinoli and F. Mondada. Collective and cooperative group behaviours: Biologically inspired experiments in robotics. In O. Khatib and J.K. Salisbury, editors, *Proceeding of the Fourth Symposium on Experimental Robotics ISER-95, Stanford, US, June 30- July 2*, Lecture Notes in Control and Information Sciences, pages 3–10. Springer Verlag, 1995.
- [Martinoli et al. 97] A. Martinoli, M. Yamamoto, and F. Mondada. On the modelling of bio-inspired collective experiments with real robots. In P. Husbands and I. Harvey, editors, *Proceedings of the Fourth European Conference on Artificial Life, ECAL-97, Brighton, U.K.*, pages 28–31. MIT Press, 1997.
- [Mataric 94a] Maja J Mataric. *Interaction and Intelligent Behavior*. Unpublished PhD thesis, Massachusetts Institute Technology, AI Laboratory, Technical Report AITR-1495, May 1994.
- [Mataric 94b] M.J. Mataric. Learning Motor Skills by Imitation. In *AAAI Spring Symposium Toward Physical Interaction and Manipulation*, March 21-23 1994.
- [Mataric 95] M J Mataric. Issues and approaches in the design of collective autonomous agents. *Robotics and Autonomous Systems*, 16(2-4):321–331, 1995.
- [Mataric 97a] M J Mataric. Learning social behavior. *Robotics and Autonomous Systems*, 20:191–204, 1997.
- [Mataric 97b] M. J. Mataric. Reducing locality through communication in distributed multi-agent learning. *Experimental*

- and Theoretical Artificial Intelligence, special issue on Learning in Distributed Artificial Intelligence Systems, 1997.*
- [Mataric 97c] M.J. Mataric. Reinforcement learning in the multi-robot domain. *Autonomous Robots, Vol. 4, No. 1*, 1997.
- [Mataric 97d] M.J. Mataric. Studying the Role of Embodiment in Cognition. *Cybernetics and Systems Vol. 28, No. 6, special issue on Epistemological Aspects of Embodied AI, pp. 457-470*, July 1997.
- [Maturana & Varela 80] H. Maturana and F. Varela. *Autopoesis and Cognition: The Realization of the Living*. Boston: D Reidel Publishing, 1980.
- [Maturana 88] H. Maturana. Ontology of observing: The biological foundations of self consciousness and the physical domain of existence. In *Conference Workbook: Texts in Cybernetics*. American Society For Cybernetics Conference, Felton, CA., 18-23 October 1988.
- [Maturana et al. 95] H. Maturana, J. Mpodozis, and J. C. Letelier. Brain, language and the origin of human mental function. *Biological Research*, 28:15-26, 1995.
- [McCulloch & Pitts 43] W S. McCulloch and W. Pitts. A logical calculus of ideas immanent in nervous activity. *Bulletin of mathematical biophysics*, 5, 1943. reprinted in J.A. Anderson and E. Rosenfeld, eds, 'Neurocomputing: Foundations of research', MIT Press, Cambridge, 1988.
- [McFarland & Steels 97] D. McFarland and L. Steels. *Cooperative Robots: A Case Study in Animal Robotics*. Cambridge, Ma: The MIT Press, 1997.
- [McFarland 93] D. McFarland. *Animal Behaviour*. Longman, 1993.
- [Melhuish et al. 98] C Melhuish, O. Holland, and Hoddell S. Collective sorting and segregation in robots with minimal sensing. In R. Pfeifer, B. Blumberg, J.-A. Meyer, and S.W. Wilson, editors, *From Animals to Animats 5, Proceedings of the Fifth International Conference of The Society for Adaptive Behavior (SAB98)*, pages 465-470. MIT Press, 1998.
- [Mellish et al. 94] CS Mellish, P Whitelock, and GD Ritchie. Techniques in natural language processing 1. Teaching Paper No 994, dept. of Artificial Intelligence at the University of Edinburgh, 1994.

- [Meltzoff 90] A Meltzoff. The human infant as imitative generalist: a 20-year progress report on infant imitation with implications for comparative psychology. In Heyes C.M. and Galef B.G., editors, *Social Learning in Animals: The Roots of Culture*. New York: Academic Press, 1990.
- [Meltzoff 96] A. Meltzoff. Towards a developmental cognitive science. the implication of cross-modal matching and imitation for the development of representation and memory in infancy. In Adele Diamond, editor, *The development and Neural Basis of Higher cognitive Functions*. Annals of the N. Y. Acad. of Sci., 608, 1996.
- [Miglino *et al.* 95] O. Miglino, H.H. Lund, and S. Nolfi. Evolving mobile robots in simulated and real environments. *Artificial Life*, 2:4:417-434, 1995.
- [Mondada *et al.* 93] F. Mondada, E. Franzi, and P. Ienne. Mobile robot miniaturisation: a tool for investigation in control algorithms. In *Proceedings of ISER'93, Kyoto, Japan, October 1993*. Paper on-line at <http://diwww.epfl.ch/lami/robots/K-family/>.
- [Morgan & Bourlard 90] N. Morgan and H. Bourlard. Continuous speech recognition using multilayer perceptrons with hidden markov models. In *International Conference on Acoustics, Speech and Signal Processing*, pages 413-416, 1990. Albuquerque, NM.
- [Mozer 93] M. C. Mozer. Neural net architecture for temporal sequence processing. In N. Gershenfelds A. Weigend, editor, *Predicting the future and understanding the past*. CA: Addison-Wesley Publishing, 1993.
- [Mugny & Carugati 89] G. Mugny and F. Carugati. *Social representations of intelligence*. European monographs in social psychology. Cambridge University Press, 1989.
- [Nehmzow 98] U. Nehmzow. An episodic mapping algorithm for mobile robot self-localisation. In Nehaniv C., editor, *Computation for Metaphor, Analogy, and Agents*. Springer-Verlag series, 1998.
- [Newell & Simon 76] A. Newell and H.A. Simon. Computer science as empirical enquiry: symbols and search. In *Communication of the association for computing machinery*, volume 19, pages 113-126, 1976. also in, John Haugeland (Ed.), *Mind Design*, Cambridge Mass: The MIT Press (A Bradford Book).

- [Nolfi 97a] S. Nolfi. Evolving non-trivial behaviors on real robots. *Robotics And Autonomous Systems*, 22:3-4:187-198, 1997.
- [Nolfi 97b] S. Nolfi. Using emergent modularity to develop control system for mobile robots. *Adaptive Behavior*, 5:3-4:343-364, 1997.
- [Nottebohm 76] F. Nottebohm. Vocal tract and brain: a search for evolutionary bottlenecks. *Annals of N.Y. Academy of Science*, 280:643-649, 1976.
- [Oja 82] E. Oja. A simplified neuron model as a principal component analyser. *Journal of mathematical biology*, 15, pages 267-273, 1982.
- [Oja et al. 95] E. Oja, J. Karhunen, L. Wang, and R. Vigario. Principal and independent components in neural networks - recent developments. In *Proceedings of VII Italian Workshop on Neural Nets WIRN'95, May 18 - 20, Vietri sul Mare, Italy*, 1995.
- [Oliphant & Batali 97] M. Oliphant and J. Batali. Learning and the emergence of coordinated communication. *Centre for Research on Language Newsletter*, 111, 1997.
- [Oxford 95] Oxford. *The concise Oxford dictionary*. Clarendon Press - Oxford, ninth edition, 1995.
- [Palm et al. 97] G. Palm, F. Schwenker, F. T. Sommer, and A. Strey. Neural associative memory. In A. Krikelis and C. C. Weems, editors, *In Associative Processing and Processors*, pages 307-326. IEEE CS Press, Los Alamitos, CA, USA, 1997.
- [Parisi & Floreano 92] D. Parisi and D. Floreano. Prediction and imitation of linguistic sounds by neural networks. In A. Paoloni, editor, *Proceedings of the 1st Workshop on Neural Networks and Speech Processing. Roma: Fondazione Bordoni*, pages 50-61, 1992.
- [Parker 94a] L. E. Parker. An experiment in mobile robotic cooperation. In *Proceedings of the ASCE Speciality Conference on Robotics for Challenging Environments, February*, pages 131-139, 1994.
- [Parker 94b] L. E. Parker. *Heterogeneous Multi-Robot Cooperation*. Unpublished PhD thesis, Massachusetts Institute of Technology, January 1994.
- [Parker 98] L. E. Parker. Alliance: An architecture for fault tolerant multi-robot cooperation. *IEEE Transactions on Robotics and Automation*, 14:2, 1998.

- [Pearlmutter 95] B. A. Pearlmutter. Gradient calculations for dynamic recurrent neural networks: a survey. *IEEE transactions on neural networks*, 6:1212–1228, 1995.
- [Perkins & Hayes 97] S. Perkins and G. Hayes. Incremental acquisition of complex behaviour using structured evolution. In N.C. Steele G.D. Smith and R.F. Albrecht, editors, *Proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms '97, Norwich, UK*. Springer-Verlag Wein, New York, 1997.
- [Pfeifer & Scheier 94] R. Pfeifer and C. Scheier. From perception to action: the right direction? In P. Gaussier and J.-D. Nicoud, editors, *Proc. From Perception to Action Conference, Lausanne, Switzerland*, pages 1–11. IEEE Computer Society Press, 1994.
- [Pfeifer & Scheier 98] R. Pfeifer and C. Scheier. Sensory-motor coordination: the metaphor and beyond. *Robotics and Autonomous Systems, special issue on practice and future of autonomous agents*, 20:2-4, 1998.
- [Pfeifer 94] R. Pfeifer. The fungus eater approach to emotion: a view from artificial intelligence. *Cognitive Studies*, 1:42–57, 1994.
- [Pfeifer 95] R. Pfeifer. Cognition – perspectives from autonomous agents. *Robotics and Autonomous Systems*, 15:25–46, 1995.
- [Pfeifer 96] R. Pfeifer. Symbols, patterns, and behavior: towards a new understanding of intelligence. In *Proc. Japanese Conference on Artificial Intelligence, Tokyo*, pages 1–15, 1996.
- [Pfeifer 98] R. Pfeifer. Embodied system life. In *Proceedings of the International Symposium on System Life, Tokyo*, July 1998.
- [Pineda 87] F.J. Pineda. Generalisation of backpropagation to recurrent neural networks. *Physic Review Letter*, 18, pages 2229–2232, 1987.
- [Pinker 94] S. Pinker. *The language instinct*. Penguin books, 1994.
- [Pollack 87] J.B Pollack. *On Connectionist Models of Natural Language Processing*. Unpublished PhD thesis, Dept. of Computer Science, University of Illinois, 1987.
- [Prem 98] E. Prem. The implications of embodiment for cognitive theories. a systems view. In *Proceedings of the 14th European Meeting on Cybernetics and Systems Research, EMCSR'98*, pages 687–692, 1998.

- [Pulvermuller 98] F. Pulvermuller. Words in the brain's language. *Behavioural and Brain Science*, 1998.
- [Rabiner 89] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:2:257–285, February 1989.
- [Rinkus 95] G. Rinkus. *A Combinatorial Neural Network Exhibiting Episodic and Semantic Memory Properties for Spatio-Temporal Patterns*. Unpublished PhD thesis, Dept. of Cognitive and Neural Systems, Boston University, Boston, MA 02215, 1995.
- [Ritchie 94] GD Ritchie. Techniques in natural language processing 2. Teaching Paper No TE9, dept. of Artificial Intelligence at the University of Edinburgh, 1994.
- [Rosenfield 93] I. Rosenfield. *The strange familiar and forgotten. An anatomy of consciousness*. Vintage book, 1993.
- [Rumelhart et al. 86] D. Rumelhart, G. Hinton, and R. Williams. Learning internal representations by error propagation. *Parallel Distributed Processing*, 1, ch.8:318–362, 1986.
- [Salomon 98] R. Salomon. Achieving robust behavior by using proprioceptive activity patterns. *BioSystems*, in press, 1998.
- [Savage-Rumbaugh 93] E. S. Savage-Rumbaugh. *Language comprehension in ape and child*. Chicago : University of Chicago Press, 1993.
- [Scassellati 98] B. Scassellati. Imitation and mechanisms of joint attention: A developmental structure for building social skills on a humanoid robot. In Nehaniv C., editor, *Computation for Metaphor, Analogy, and Agents*. Springer-Verlag series, 1998.
- [Schaal 97] S. Schaal. Learning from demonstration. *Advances in Neural Information Processing Systems*, 9:1040–1046, 1997.
- [Schaal 99] S. Schaal. Nonparametric regression for learning non-linear transformations. In H. Ritter and O. Holland, editors, *Prerational Intelligence in Strategies, High-Level Processes and Collective Behavior*. Kluwer Academic Press, 1999.
- [Schmajuk & Zanutto 97] N.A. Schmajuk and B.S. Zanutto. Escape, avoidance, and imitation: A neural network approach. *Adaptive Behaviour*, 6:1:63–129, 1997.

- [Schwenker *et al.* 96] F. Schwenker, F. T. Sommer, and G. Palm. Iterative retrieval of sparsely coded associative memory patterns. *Neural Networks* 9, pages 445–455, 1996.
- [Searle 80] J. R. Searle. Minds, brains and programs. *Behavioral and Brain Sciences*, 3:417–424, 1980.
- [Shoham & Tennenholtz 96] Y. Shoham and M. Tennenholtz. On social laws for artificial agent societies: Off-line design. In P.E. Agre and S. J. Rosenschein, editors, *Computational Theories of Interaction and Agency*. MIT Press, 1996.
- [Sommer & Palm 98] F. T. Sommer and G. Palm. Bidirectional retrieval from associative memory. *Advances in Neural Information Processing Systems*, 10:675–681, 1998.
- [Steels & Vogt 97] L. Steels and P. Vogt. Grounding adaptive language games in robotic agents. In P. Husbands and I. Harvey, editors, *Proceedings of the Fourth European Conference on Artificial Life, ECAL97*, pages 473–484. MIT Press, 1997.
- [Steels 92] L. Steels. The pdl reference manual. Memo published at the Artificial Intelligence Laboratory, Vrije Universiteit Brussel, March 1992.
- [Steels 94a] L. Steels. A case study in the behaviour-oriented design of autonomous agents. In *Proceedings of the simulation of adaptive behaviour conference, Brighton, U.K.* MIT Press: Cambridge, 1994.
- [Steels 94b] L. Steels. Emergent functionality through on-line evolution. In *Proceedings of the 4th artificial life conference*. MIT Press: Cambridge, 1994.
- [Steels 96a] L. Steels. A self-organising spatial vocabulary. *Artificial Life Journal*, 2(3), 1996.
- [Steels 96b] L. Steels. The spontaneous self-organisation of an adaptive language. *Machine Intelligence*, 15, 1996.
- [Steels 97a] L. Steels. Constructing and sharing perceptual distinctions. In G. Widmer van Someren M., editor, *Proceedings of the European Conference on Machine Learning*. Springer-Verlag, Berlin, 1997.
- [Steels 97b] L. Steels. Language learning and language contact. In Prague. Daelemans, W., editor, *Proceedings of the workshop on Empirical Approaches to Language Acquisition*, 1997.
- [Steels 98] L. Steels. The origins of grammatical function. *Applied artificial intelligence*, 1998. to appear.

- [Sutton & Barto 98] R. S. Sutton and A. G. Barto. *Reinforcement learning : an introduction*. Adaptive computation and machine learning. Cambridge, Mass. : MIT Press, 1998.
- [Tani & Fukumura 95] J. Tani and N. Fukumura. Embedding a grammatical description in deterministic chaos: an experiment in recurrent neural learning. *Biological Cybernetics*, vol.72, pages 365–370, 1995.
- [Tani 96] J. Tani. Model-based learning for mobile robot navigation from the dynamical systems perspective. *IEEE Trans. on System, Man and Cybernetics, Part B, Special Issue on Robot Learning, Vol.26, No.3*, pages 421–436, 1996.
- [Tatsuno 96] J. et al. Tatsuno. Human friendly teaching for industrial robots. In *Proceedings of 5th IEEE International Workshop on Robot and Human Communication ROMAN'96, Tsukuba*, pages 549–550. IEEE, New York, NY, USA, 1996.
- [Torrance 92a] M. C. Torrance. The case for a realistic mobile robot simulator. In *Working Notes of the AAAI Fall Symposium on Applications of Artificial Intelligence to Real-World Autonomous Mobile Robots, Cambridge, MA, October, 1992*.
- [Torrance 92b] M.C. Torrance. Natural communication with robots. Unpublished M.Sc. thesis, MIT Department of Electrical Engineering and Computer Science, January 1992.
- [Uny Cao et al. 95] Y. Uny Cao, A. S. Fukunaga, A. B. Kahng, and F. Meng. Cooperative mobile robotics: antecedents and directions. In *Proceedings of IEEE/RSJ international conference on intelligent robots and systems, Pittsburgh, USA, 1995*.
- [Varela et al. 93] F. Varela, E. Thompson, and E. Rosch. The embodied mind. *American Journal of Psychology*, 106:121–6, 1993.
- [Vaughan et al. 98] R. Vaughan, N. Sumpter, A. Frost, and S. Cameron. Experiments in automatic flock control. In R. B. Fisher and G. M. Hayes, editors, *Proceedings of Symposium on Intelligent Robotics Systems, SIRS'98, Edinburgh, U.K.*, pages 47–54. University of Edinburgh Printing Services, 1998.
- [Verschure & Pfeifer 92] P. Verschure and R. Pfeifer. Categorization, representations, and the dynamics of system-environment interaction: a case study in autonomous systems. In

- & S. Wilson J.A. Meyer, H. Roitblat, editor, *From Animals to Animats: Proceedings of the Second International Conference on Simulation of Adaptive behavior*, pages 210–217. Cambridge, Ma.: MIT, 1992.
- [Vogt 97] P. Vogt. A perceptual grounded self-organising lexicon in robotic agents. Unpublished M.Sc. thesis, Dept. of Cognitive Science and Engineering at University of Groningen, 1997.
- [Vogt 98] P. Vogt. The evolution of a lexicon and meaning in robotic agents through self-organization. In J. R. Hurford C. Knight, editor, *The Evolution of Language, proceedings of the 2nd International Conference on the Evolution of Language, London*, 1998.
- [Walker et al. 93] A. Walker, J Hallam, and D. Willshaw. Bee-havior in a mobile robot: The construction of a self-organised cognitive map and its use in robot navigation within a complex natural environment. In *IEEE Int conf. on Robotics and Automation*, pages 1451–1455, 1993.
- [Webb 95] B. Webb. Using robots to model animals: a cricket test. *Robotics and Autonomous Systems*, 16:2-4:117–134, 1995.
- [Werbos 74] P.J. Werbos. *Beyond Regression: New tools for prediction and analysis in the behavioural science*. Unpublished PhD thesis, Harvard University, 1974.
- [Werger 98] Barry Brian Werger. Ullanta performance robotics. on-line publications at <http://www-robotics.usc.edu/barry/ullanta/>, USC CS dept., 1998.
- [Whiten & Ham 92] A. Whiten and R. Ham. On the nature and evolution of imitation in the animal kingdom: Reappraisal of a century of research. *Advances in the Study of Behaviour*, 21:239–83, 1992.
- [Williams & Zipser 89] R.J. Williams and D. Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1:270–280, 1989.
- [Willshaw et al. 69] D. Willshaw, O. Buneman, and H. Longuet-Higgins. Non-holographic associative memory. *Nature*, 222:960–962, 1969.
- [Wilson 71] E.O. Wilson. *The insect society*. Harvard University Press, 1971.

- [Wilson 80] E.O. Wilson. *Sociobiology: the new synthesis*. Social Sciences, 2nd edition, 1980.
- [Wilson et al. 97] M.S. Wilson, Clive M. King, and John E. Hunt. Evolving hierarchical robot behaviours. *Robotics And Autonomous Systems*, 22:3-4:215–230, 1997.
- [Worden 98] R. Worden. The evolution of language from social intelligence. In Studdert-Kennedy & Knight Hurford, editor, *The Evolution of Phonology and Syntax*. Cambridge University Press, 1998.
- [Wyatt 97] J. Wyatt. *Exploration and Inference in Learning from Reinforcement*. Unpublished PhD thesis, Dept. of Artificial Intelligence, University of Edinburgh, March 1997.
- [Wyatt 98] J. Wyatt. Bibliography on reinforcement learning: on-line at: <ftp://ftp.cs.bham.ac.uk/pub/authors/J.L.Wyatt/az.bib>, 1998.
- [Wyatt et al. 98] J. Wyatt, J Hoar, and G. Hayes. Design, analysis and comparison of robot learners. *Robotics and Autonomous Systems, Special Issue on Scientific methods in mobile robotics*, 1-2: No. 24, 1998.
- [Yamauchi & Beer 94] B. M. Yamauchi and R. D. Beer. Sequential behavior and learning in evolved dynamical neural networks. *Adaptive Behavior*, 2(3):219–246, 1994.
- [Yanco & Stein 93] H. Yanco and L. A. Stein. An adaptive communication protocol for cooperating mobile robots. In H.L. Roitblat J.-A. Meyer and S.W. Wilson, editors, *From Animals to Animats 2: Proceedings of the Second International Conference on the Simulation of Adaptive Behavior*, pages 478–485. The MIT Press, Bradford Books, 1993.
- [Yanco 94] H. A. Yanco. Robot communication: issues and implementations. Unpublished M.Sc. thesis, Massachusetts Institute of Technology, May 1994.
- [Zrehen 95] S. Zrehen. *Elements of Brain Design for Autonomous Agents*. Unpublished PhD thesis, Swiss Federal Institute of Technology, Lausanne, Computer Science Department, June 1995.
- [Zrehen et al. 98] S. Zrehen, H. Kitano, and M. Fujita. Learning in psychologically plausible conditions: The case of an imaginary pet robot. In *From animals to animats 5, Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*, pages 48–58. The MIT Press, 1998.

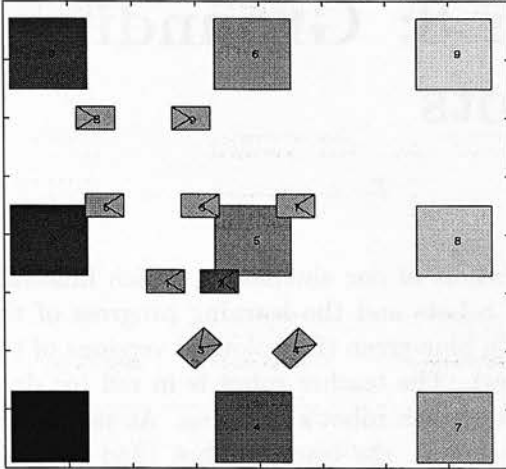


## Appendix A

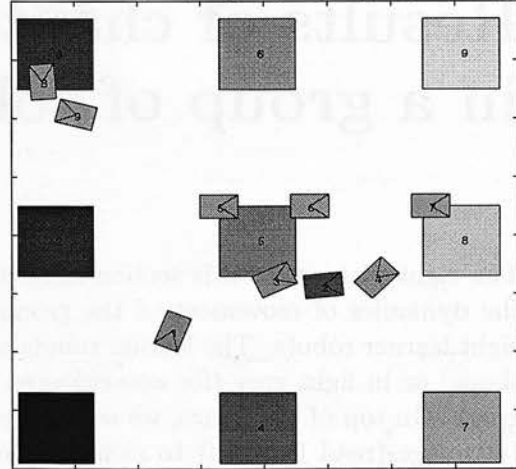
# Results of chapter 8: Grounding in a group of robots

The eight pictures of this section show snapshots of one simulation, which illustrate the dynamics of movement of the group of robots and the learning progress of the eight learner robots. The learner robots are in blue-green (for coloured versions of the thesis) or in light grey (for non-coloured ones). The teacher robot is in red (or dark grey). On top of the figure, we see the result of each robot's speaking. At step 1, i.e. 1st image (read from left to right, top to bottom), the teacher robot (2nd column) speaks, sending the radio signal '16' to label the coloured patch on the middle of the arena, onto which it runs. The learner robot No1, which follows it, says nothing, i.e. it outputs '0'. It has not yet see the object and has not yet made the signal-object matching. The same happens at step 13 (3rd image), where the teacher robot sends the signal '128' to label the patch on the middle of the right side of the arena, which it detects by being close enough to see it (note that the robots detect the patches from a short distance away from them). However, at steps 31, 45, 87 and 105 (images 5 to 8), the learner robot (i.e. learner No 3 in images 5, 6 and 8 and learner No 9 in image 7), which follows most closely the teacher robot, emits the correct signal, that is it outputs the same signal as the teacher. This shows that it has made the correct signal-object association, as it is now close enough to detect the patch and to associate this observation with the teacher's signal. Images 2,3 and 4 shows how the robots manage to form columns, by competition of avoidance and following behaviours.

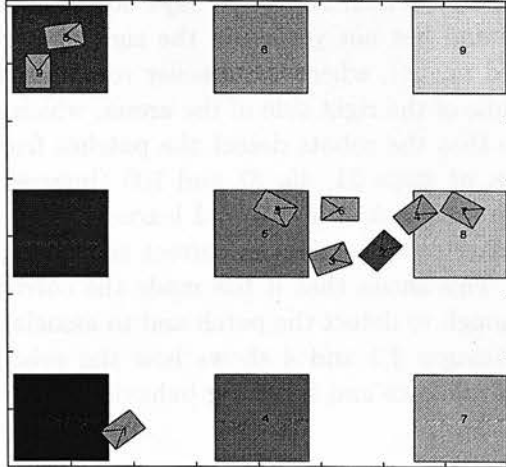
Dialogue: 1 2 3 4 5 6 7 8 9  
0 16 0 0 0 0 0 0 0



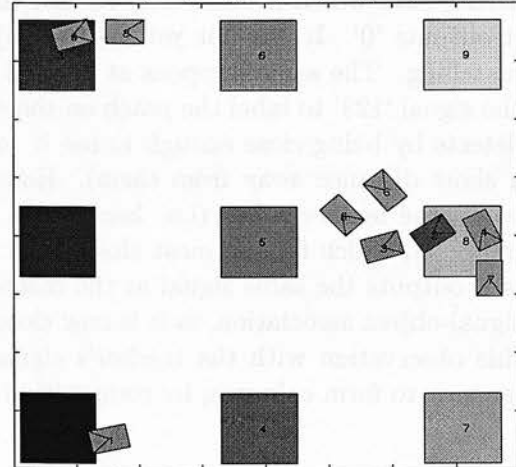
Dialogue: 1 2 3 4 5 6 7 8 9  
0 0 0 0 0 0 0 0 0



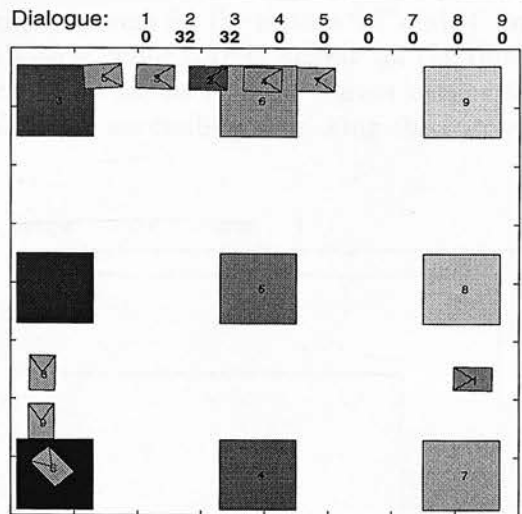
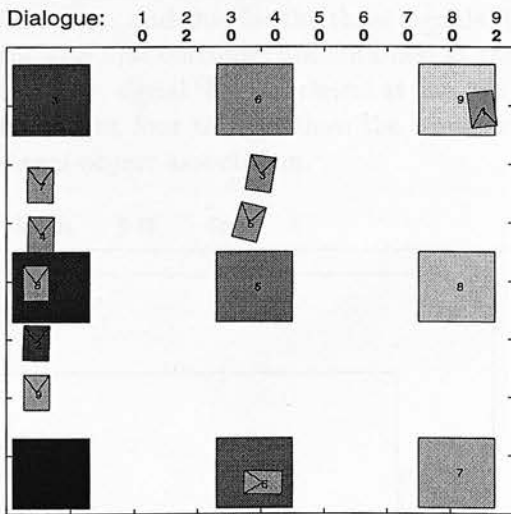
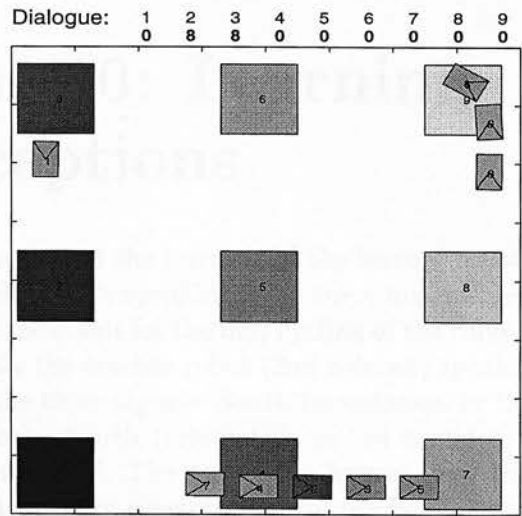
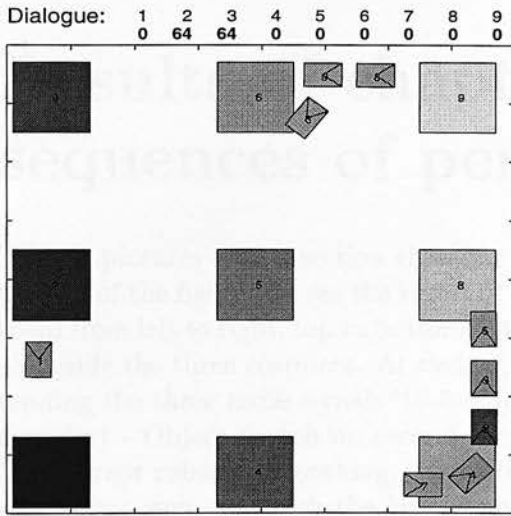
Dialogue: 1 2 3 4 5 6 7 8 9  
0 128 0 0 0 0 0 0 0



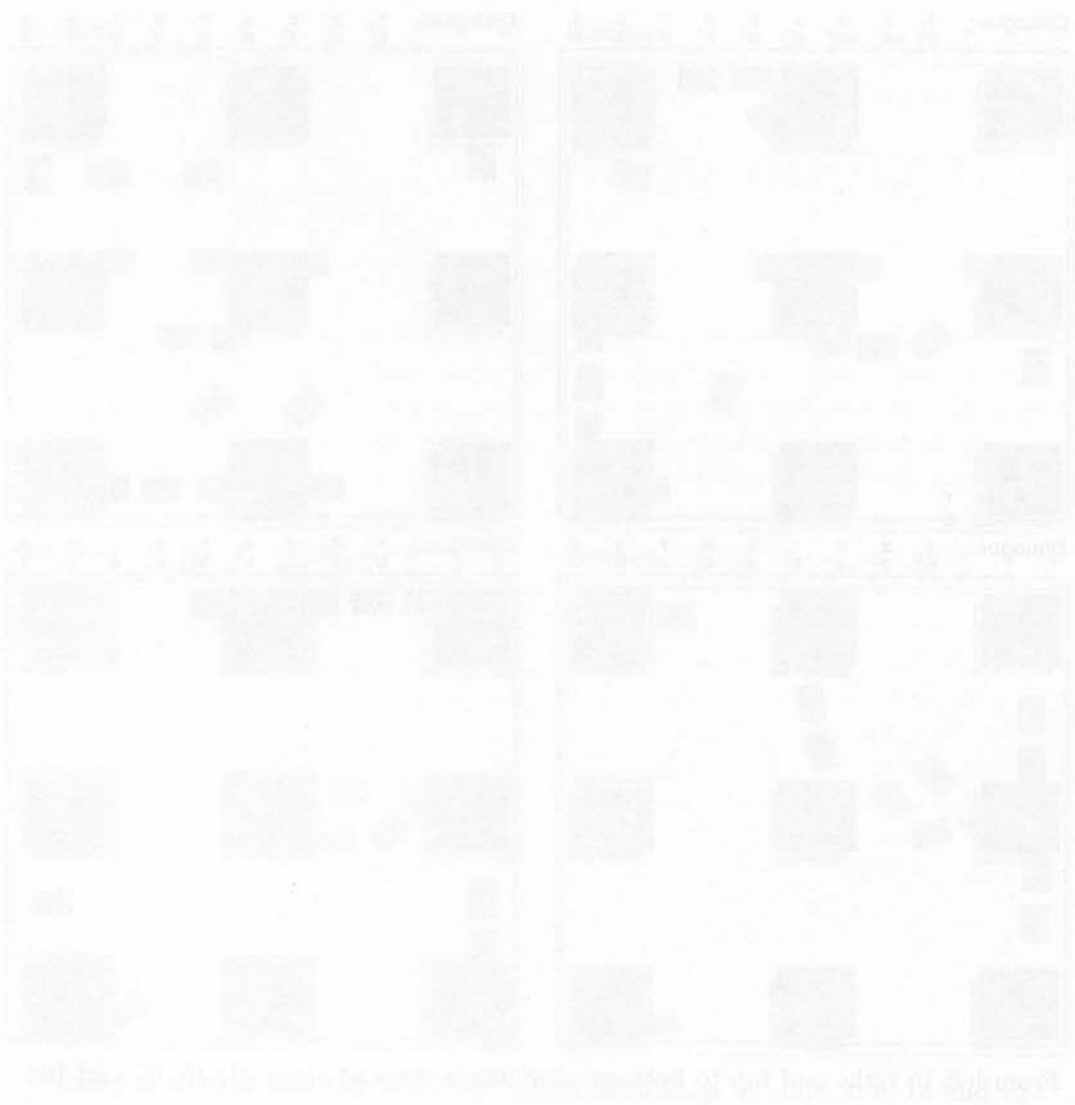
Dialogue: 1 2 3 4 5 6 7 8 9  
0 0 0 0 0 0 0 0 0



From left to right and top to bottom: simulation stage at steps 2, 8, 13 and 18.



From left to right and top to bottom: simulation stage at steps 31, 45, 87 and 105.

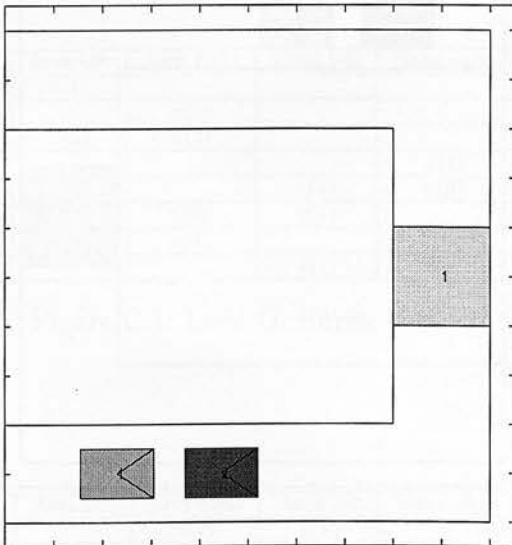


## Appendix B

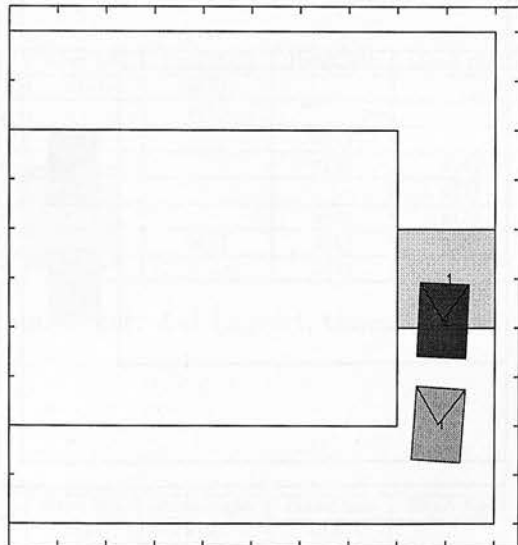
# Results of chapter 10: Learning sequences of perceptions

The six pictures of this section show the progress of the learning of the learner robot. On top of the figure, we see the result of each robot's speaking. The three first images (read from left to right, top to bottom) show the result for the first cycling of the robots alongside the three corridors. At cycle 1, only the teacher robot (2nd column) speaks, sending the three radio signals '16-4-8' for the three signals 'South (orientation in 1st corridor) - Object (patch on second corridor) - North (orientation in 3rd corridor)'. The learner robot says nothing, i.e. its output is '0'. The next three figures show the simulation step, at which the learner robot answers correctly to the teacher for the first time, and this for the three signals. Correct answers for the signals '16' and '4' for the two first corridors are obtained at the 5th cycle, while correct answer for the third corridor, signal '8' is retrieved at the 4th cycle. This means that the learner needed to be taught four to five times the same signal before succeeding in making the correct signal-object association.

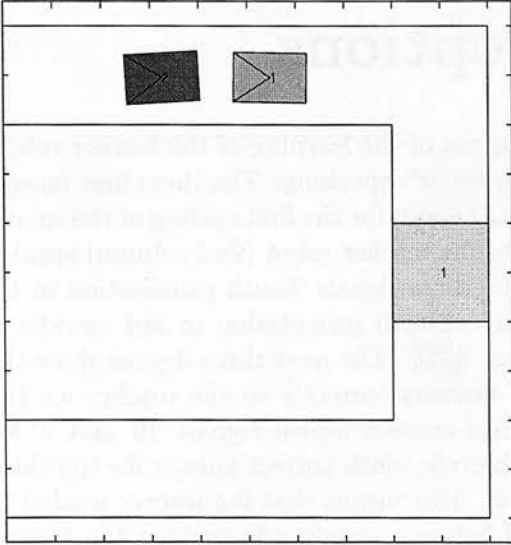
Dialogue: 0 16      Cycle: 1



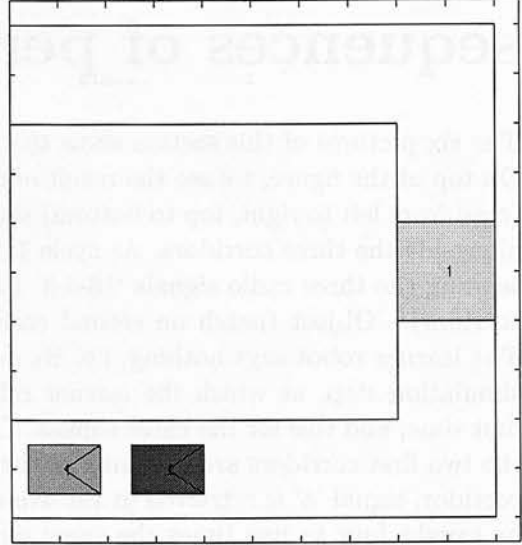
Dialogue: 0 4      Cycle: 1



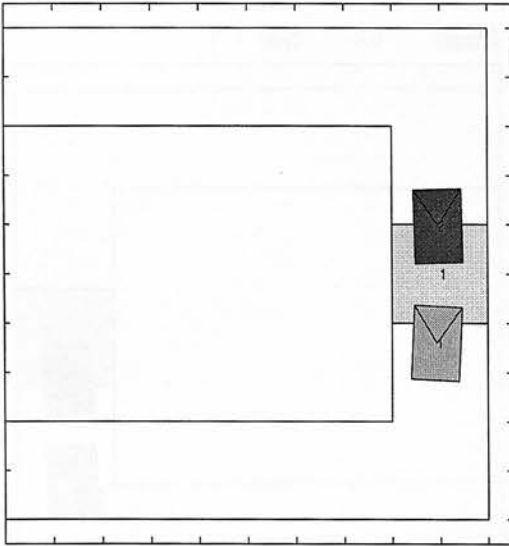
Dialogue: 0 8 step: 1



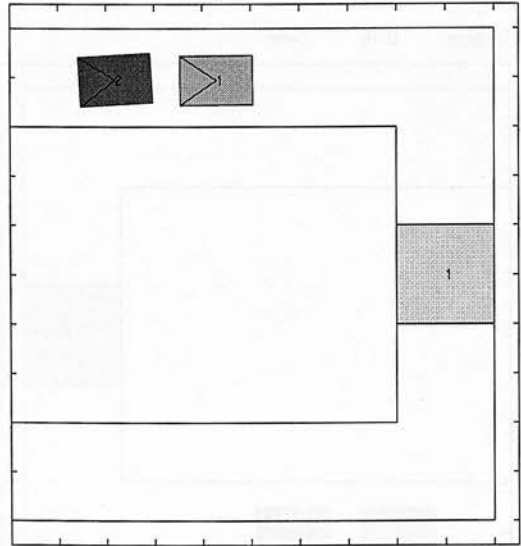
Dialogue: 16 16 Cycle: 5



Dialogue: 4 4 Cycle: 5



Dialogue: 8 8 Cycle: 4



# Appendix C

## Results of chapter 11: Experiments with Robota, the doll robot

### C.1 Results of section 11.3.1: learning dance pattern

The tables below show ten different teachings of action sequences, performed by 3 different persons. The names of the persons as well as the average timing it took them for the teaching is given in the tables' captions. The columns of the table represent the four different actions the robot could perform. The lines of the tables refers to each key of keyboard. For each key, there is a different combination of actions. The ordering of the actions is given in the number following the 'x', e.g. 'x(1)' and 'x(4)' means first and fourth action of the sequence respectively.

	Arm left	Arm right	Head left	Head right
1	x(1)			
2		x(1)		
3	x(1)	x(2)		
4				x(1)
5			x(1)	x(2)
6		x(2)	x(1)	
7	x(2)	x(1)		
8			x(1)	x(2)

	Arm left	Arm right	Head left	Head right
1	x(1)	x(2)		
2		x(1)		
3			x(1)	
4			x(2)	x(1)
5	x(2)			x(1)
6			x(1)	x(2)
7		x(2)	x(1)	x(3)
8	x(2)		x(1)	

Figure C.1: Left: G. Hayes; time≈10-15 min. Right: A-J Ijspeert; time≈10 min.

	Arm left	Arm right	Head left	Head right
1		x(2)	x(1)	
2		x(2)	x(3)	x(1)
3	x(1)	x(2)	x(3)	
4	x(1)	x(2)		
5	x(3)	x(2)		x(1)
6			x(1)	
7				x(1)
8		x(1)		

	Arm left	Arm right	Head left	Head right
1	x(1)	x(2)	x(3)	
2	x(2)	x(1)	x(3)	x(4)
3	x(1)	x(2)	x(4)	x(3)
4	x(1)	x(2)		
5	x(2)	x(1)		
6			x(1)	x(2)
7			x(2)	x(1)
8		x(3)	x(1)	x(2)

Figure C.2: Name: A-J Ijspeert; time≈10 and 15 min. resp.

	Arm left	Arm right	Head left	Head right
1	x(1)			
2	x(1)	x(2)		
3	x(2)	x(1)		
4			x(1)	x(2)
5	x(2)	x(1)	x(3)	
6	x(2)	x(1)	x(3)	x(4)
7	x(4)	x(3)	x(1)	x(2)
8	x(3)	x(1)	x(2)	x(4)

	Arm left	Arm right	Head left	Head right
1	x(2)	x(1)		
2	x(1)	x(2)		
3	x(1)	x(2)	x(3)	
4		x(2)	x(1)	
5	x(3)	x(1)	x(2)	
6	x(1)			x(2)
7		x(2)	x(3)	x(1)
8	x(3)	x(4)	x(1)	x(2)

Figure C.3: Name: A. Billard; time $\simeq$ 15 and 10 min. resp

	Arm left	Arm right	Head left	Head right
1	x(1)			
2		x(1)		
3	x(1)	x(2)	x(3)	
4	x(2)	x(1)	x(3)	x(4)
5			x(1)	x(2)
6		x(2)	x(1)	
7	x(2)	x(1)		
8	x(1)	x(2)		

	Arm left	Arm right	Head left	Head right
1	x(2)	x(1)		
2	x(1)	x(2)		
3	x(1)	x(2)	x(3)	
4		x(2)	x(1)	
5	x(3)	x(1)	x(2)	
6	x(1)			x(2)
7		x(2)	x(3)	x(1)
8	x(3)	x(4)	x(1)	x(2)

	Arm left	Arm right	Head left	Head right
1	x(1)			x(2)
2	x(2)			x(1)
3		x(1)	x(2)	
4		x(1)	x(2)	
5			x(1)	x(2)
6	x(1)	x(2)	x(3)	
7	x(2)	x(1)	x(4)	x(3)
8	x(1)	x(2)		

	Arm left	Arm right	Head left	Head right
1		x(2)	x(1)	x(3)
2		x(1)		x(2)
3	x(3)	x(1)	x(2)	
4	x(1)	x(2)		
5			x(1)	x(2)
6			x(2)	x(1)
7		x(1)	x(2)	x(3)
8	x(2)	x(1)		

Figure C.4: Name: A. Billard; time $\simeq$ 10 min. for all

## C.2 Results of section 11.3.2: experiment on word combinations

The tables below show five different teachings of word combinations, performed by 5 different persons. The names of the persons as well as the average timing it took them for the teaching is given in the tables' captions. The columns of the table represent the keyboard keys, each key has a different meaning for each experiment, depending on the choice of the instructor. The lines of the tables show the key combination, which the robot was taught for its sensor and actuator activation. 'Hand L' means touching the switch in the left hand, while 'Arm L' means lifting the left arm. The 'x' in the tables represent the taught keys, the 'R' represent the key retrieved without teaching.

Choice:	I	move	you	touch	arm	up/down	left	right
Hand L								
Hand R								
Foot L			x	x	x			
Foot R								
Mouth								
Incl.	x	x	x			x		
Arm L		x	x					
Arm R		x	x		x			x
Head L		x	x				x	
Head R		R	R					

Table C.1: Name: R. Kortenmann; time $\simeq$ 20min.

Choice:	I	move	touch	left	foot	arm	head	-
Hand L	x		x	x		x		
Hand R								
Foot L	x		x	x	x			
Foot R						R		
Mouth							x	
Incl.								
Arm L	x	x		x		x		
Arm R								
Head L		R					R	
Head R	x	x						

Table C.2: Name: G. Hayes; time $\simeq$ 10 min

Choice:	I	move	foot	touch	arm	up-down	left	right
Hand L				R				
Hand R	x			x	x		x	
Foot L	x		x	x		x		
Foot R	x	x	x					x
Mouth								
Incl.								
Arm L	x	x			x		x	
Arm R								
Head L								
Head R								

Table C.3: Name: A. Espinoza; time $\simeq$ 5 min

Choice:	You	I	touch	turn	foot	head	left	right
Hand L								
Hand R								
Foot L		x	x		x		x	
Foot R								
Mouth						R		
Incl.								
Arm L								
Arm R								
Head L	x			x		x	x	
Head R	x			x		x		x

Table C.4: Name: A. Ijspeert; time $\simeq$ 10 min

Choice:	foot	head	arm	right	left	you	touch	I move
Hand L			x		x	x	x	
Hand R				R		R	R	
Foot L	x				x	x	x	
Foot R	x			x		x	x	
Mouth		x				x	x	
Incl.						x	x	
Arm L				x		x		x
Arm R			x	x				x
Head L								
Head R								

Table C.5: Name: A. Billard; time $\simeq$ 5 min



# Appendix D

## List of publications

An important part of the material presented in this thesis has been published as journal articles, book chapter and conference and workshop papers. These were all peer reviewed. The list is given below. In the text of the thesis, it was clearly referred to which paper each part was published in.

### Publications:

#### In Journal :

- “Experiments in social robotics: grounding and use of communication in autonomous agents“, Billard A., Dautenhahn K., accepted (Sept. 1998) for publication in *Adaptive Behaviour Journal*, special issue on 'simulation of social agents', K. Dautenhahn Guest editor.  
**Experiments of chapter 8 and section 9.2.**
- “DRAMA, a connectionist architecture for control and learning in autonomous robots“, Billard A. and Hayes G. (1999), Vol. 7:1 *Adaptive Behaviour Journal*.  
**Chapters 4 and 5, description of DRAMA.**
- “Grounding communication in autonomous robots: an experimental study“, Billard A., Dautenhahn K. (1998), In *Robotics and Autonomous Systems*, special Issue on "Scientific methods in mobile robotics", M. Recce and U. Nehmzow (eds.), No. 24, Vols. 1-2, 1998.  
**Experiments of section 7.2.2.**

#### In Book :

- “Transmitting Communication Skills through Imitation in Autonomous robots“, A. Billard and G. Hayes. in "Learning Robots: A Multi-Perspective Exploration", A. Birk and J. Demiris (eds.), LNAI Series, Springer-Verlag, 1998' (in Press).  
**Experiments of section 7.1.3.**

### In Conference or Workshop Proceedings :

- “Experiments on human-robot communication with Robota: an imitative learning and communicative doll robot“, A. Billard, K. Dautenhahn, G. Hayes. Proceedings of Socially Situated Intelligence Workshop, as part of SAB’98 conference, August 98, Zurich.  
**Experiments of chapter 11.**
- “Learning to Communicate through Imitation in Autonomous robots“, A. Billard and G. Hayes. Proceedings of ICANN97, 7th International Conference on Artificial Neural Networks, pp 763-768, Wulfram Gerstner et al. (eds.) Springer-Verlag, Lausanne, Oct 97, CH.  
**First experiment of section 7.1.3.**
- “Grounding communication in situated, social robots“. A. Billard and K. Dautenhahn. Proceedings of TIMR97, British Conference on Autonomous Mobile Robotics and Scientific Methods in Mobile Robotics Conference, Manchester 1997. Technical Report Series, Department of Computer Science, Manchester University, ISSN 1361 - 6161. Report number UMCS-97-9-1.  
**Second experiment of section 7.1.2 and experiment of section 9.1.**
- “Transmitting Communication Skills through Imitation in Autonomous robots“, A. Billard and G. Hayes. Proceedings of EWLR97, Sixth European Workshop on Learning Robots, Brighton, July 97.  
**Experiment of section 7.1.3.**
- “Robot’s first steps, Robot’s first words ...“, A. Billard and G. Hayes. In the proceedings of GALA’97, Groningen Assembly on Language Acquisition, Edinburgh, 4-6 April 97.  
**First experiment of section 7.1.2 and earlier version of the DRAMA architecture.**