Reinforcement Learning in Autonomous Robots: An Empirical Investigation of the Role of Emotions

Sandra Pinto Clara do Carmo Gadanho

Ph.D.
University of Edinburgh
1999

Abstract

This thesis presents a study of the provision of emotions for artificial agents with the ultimate aim of enhancing their autonomy, *i.e.* making them more flexible, robust and self-sufficient. In recent years, the importance of emotions and their assistance to cognition has been increasingly acknowledged. Emotions are no longer considered undesirable or simply useless. Their role in various aspects of human and animal cognition like perception, attention, memory, decision-making and social interaction has been recognised as essential. The importance of emotions is much more evident in social interaction and therefore much of the emotions research done in artificial systems focuses on the expression and recognition of emotions. However, recent neurophysiological research suggests that emotions also play a crucial part in cognition itself.

This thesis investigates ways in which artificial emotions can improve autonomous behaviour in the domain of a simple, but complete, solitary learning agent. For this purpose, a non-symbolic emotion model was designed and implemented. It takes the form of a recurrent artificial neural network where emotions influence the perception of the state of the world, on which they ultimately depend. This is done through a hormone system that acts as a persistence mechanism. This model is somewhat more sophisticated than those usually found in equivalent non-symbolic systems, yet the emotions themselves were restricted to a few simplified emotions that do not try to mimic the complexity of the human counterparts, but are afforded by the agent's interaction with the environment.

Several hypotheses were investigated of how the emotion model above could be integrated in a reinforcement learning framework which, by itself, provides the base for the adaptiveness necessary for autonomy. Experiments were carried out in a realistic robot simulator that compared the performance of emotional with non-emotional agents in a survival task that consists of maintaining adequate energy levels in an environment with obstacles and energy sources. One of the most common roles attributed to emotions is as source of reinforcement and was therefore examined first. In experiments with a controller that selects between primitive actions, the reinforcement provided by emotions was found inappropriate because of the time scale discrepancies introduced by the emotion model. The reinforcement provided by emotions proved to be much more successful when used by a controller that selects between behaviours rather than actions, achieving equivalent performance to that of a standard reinforcement function. One of the crucial issues for efficient and productive learning, highlighted by the latter experiments, is to determine exactly when the controller should re-evaluate its decision concerning which behaviour to activate. The emotions proved to be particularly helpful in this role, enabling better performance with substantially less computational effort than the best suited interruption mechanism using regular time intervals. The modulation of learning parameters such as learning rate and the exploration *vs.* exploitation ratio was also explored. Experiments suggested that emotions might also be useful for this purpose.

This research led to the conclusion that artificial emotions are a useful construct to have in the domain of behaviour-based autonomous agents, because they provide a unifying way to tackle different issues of control, analogous to natural systems' emotions.

Acknowledgements

First of all I would like to thank my parents for all that they have given me and in particular for the education which they provided me with.

I am also grateful to the many people who provided the support and motivation that made it possible for me to start the PhD in the first place. In particular, I would like to thank Professor Steiger Garção for his invaluable support and to Dr. Luís Correia for introducing me to mobile robotics research during my undergraduate degree.

Next, I would like to thank my supervisors John Hallam and Chris Malcolm for their help. In particular, I would like to thank John Hallam who always found time for me in his busy schedule and who provided numerous helpful comments on the many earlier drafts of my thesis and other publications.

For discussions and the friendly atmosphere provided by the Mobile Robots Group at the Department of AI, special thanks to Bridget Hallam, Simon Perkins, Richard Reeve, Herbert Peremans, Tim Taylor, Arturo Espinosa-Romero, Mike Spratling and Heba Al-Lakany. Very special thanks go to Jeremy Wyatt, who pointed the way to the relevant Reinforcement Learning literature, and Yiannis (John) Demiris who on top of everything else proof-read a draft of my whole thesis and gave me some very useful comments.

My most special thanks are to Nuno Chalmique Chagas who stood by me during all these years giving me full support and endless hours of fun.

I also want to thank my examiners, Dr. Gillian Hayes and Dr. Rosalind Picard, who gave me very insightful comments on this thesis which helped to improve the final version.

There were also many other researchers in the field who inspired me with their work and who helped me by their comments in conferences and through e-mail. To those and all the other people who helped me along the way that I have not mentioned my thanks and apologies.

Finally, I would like to thank the Portuguese programme PRAXIS XXI for awarding me scholarship BD/3624/94 which provided financial support for my research.

Declaration

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

Sandra Pinto Clara do Carmo Gadanho
Edinburgh
June 22, 1999

Contents

List of Figures

Chapter 1

Introduction

1.1 Project Aims

On account of its preoccupation with knowledge as the cause of intelligent behaviour, Artificial Intelligence today faces several problems. It has been said that Artificial Intelligence is having difficulties in packaging common sense into knowledge systems by the discovery of more sophisticated rules, because such a task is not possible (Varela, 1992), or that there are serious embodiment and grounding demands which have been ignored (Brooks, 1986a; Harnad, 1989). What many seem to agree upon is that more research should be made on self-referential systems that perceive the world in a contextual way, strongly influenced by their embodied and individual history (Varela et al., 1991) .

It seems that trying to develop a better understanding of the mechanisms underlying autonomy might provide some of the necessary tools to overcome these difficulties of Artificial Intelligence.

In the field of robotics, the criteria used to define whether a robot is autonomous or not are not well established. In general, simply requiring that, once the robot is finished, it does its task without human intervention is enough. The word's root meaning suggests an alternative definition of autonomy that has stronger requirements: namely, a truly autonomous robot should also develop the laws that govern its behaviour.

The design of truly autonomous systems is still a very open research subject, particularly if one requires the meaning of autonomy to include self-motivation, instead of mere automaticity. However, there is much argument in its favour. Having autonomous agents seems to be a clear advantage in several diverse fields, like for example robotics, animal robotics[1] (McFarland, 1994), agents theory (Ferguson, 1992) and interactive virtual environments (Blumberg, 1995). Although, arguments against having fully autonomous agents are easy to find (in general, the robot is supposed to do something useful and not whatever it wants, which might even be detrimental), it is generally accepted that it is beneficial to have autonomy, at least to a certain extent which is still far from being achieved in today's systems. It is often difficult or even impossible for the designer to anticipate all possible scenarios the robot will be confronted with, and autonomy can help the robot to deal with the unexpected situations.

The way to accomplish true autonomy in a robot is by developing an adaptive controller that improves its performance by unsupervised learning when interacting with its environment. Such improvement of performance must be and is always grounded in some kind of value scheme defined by the designer. Whether this scheme involves reinforcement values, instincts, credit systems, goals or heuristic rules the result is always the same: to give the robot some guidance in its learning task. For an autonomous agent, detecting the regularities in the environment by self-organisation is not enough as a learning capability; it also needs some sort of internal motivation to decide what to do. It is also necessary for it to have some way to establish its goals without the aid of an external teacher. For this purpose it needs to be endowed with some innate mechanisms that allow it to determine what are the crucial features of its interaction with the environment and whether there are positive or negative connotations associated with these features. These basic learning mechanisms have to be hard-wired by evolution, if the agent is to learn anything useful during its lifetime. A generic evaluation in terms of survival abilities can only be useful for natural selection through genetic evolution.

The fact that emotions are considered to be essential to human reasoning suggests that they might play an important role in achieving the self-motivation necessary to support strong autonomy.

The present research focuses on how to use emotions in the control of an autonomous

[1] Modelling of animal behaviour using robots.

agent that adapts to its environment using reinforcement learning techniques. The social connotations usually associated with emotions might suggest that this work also addresses social robots, but this is not the case. Emotions were used solely in the autonomous control of a single solitary agent.

A view shared by many researchers in the emergent field of emotional agents is that emotions serve a purpose in cognition and that it is this functional aspect of emotions that should be taken into account when modelling emotional agents (Frijda and Swagerman, 1987; Cañamero, 1998). In particular, researchers should be careful to avoid getting their attention caught by specific human emotions that probably do not even make sense in terms of the artificial agent-environment interaction. This is a view akin to the Artificial Life methodology (Langton, 1992) that does not consist of trying to imitate biology by constructing realistic models, but of trying to abstract the fundamental principles underlying biological phenomena and recreate them in artificial systems.

This was the approach followed by the current work. Several functional roles of emotions were tried out under an animat philosophy (Wilson, 1991), by building a complete agent where emotions form an integral part of the whole. Furthermore, these functional roles were tested in comparison with other non-emotional mechanisms. It was considered important not only to develop a fully functional agent that successfully performs the task that it is devised for, but also to demonstrate that the introduced mechanisms are advantageous when compared with more traditional mechanisms. More important than designing an agent that solves an arbitrary task is to establish the utility of mechanisms used.

In order to establish if there is an advantage in having emotions as the source of self-motivation in an autonomous robot, experiments were carried out on a simulated Khepera robot (Michel, 1996) in an animal-like adaptation task. The experiments focus on how to use emotions in the control of the robot, and in particular in its adaptation to the environment. The utility of different roles of emotions was explored in terms of the adaptiveness of the robot's final behaviour.

For this purpose an emotion model was designed and implemented. This is a simple model based on a recurrent network, where perception and emotions influence each other. Through this mutual influence some persistence of emotional state is achieved, while maintaining reactiveness to new perceptual states. The model endows the agent with emotional states that are coherent with its contextual interaction with the environment by attributing value to the relevant features of this interaction. The robot's possible emotional states were named after four basic human emotions — Happiness, Fear, Sadness and Anger — but are much simpler than the human counterparts.

Apart from the influence on perception imposed by the model used, emotions were used in the reinforcement learning framework to fulfill the following roles: reinforcement specification, detection of significant events, modulation of the learning parameters of learning rate and the ratio of exploration/exploitation.

The results showed that emotions can be used successfully as a source of reinforcement if, and only if, the controller architecture is selected with care. It was found necessary to upgrade from an action-based to a behaviour-based architecture in order to have the emotion-based reinforcement work properly. The intrinsic time scales of a behaviour-based architecture were found more appropriate for the use of emotions. If the robot has to select and evaluate a primitive action at each time step the emotions' persistence in time becomes a severe hindrance for their successful use as reinforcement. The use of emotions to modulate the learning parameters of the action-based controller proved much more promising.

The behaviour decomposition of the controller introduced the need for determining when to trigger control, *i.e.* when to re-evaluate the previously selected behaviour and select a new one. It was found empirically that selecting the control triggering mechanism correctly was a crucial step towards success in the learning task. Based on the fact that the agent's emotional state always reflects the occurrence of significant events, an event-detection mechanism was designed that consisted of triggering control whenever a significant change in emotional state was found. This emotion-driven event detection mechanism was experimentally compared with triggering control at regular intervals and proved to be helpful for the robot's learning.

The influence of emotions in perception was also examined, but results failed to show any difference between the performance of agents with emotion-influenced perception and non-emotional perception. However, these results are probably controller and task dependent.

In essence, the reported research work shows how emotions can influence control in multiple ways. Although the "emotions" used were much simplified, they were still named emotions as they tried to capture more functional aspects than those provided by a traditional reinforcement function. Moreover, calling them emotions enables this research to be identified with other emotion research so that developments in the field may be brought together and integrated to produce further richness of emotions functionality and added complexity of artificial agent's behaviour.

1.2 Thesis Outline

The rest of this thesis is organised as the following chapters:

Autonomous and Learning Robots (Chapter 2)

> Survey of autonomy's definitions in the field of robotics and establishment of a working definition, followed by an overview of robot control architectures in general and reinforcement-learning techniques in particular. Review of a reinforcement-learning architecture similar to the one used in the robot experiments.

Natural and Artificial Emotions (Chapter 3)

> Literature review on emotions in natural and in artificial systems. Presentation of the emotion model used in the experiments.

Action-Based Control (Chapter 4)

> Experiments employing an action-based learning controller. The emotions system was integrated with the control system by influencing perception, providing reinforcement value and modulating learning parameters.

Behaviour-Based Control (Chapter 5)

> Experiments employing a behaviour-based controller. Emotions were used within

the behaviour based architecture in three different roles: influencing perception, providing reinforcement value and detecting events for control triggering.

Concluding Discussion (Chapter 6)

General discussion of the achievements attained through the integration of emotions in the two different controllers and of the possible extensions and directions of future research.

Recapitulation of the issues and problems inherent to autonomous robot research that were found in course of this work.

Appendices

Presentation of further experimental details and of publications regarding the work addressed by this thesis.

Chapter 2

Autonomous and Learning Robots

2.1 Introduction

This chapter will start by examining the meaning of autonomy when applied to robotics. Autonomy is one of the most used words in robotics, yet there is no real consensus on what it means. An autonomous system is by definition a self-governing system. This definition has been given several interpretations in the field of robotics ranging from automatic (*i.e.* it works without human intervention) to self-motivated (*i.e.* it defines the rules that govern its behaviour). Sometimes autonomy is even identified with self-sufficiency and the robot is said to be autonomous if it is able to recharge itself without assistance. In fact, most of the research that is said to be done in autonomous systems is more concerned with other issues, such as navigation, learning, adaptation and self-sufficiency than autonomy itself. The autonomy definition adopted in this thesis is that of complete self-government, *i.e.* not only following one's rules but also making them. However, autonomy will not be defined as an all-or-nothing property but as one of degree: the extent to which the agent is self-governing.

Within the discussion about autonomy in robotics, some considerations are presented about the requirements and difficulties of robot autonomy. This discussion is structured in terms of automaticity, self-sufficiency, self-referentiality, self-controlling, self-motivation, autopoiesis and degrees of autonomy; concluding with the presentation of some guidelines for autonomous robot design.

Next in this chapter, some other issues relevant to autonomous robot design are presented, namely: the importance of reactivity and low level control in autonomous robots; some of the difficulties faced by the necessary decomposition of behaviour in complex tasks; the appeal of generalisation and the advantages of specialisation; the limitations of knowledge representation and the need for strong interaction between perception and behaviour.

This is followed by a short overview of learning and adaptation mechanisms that pays particular attention to reinforcement learning. The short review of reinforcement-learning is a basic introduction which describes its advantages and presents some of its problems and tentative solutions.

Finally, this chapter concludes with presentation of the basic controller architecture that was selected for the present work. The description will focus on a reinforcement-learning architecture that employs the basic mechanisms used in the two learning controllers developed for the experimental work reported by this thesis.

2.2 Autonomy

2.2.1 Automaticity

Autonomous — f. Gr. *αύτόνομος* making or having one's own laws, independent (f. *αύτο-* self, own + *νόμος* law) + -ous (The Oxford English dictionary, 1989)

Autonomy is a word formed by 'autos' (self) and 'nomos' (rule or law). This can mean either making or having one's own laws. In robotics, autonomy is much more often associated with having one's laws than with making them. Usually, autonomy is said to have been achieved when the system can fulfil its goal without human intervention or intervention from any other system (*e.g.*, Blidberg, 1989; Yavnai, 1989; Kirchhoff, 1989; Giralt et al., 1989). If, after being constructed and programmed, the robot is left alone doing successfully whatever task it is meant to, then the robot is autonomous.

In nature, the definition of the behavioural rules of the simplest creatures' behaviour is done mostly through evolution rather than by the creatures themselves. Nevertheless, higher creatures do have the power to adapt during their lifetime by defining their own rules and can for that reason be considered more autonomous.

Besides being responsible for following the rules on its own, which can be considered a weak form of autonomy, a definition of autonomous systems should also require the system to develop its rules, which is a much stronger demand. However, this definition should be regarded as the extreme of autonomy as examples of weaker autonomy can be found that do not fully comply with it.

The greatest advantage of self-ruled systems over other systems is that they are able to step outside the boundaries of what was foreseen by the designers (Steels, 1994b; Reeke, Jr., 1996). This way, their capacity to deal with the infinitely rich and dynamically changing real world is increased.

2.2.2 Self-sufficiency

Autonomy is sometimes identified with self-sufficiency. Self-sufficiency can be seen as a requirement if one is very strict about having a system able to keep going without external assistance. If the agent can recharge itself without outside assistance then it can carry out its task over a longer period of time, and therefore it will be more independent, i.e., “autonomous”.

However, as McFarland (1992) points out, autonomy and self-sufficiency are very different concepts, that can easily be distinguished by a simple example: an over-domesticated lap dog. Although it would probably not survive without the free meals given by its owner, it is considered to be autonomous.

In robotics practice, the distinction between self-sufficiency and autonomy can also be drawn with two simple examples: a robot with energy supplied by batteries that does not recharge itself and a robot with energy supplied by an umbilical cord. The first robot is usually considered more self-sufficient, but its life-span will be very short on account of the small amount of energy available from the batteries. In a few hours the robot will not have had enough learning experiences to be able to acquire complex autonomous behaviour. Its autonomy will be seriously limited. This will not happen in the case of the second example. That robot will not have time limitations to constrain its behavioural complexity, but the robot will not be self-sufficient. Nevertheless, if the vehicle is to be tethered to some place this will impose limitations on its movements. For some types of goal this can also be a serious drawback. Sub-sea and space exploration are some good examples, because they imply unbounded environments. But even in bounded environments wires can easily get tangled unless serious restrictions are made to the environment in order to avoid the problem.

A self-sufficient robot, in the sense that it is able to replenish its energy, maintaining itself viable for long periods of time without human intervention, is something to seek for. Nevertheless, the robot will be indirectly dependent upon humans, because it was designed for a particular man-made niche where energy is provided by humans, and will not survive outside that niche. It is not easy to achieve self-sufficiency in a human-made ecosystem. Humans themselves depend too much on each other to be considered self-sufficient and yet no doubts are ever cast over human autonomy.

In general, too much emphasis is attributed to the importance of self-sufficiency of autonomous robots. However, self-sufficiency is an excellent way to test autonomy more in the sense that it provides a source for self-motivation than for independence.

2.2.3 Self-referentiality

Bourgine and Varela (1992) suggest two alternative ways of viewing a system regarding autonomy:

- **Heteronomous**

  It is addressed as an input and output device whose output is the result of some internal processing of the input.

- **Autonomous**

  The centre of attention is placed on emergent behaviours and internal self-organising processes which define what counts as relevant interactions.

In the case of an autonomous system, it is the nature of the internal dynamics of the system that determines how the arriving interactions are interpreted, rather than reacting to inputs in terms of externally supplied (by the designers) semantics.

Autonomous systems are not defined by their inputs and outputs. They can be perturbed by independent events and undergo internal changes which compensate for these perturbations. Whichever series of internal changes takes place, however, they are always subordinated to the maintenance of the system's organisation, *i.e.* the relations that must exist among its components for it to be of a specific class. Autonomous systems actively maintain an identity which is independent of their interactions with an observer, by keeping their organisation as an invariant. The heteronomous systems' identity depends on the observer, who specifies it by their inputs and outputs.

It can be argued that a system is more or less autonomous to the extent to which it can be said to be a self-sufficient cause. However, it should be noticed that much of the difference between an autonomous and a heteronomous system lies in the point of view. For example, humans can be regarded as heteronomous vehicles of the "purposes" of their "selfish" genes (Dawkins, 1976).

2.2.4 Self-controlling

Most present-day robots are automata, because their behaviour is entirely controlled by an outside agent. When in a particular state, they obey a particular behavioural rule that is externally imposed. The rules are influenced both by environmental conditions and by the robot's own behaviour, but do not depend upon the robot's history.

According to McFarland an autonomous system is self-controlling. It has the knowledge and motivation to control its own behaviour. An important implication of autonomy is that the autonomous agent cannot be completely controlled by an outside agent. This happens because the system is not completely observable (McFarland, 1992).

Autonomous agents are self-interested and will choose their actions according to their own motivations. Like dogs and cats, they are self-controlling and controllable only to a limited extent by outside agents.

Nevertheless, a self-controlling robot can be made useful. The robot can actually want to do the task it is needed for, or in the worst case it can be tamed to "like" it. If, on the other hand, the robot only does exactly what it has been told then it will suffer from lack of opportunistic and improvisation capabilities. Being able to do more than what it is told explicitly or implicitly by the programmer can be a great advantage to the autonomous robot by giving it the ability to deal with what was not anticipated by the programmer.

2.2.5 Self-motivation

It is often stated that an autonomous agent should exhibit goal-directed behaviour (Covrigaru and Lindsay, 1991). In particular, that it should have multiple goals from which to select at any given time (Covrigaru and Lindsay, 1991; McFarland and Spier, 1997).

To emphasize the existence of a hierarchy of goals, the goals themselves are often attributed a secondary role as the means to satisfy some internal motivations. An example of this is the formal framework for the autonomy definition proposed by Luck and d'Inverno (1995). In this framework, an autonomous agent is defined as an object with goals and motivations and some potential means of evaluating behaviour in terms of the environment and these motivations. Its motivations are desires or preferences that can lead to the generation and adoption of goals, while its goals are simply states of affairs to be achieved in the environment. Brustoloni (1991) asserts a similar partition, but in terms of goals and drives. Again, goals are only attributed an instrumental function, while the agent's actions are ultimately directed by its drives.

As Covrigaru and Lindsay (1991) point out, the ultimate goals of an autonomous agent, or its motivations, should be of a homeostatic nature. The agent should not have a bounded task of accomplishing some reachable goals, but its task should consist of maintaining a few homeostatic goals.

A still greater degree of autonomy than the sole self-generation of the subgoals that govern the agent's behaviour is motivational autonomy (Cariani, 1992a), *i.e.* the self-generation of the performance evaluation mechanisms that guide the selection of the agent's subgoals.

2.2.6 Autopoiesis

Animals are the most distinctively autonomous entities that we know. They are goal-oriented, adaptive, opportunistic, plastic and robust (Beer, 1990). Some effort should be made in trying to understand what makes them autonomous without falling into the temptation of trying to imitate everything, even those properties which are obviously inadequate to model with the available technology.

When looking into animal autonomy, one cannot avoid looking into autopoiesis. Autopoiesis (Maturana, 1969; Maturana and Varela, 1973; Varela, 1979) is a concept that was created to overcome the difficulties in trying to define a living being. A living being is usually defined by a list of properties including chemical composition, capacity to move or reproduce. However, this kind of approach to defining living beings has many faults and seems always to be context dependent. Autopoiesis is what distinguishes the living from the non-living: the fact that living entities are continually self-producing and the producer cannot be separated from the product, *i.e.* the living entity is the continuous producer of itself.

Autonomy is usually included among the properties of the list that attempts to define living beings; but, just like the other properties, it can been seen as a consequence of autopoiesis. Animals produce themselves, and by doing so, they produce the rules by which they act. Their behaviour is the result of the internal correlations between the sensing and the action that they self-produced. Autopoiesis is a sufficient property for attaining a system's autonomy.

2.2.7 Degrees of autonomy

Some researchers (*e.g.*, Luck and d'Inverno, 1995) take autonomy as an all-or-nothing property: either a system is autonomous or it is not. Yet, if we try to appeal to our common knowledge, we find it very difficult to say whether some of the things that surround us are autonomous or not. Usually, we have the tendency to say that something is more or less autonomous. Even biologists do not agree among themselves on which living beings are autonomous. Some endow every living thing with autonomy (Maturana and Varela, 1987) while others are much more selective (McFarland, 1992).

Boden (1993) defends the view that there are different degrees of autonomy and, furthermore, that there are several dimensions to autonomy:

- The extent to which responses to the environment are direct or indirect (*i.e.* mediated by inner mechanisms dependent on the creature's history).
- The extent to which the controlling mechanisms are self-generated rather than externally imposed.
- The extent to which inner directing mechanisms can be reflected upon and/or selectively modified.

According to Boden the degree of autonomy of the system increases with the extent to which the controlling mechanisms are self-generated rather than externally imposed. Boden's view is supported by much of what has been discussed in the previous sections: Autonomy is a multi-faceted concept with many gradations.

Yavnai (1989) proposes an alternative definition of degrees of autonomy which entails a more practical point of view that reflects the current development of robotic technology. Some of the factors for measuring autonomy proposed by that author are:

- the degree of abstraction of the commands received by the system, *i.e.* the system is more autonomous if it can deal with higher level commands instead of only primitive actions;
- the duration for which the system can function without external intervention, which is usually very short in the case of mobile robots whether due to short battery life or frequent system breakdowns[1];
- the amount of complexity and uncertainty that has to be dealt with by the system.

[1] These breakdowns often derive from software design problems, but are also associated with the need for hardware maintainance or repair.

2.2.8 Autonomous robots' design

The previous discussion presented several definitions of autonomy, showing how multifaceted this concept can be. For this reason, the design of a robot that exhibits a high degree of autonomy has to consider the fulfilment of various abstract conditions which can make the design process difficult. Looking at animals for concrete instances of autonomy can provide some assistance to this process.

A few basic behavioural capabilities extracted from animal behaviour (Hallam and Hayes, 1992) that can be taken into consideration in the design of an autonomous robot are:

Perception — *The robot should be sensor-rich, both in terms of types of sensors and quantity of information provided by each sensor type.* An important challenge to the autonomous robot is being able to deal with a rich perception in a timely fashion, namely by expeditious mechanisms of focusing attention.

Movement — *The robot should be able to competently move around its environment and perform more elaborated actions such as moving objects.* The movement repertoire of the agent should provide flexibility of choice.

Homeostatic Goals — *The robot should have a few internal variables to keep within bounds, an example of such a variable being energy level.* This can serve as the basis for its internal motivation. Furthermore, it is important that the robot can function unattended for long periods of time. For this reason some care should be taken to avoid the robot life being shortened by lack of energy or the need for assistance in recharging.

Reactions and Learning — *The robot should be able to exhibit quick reactions to some of the stimuli of its environment and still be plastic to learn the relevance of the stimuli.* The agent should have adaptive capabilities, but those should not curtail its performance.

Navigation — *The robot should have a home base to return to.* Although it might not be essential for an autonomous agent, the ability to return to some referential points of its environment allows the agent to employ more complex behaviour.

This list provides a useful compilation of simple guidelines that should be taken into consideration in the design of an autonomous robot as they provide an adequate basis for autonomous behaviour. These were therefore used in the current work for the design of an autonomous agent controller and its task.

Learning is an important ability for an autonomous agent because it endows it with the necessary plasticity to be independent. However, learning should not compromise autonomy and therefore has specific requirements when applied to autonomous agents. To begin with, the learning mechanism should be sufficiently plastic to deal with the problems the agent faces without requiring much domain-dependent external parametrisation. In particular, there should be no external adjustment of learning parameters while the agent is performing its task. Finally, the agent should be able to learn on-line by itself, and not by carefully chosen examples given by external assistance. Furthermore, it should do so in a efficient and robust manner. Unfortunately, these requirements are quite hard to obtain with the available learning algorithms and in general some compromises have to be made.

People often propose a constructivist approach to the design of an autonomous system. This consists of the design of a self-organising system made of small and simple constructional blocks and simple self-organising rules moderated by some internal set of motivations (Luck and d'Inverno, 1995). One of the major drawbacks of this approach is relying on uniformity for greater plasticity when living organisms themselves benefit from having different types of components and connections (Winograd and Flores, 1986) and even specialised brain regions (Damásio, 1995). A uniform approach will probably suffer from lack of domain information to be able to cope with all the complexity of the outside world at once.

2.3 Issues in Architecture Design

Several approaches have been proposed for mobile robot control which have influenced the selection of the control architecture used in the experiments reported in this dissertation. This section briefly presents some of the main issues involved.

2.3.1 Reactiveness *vs.* deliberation

The classical Artificial Intelligence approach[2] to robotics relies on human-defined models of the world that the robot employs in its interaction with the world, assuming that intelligence is based on the representation and manipulation of knowledge. However, explicit deliberation about the effects of low-level actions is too expensive for the production of real-time behaviour in robots (Russell and Norvig, 1995). It makes the reactions of the robot to the external world slow and its performance very susceptible to slight environmental changes. These are serious drawbacks to the robot's autonomy. In general, the classical approach gives too much emphasis to methods for representing and manipulating knowledge while it ignores the dynamic properties of the robot-environment interaction (Verschure et al., 1992).

This approach has also been criticised for creating unnecessary symbolic abstractions that make sense in the programmer's view point but are ungrounded by the robot-environment interaction (Brooks, 1991). Instead of having the agent-environment interaction obstructed by an externally imposed formal description of the environment, systems should take advantage of a direct interaction. For instance, the physics of the sensory systems can be exploited in the discrimination of relevant stimuli (Hallam and Malcolm, 1993). This also supports the claim that the dynamic interactions between agent and environment can only be properly studied by building complete real agents (Smithers, 1992).

In recent years a new approach to the design of autonomous systems has been developed (Maes, 1991b). This approach tries to overcome some of the difficulties that the classical Artificial Intelligence approach faces when applied to the field of robotics. In order to have a more robust real-time performance, the new solutions try to avoid the use of knowledge-based rational choice and problem solving. Instead, they take advantage of a more direct coupling of perception to action which increases the system's distributedness, decentralisation and dynamic interaction with the environment.

Synthetic design (Donnett, 1992) or empirical bottom-up synthesis of the agent is also

[2] Also designated as Good Old Fashioned Artificial Intelligence (Haugeland, 1985), or GOFAI for short.

used, replacing the need for an *a priori* formal mathematical analysis. As Braitenberg (1984) defended and proved in theory, the basic organisation of living things underlying all their complexity is not, of necessity, complex in itself. Understanding a device that behaves in a complicated way is an uphill struggle, whereas actually building it might be quite easy. There are many examples of complex artificial behaviour achieved by surprisingly simple means, starting with Grey Walter's learning tortoises made of rudimentary electronic devices (Grey Walter, 1950, 1951). Simple solutions can also often explain the behaviour of natural systems (Webb, 1994; Jamon, 1991). On many occasions, it is preferable for animals to resort to simple approximations or tricks that can be achieved within the limited resources available, than to construct expensive abstract computations that provide the perfect solutions (Weher, 1987). The need for three-dimensional representation of Newtonian space (Weher, 1987) or explicit symbolic control (Liaw, 1995) can often be avoided. In the domain of collective robots the same principles can be applied and research has shown the emergence of complex group behaviour through the use of simple rules by individuals (Beckers et al., 1994; Melhuish et al., 1998; Mataric, 1995).

An effective demonstration of how sound practical results can be achieved through this approach was the subsumption architecture (Brooks, 1986b, 1989). In this architecture, the mobile robot control system was decomposed into task achieving behaviours that run in parallel, in opposition to the traditional decomposition of the control system into functional modules. The behaviours themselves are organised into layers of competence and have the ability to subsume behaviours in lower layers either by inhibition of outputs or suppression of inputs.

This methodology provides a clear example of direct robot interaction without the need of planning by means of an externally imposed model of the world. Nevertheless, there is no real plasticity in the rigid behavioural architecture developed. The problems that the robot has to face are once more transferred to the designers. The success of the agent depends solely on the ability of the programmer to describe the complete task domain (Verschure et al., 1992).

It can be right to defend (Brooks, 1986a) that the essence of being and reacting provides a necessary basis for the emergence of true intelligence, but it is not necessarily sufficient. Simulating this basis by a rigid automaton probably does not provide the necessary requirements for problem solving behaviour, language and expert knowledge. There are serious doubts as to whether this architecture can scale up in complexity towards full problem-solving (Russell and Norvig, 1995; Verschure et al., 1992).

Brooks' subsumption architecture was based on an evolutionary view of natural systems where layers of expertise are incrementally constructed upon older layers (Brooks, 1989). The methodology implicitly attributes to evolution the responsibility for all kinds of adaptation. In later research (Brooks, 1991), Brooks acknowledged the need for runtime adaptation and incorporated some forms of self-calibration in his system.

In Maes and Brooks (1990), the flexibility of the system is increased by learning the preconditions list associated with activation of each behaviour. This solution also tries to solve yet another problem with the subsumption architecture: the unnecessary loss of valuable computation time that can be avoided by suppressing not only the output value but also the computation of that value. Pebody (1995), on the other hand, proposed an enhancement of the subsumption architecture in terms of its basic units, the augmented finite-state machines, that allows on-line incorporation of complex sensory input into the unit's activation condition by associative learning.

The architecture developed by Maes (1989) provides more potential for the design of complex systems by having goals. In this architecture, the agent is also composed of a collection of competence modules: the actions. The fundamental difference consists of the existence of a selection mechanism that is an emergent property of the activation and inhibition dynamics among these modules. This modules are linked in a network of predecessor and successor links that are used to spread activation. Although this architecture has no global forms of control that might entail its robustness, it is goal-oriented. The final emergent behaviour is reactive, flexible and opportunistic; but also unpredictable (Maes, 1991a). How to achieve the desired global functionality is not always straightforward. Some plasticity has been introduced in this architecture by the introduction of the on-line learning of the links between modules (Maes, 1992).

The PDL language (Steels, 1994c) employs a dynamical systems approach where every process is always active and all process results are added, avoiding the need for action selection. On account of every process being very simple, complex systems seem to be difficult to design too. Nevertheless, several efforts have been made towards adding online learning to the architecture (Steels, 1994d; Boer, 1994) and therefore some sort of adaptation is provided by the system. Boer (1994) reported problems with instability when too many processes were available. For this reason, he introduced a two level hierarchy that groups several processes into a single behaviour. Only one behaviour is active at a time, depending on certain imposed criteria which Boer suggests be learned by genetic algorithms.

These architectures, where behaviour is an emergent property of the interaction of simple components, raise the problem of inverse emergence or behaviour generation (Prem, 1995). Finding the correct set of components and the right interaction dynamics between them can be both difficult and time consuming.

In time, the shortcomings of pure reactiveness have become evident and people have started to develop hybrid architectures. Hybrid architectures try to combine the immediate responsiveness to the current situation of the reactive architectures with the goal-oriented planning of the deliberative architectures. In general, this type of architecture is structured in several layers with separate layers for the reactive and the deliberative subsystems. This arrangement provides an elegant separation between the high-level goals of the robot and the local problems faced by the robot while pursuing these goals.

One straightforward example of how a hybrid architecture can be made useful is given by Malcolm and Smithers (1989). The system presented has two layers: one of them makes a sketchy plan and the other one intelligently executes it, simultaneously filling in the details of the plan. This architecture relates to the work of Agre and Chapman (1991) in that it transfers decision power from the planner to the executor. These authors argue that the central role of the plan, that deals with all the details of the task, should be reduced to a mnemonic device that the agent can resort to when deciding what to do next.

The GLAIR architecture (Hexmoor et al., 1993; Lammens et al., 1993) is a hybrid architecture with a three layer organisation. This architecture aims at consciousness, in the sense of being aware of one's environment. It has two different layers to address unconscious (automatic) behaviour and one to address conscious (reasoned) behaviour. One of the unconscious layers, named the perceptuo-motor level, consists of an automaton that is initialised with a very small primitive number of actions and sensations. The conscious layer notices and records the emergence of action sequences that make improvements and adds them to the perceptuo-motor level. The aim of the other unconscious layer is only to provide an extra level of abstraction by hiding away the low-level sensory input and actuators output.

Another example of a three-layer hybrid architecture is the Touring Machine (Ferguson, 1992) that aims to solve the problem of achieving real-time competent behaviour using limited resources. The lower layer provides reactive behaviour to deal with immediate problems. It uses simple symbolic rules for this purpose. The higher layers provide means to focus the agent's attention by changing those rules.

Another hybrid architecture that changes the dynamics of the reactive behaviour according to higher level intentions was proposed by Michaud et al. (1996). This architecture is different in that it is not organised in layers but in interactive modules that perform different roles. The behaviour-based module is in charge of producing actions efficiently. The motives module supervises the agent's performance by taking into consideration the information provided by the three different recommendation modules: the external situation, the internal needs and what the agent has learned about its world.

The information available to an agent is widely distributed both in time and space, requiring the agent to search for relevant information and recall past information. Opportunities and threads must be constantly monitored, although the global behaviour should have coherence in order to be able to successfully complete a task described by some, at least sketchy, planning done previously. For a system to achieve such aims under bounded computational resources, Wright (1994) proposed an emotional agent that has the ability to select between multiple goals, prioritise goals and decide on the level of commitment towards current intentions. The reason why emotions were suggested by Wright is that an important subset of emotional phenomena is closely connected with the interruption of a resource-limited control mechanism.

2.3.2 Behaviour decomposition

The need for behaviour decomposition has been illustrated before, by several examples. As the autonomous robot task becomes more complex it is usually necessary to introduce some form of hierarchy of behaviour which can simply consist in the decomposition of the task into a set of simpler skills, or behaviours. The re-combination of these behaviours is not straightforward and there are various methods used. To start with, the behaviours can be combined in parallel or in sequence. The behaviours can run simultaneously and produce influences on each other and different outputs. Or the behaviours can take control of the final overall agent behaviour one by one. An example for each of these approaches are respectively the architectures by Brooks and Malcolm described previously.

The selection of whether the behaviours should run in parallel or in sequence depends mostly on the behaviour specification. If the behaviour is self-sufficient and requires total control over the robot's actuators in order to fulfill its purposes correctly then behaviour composition should be sequential. Architectures that rely on the emergence of complex overall behaviour from simpler component behaviours may require parallel behaviour composition.

As discussed previously, the design of these architectures composed of very simple components is not easy. The sequential composition of behaviours is not easy either and it is particularly difficult to learn[3]. Simple selection rules based on sensory input are usually not enough. Some examples of the problems found with this approach and the solutions proposed in the domain of non-learning architectures are presented next. The problem most often reported is the need for the behaviour selection to have more persistence in time than that given by a reactive coordination of behaviours.

The need to artificially add persistence to the currently active behaviour to avoid dithering between behaviours is reported by Blumberg (1994). However, the mechanism used there also ensures that opportunistic behaviour can take place and that long-running behaviours are terminated by fatigue.

To solve the same problems, Correia and Steiger-Garção (1995) and Correia (1995)

[3] See Section 2.4 on learning for examples.

suggest an architecture where the behaviours themselves determine their level of activation which acts as a priority value for their selection. This activation level depends not only on the sensory input, but temporal rules of activation. These levels of activation are then taken into consideration by a structure of arbitration composed of simple blockers. The blockers endow currently selected behaviours with a small selection advantage that restrains behaviours with similar activations from being selected.

2.3.3 Generalisation *vs.* specialisation

The all-purpose robot has always been a human dream aimed at solving all our problems. In the design of autonomous robots such a dream is sometimes considered as a condition, in the form of strong flexibility demands. Nevertheless, everything that we have very successfully made so far is highly specialised (*e.g.*, Boeing 747, vending-machines, washing-machines, vacuum-cleaners...). Evolution itself developed living beings highly specialised to their particular niche.

In the field of robotics, specialisation also seems to be a good answer to how to minimise our problems (Steels, 1994a). The robot Polly (Horswill, 1993) is a good example of a solution that, because of its specialisation, performs well and in a very cost-effective way as long as conditions are appropriate.

Instead of trying to create the perfect robot that can understand and overcome all the difficulties posed by the world, one should take the world as it is and turn the problems into advantages. Even the troublesome noise of real world sensors and actuators can be helpful, as was shown by genetic algorithms experiments (Cliff et al., 1992).

One can argue to some extent that an autonomous system should be very adaptable and able to face the unexpected. Nevertheless, autonomy should not be totally identified with this kind of adaptation because solutions to specific environments can be found that can be called autonomous. For example, there are species high in the hierarchy that seem to be very adaptable and yet are only able to learn very constrained generalisations about their environment. Their learning abilities are adapted specifically to the ecological constraints typical of their normal way of life. This kind of learning limitation is demonstrated in an experiment done with rats by Garcia and

Koelling (reported in McFarland, 1993, page 362), which showed that rats are able to associate the taste of food with sickness but not with electric shock and are able to associate visual and auditory stimuli with shock but not with sickness. Several examples are also given by Gallistel et al. (1991) of how animals treat certain stimulus-reward, stimulus-response and/or stimulus-stimulus pairings as privileged. These provide persuasive arguments in favour of domain-specific determinants in animal learning, and for the authors' claim that, through evolution, the learning mechanisms of each species have been shaped by their specific problems.

2.3.4 Enactive approaches

The theory of autopoiesis is very extensive and it is not the goal of this dissertation to try to describe it in detail. Nevertheless, it is worth mentioning the stream of the cognitive sciences of today that Varela et al. (1991) consider to be the most realistic in terms of the theory of autopoiesis: enactment, which is analogous to the Artificial Intelligence stream commonly referred to as "behaviour-based". Varela et al. (1991) divide current research in Artificial Intelligence, Linguistics, Philosophy, Cognitive Psychology and Neuroscience into three main streams:

Cognitivism — Also named as the symbolic or computational approach is the stream that dominates present research. The central tool and guiding metaphor of cognitivism is the digital computer. Cognition is seen as the manipulation of symbols that are a mental representation of the environment.

Emergence — In this approach symbol processing is localised and only the physical form of the symbols is used. A representation is not a function of particular symbols, but consists in the correspondence between an emergent global state and properties of the world. It is also called connectionism, because the systems are made up of many simple components, which are connected by appropriate rules that give rise to a global behaviour corresponding to the desired task.

Enactive — This approach questions the centrality of the notion that cognition is fundamentally representation. More exactly, it questions the two following assumptions:

- the world has particular properties;
- individuals internally represent these properties;

> Cognition is not considered the representation of a pre-given world by a pre-given mind but is rather the enactment of a world and a mind on the basis of a history of the variety of actions that a being in the world performs.

The first approach described — Cognitivism — was the first to be seriously undertaken by a large research community, but its limitations became obvious with time. In this approach the information processing is sequential and localised. Therefore, it faces difficult problems of bottlenecks and robustness (Varela et al., 1991).

The second approach — Emergence — tries to work out these problems by having very simple and non-cognitive components with many connections to the other units. The global cooperation between units gives place to a global coherence without the need for a central unity to control the whole operation. The emergent approach also abandons the form and meaning distinction and associates meaning with the system's global state. This approach has had a few convincing results and has allowed us to shorten the distance between the study of biological and artificial beings.

In one way or another, both these approaches assume that there is a describable external world and that cognition is the representation of this world. However, our everyday experience reveals that the greatest ability of cognition is not to represent the world, but to distinguish in the great diversity of properties of the world those that are relevant. What counts as relevant is not pre-given, but is enacted or brought forth from the background by our common sense, in a contextual way.

Varela (1992) defends the notion that common sense cannot be packaged into knowledge by the discovery of more sophisticated rules. According to Varela, common sense is rather a *readiness-to-hand or know-how based on our lived experience* which entails an embodied history. Furthermore, Varela remarks that cognition cannot be properly understood without common sense, also referred to by Varela as the subject's bodily and social history, concluding that the knower and known stand in relation to each other in mutual specification.

The enactive approach (Varela, 1992) gives perception a fundamental role in terms of cognition. The author gives examples of perception of the world (*ibid.*) that show how, for example, colour and smell are always perceived in a contextual way that depends on the individual history of the knower. Thus, colour can only be understood as a visual experience of an embodied individual and in general cannot be identified with local surface spectral reflectance (Thompson et al., 1992). The colour perceived by the individual is often different from the colour that the physical properties of the light might lead to predict (Cytowic, 1993). The significance of the external signals depends on the individual who can share only similar experiences with individuals of the same species. Perception is an integrated part of the individual's interaction with the external world.

According to this theory, learning is the transformation through experience of the behaviour of an individual in a manner that is directly or indirectly subservient to the maintenance of its autopoiesis. What the observer calls memory is not a process through which the individual confronts each new experience with a stored representation before making a decision, but the expression of a modified system capable of synthesising a new behaviour relevant to its present state of activity.

The process of cognition then does not consist of the apprehension of the description of an independent universe, but is the result of a certain internal correlation that is being maintained between a sensory system capable of admitting certain perturbations and a motor system capable of generating movement. The nervous system plays an important role in cognition, because it expands the realm of possible states of the individual and enhances the organism associations with interactions with many different internal states.

2.3.5 Perception

It is often said that one of the essential abilities of autonomous agents, and animals in particular, is their ability to make sense of their input streams by recognising what is relevant. In nature, even the sensors themselves are selected during the genetic evolution of the species and, on a smaller scale, during the ontogeny of the individual living beings themselves. This allows for the selection of those sensors that make the discriminations important for survival. This ability endows animals with semantic adaptiveness (Cariani, 1992b), *i.e.* the ability to modify the relationship between their internal state and the external world in order to enhance survival. In particular, they can develop new sensory distinctions when in presence of ill-defined real world problems (Cariani, 1992b).

As far as living beings are concerned, this is not very difficult, because they are autopoietic (Maturana and Varela, 1980). The fact that they are producer and product in one allows them much freedom of choice. In robots such freedom is not possible with current technology: the most the agent can do is to calibrate the sensors it has available.

Active exploration of the environment can be considered an essential ability for autonomous systems which must operate in rich unstructured environments. Passively accepting measurements of the world is often not acceptable, because it only produces incomplete data from which only inferences full of uncertainties can be reached (Whaite and Ferrie, 1993). However, a model of uncertainty can be effectively used to direct perception to maximise knowledge acquisition (*e.g.*, Whaite and Ferrie, 1993). The work reported by Scheier and Pfeifer (1995) is one example of active perception where the agent manages to solve the perceptual aliasing problem[4] by active exploration. Another example of active perception is the work of Walker et al. (1998) in which ear movement is used to enhance the extraction of auditory cues for target localisation.

Perception should not be an end in itself; behaviours or the current intentional state of the agent should be responsible for determining what perceptual information is necessary at any one time. In the field of vision, animate vision (Ballard, 1991) showed how computation can be enormously reduced, and often be done in real time, if vision does not try to extract a three-dimensional representation of the world but operates in the context of behaviour. In particular, perceptual "objects" should be emergent entities of the agent's interaction with the environment and should not be confused with the objects that exist in the environment independently of the agent (Stewart, 1995).

[4] Derives from the fact that the sensory input of the same object can vary a lot dependent on ambiance conditions, distance, orientation, etc.

2.4 Learning and Adaptation

2.4.1 Introduction

There are three basic adaptation mechanisms available in natural systems (see for example Baldwin, 1896):

Phylogeny — the adaptation from generation to generation that results from natural selection through evolution;

Ontogeny — the adaptation provided by the system's learning during its life;

Heredity — The adaptation transmitted between individuals; social heredity, in particular, allows adaptation capabilities to be secured by the use of imitation from generation to generation.

Although heredity is often left out of development theories it also plays a very important role in the preservation of significant adaptive traits. Imitation, in particular, is very important for the transmission of complex behaviour as it has a clear advantage over learning by trial and error for the learning of complex sequences of actions. It can also have useful practical applications in the domain of robotics, by allowing the robot to learn its task by demonstration instead of being programmed by detailed instructions (Demiris et al., 1997; Kuniyoshi et al., 1994). However, in the domain of autonomous agents, imitation should have a critical element associated with it, *i.e.* the agent should be able to assess the intrinsic value of what it imitates in terms of its internal motivations.

Another important social factor is the help provided by the parents in guiding their children's learning through fruitful experiences (Rutkowska, 1995). Some researchers use similar techniques in robot learning (Lin, 1993) by giving their robots examples of how they can correctly accomplish their task. Other robotic researchers developed training methods where a human tutor provides frequent rewards or punishments to the robot in order to make the learning task easier (Nehmzow, 1994; Dorigo and Colombetti, 1993), a technique designated as robot shaping (Dorigo and Colombetti, 1993) by its analogy with shaping experiments in animals.

Nowadays, it is trendy to use evolutionary techniques in robotics, although their application to robots is objectionable to some (Mataric and Cliff, 1996), mostly due to the time consuming nature of the experiments. This is aggravated in autonomous robots research where on-line adaptation is a requirement (Winograd and Flores, 1986). The use of evolutionary techniques requires numerous evaluations of different adaptive agents each requiring a significant amount of time for a proper evaluation of their adaptation capabilities. In practice, waiting for the emergence of truly autonomous agents by evolution may require infinite patience (Toda, 1994).

Evolutionary techniques are suitable for solving problems where the fitness of any particular solution can be assessed efficiently. This does not mean that they are totally inadequate for autonomous agent research, but implies that they should not be expected to find the solution to autonomous behaviour from scratch.

Evolutionary techniques can be useful for testing alternative learning techniques or exploring the value of different learning parameters. There are several examples of the use of genetic algorithms for this purpose in different domains. Floreano and Mondada (1996) describe experiments where both phylogeny and ontogeny are used simultaneously in robots endowed with associative learning. Almássy and Verschure (1992) report the evolution of parameters in the domain of a model of classical conditioning. Kitano (1995) reports the genetic evolution of a genetic reaction-network and an evaluation network. The latter tries to model the role of hormones in learning and provides both a reinforcement function for on-line policy acquisition and a focus of attention to discriminate the more important features of the environment. The detection of relevant features during on-line learning leads to an increase in the learning rate and the number of mental rehearsals. The experimental results (*ibid.*) showed that the use of both reinforcement and focus contributed to the agent's adaptation. However, Kitano states that the slow learning of these agents would lead them to extinction if they were to co-evolve with purely reactive agents.

Another alternative to the use of genetic algorithms in autonomous robots research is their use in the learning algorithm itself. An example are the robotic applications (Reeke, Jr. and Sporns, 1993) of Edelman's neuronal group selection theory which states that neurons compete for survival as the embryonic brain is developing and that, only after birth, amplification and attenuation of synaptic connections strengths between neurons takes primacy. Other examples include the use of genetic algorithms in a schema-based architecture as a basis for unsupervised learning (Ram et al., 1994) and their application to classifier systems (Patel and Schnepf, 1992; Dorigo, 1995).

Furthermore, the interaction between learning and evolution can be exploited in the engineering of autonomous robots (Floreano and Urzelai, 1998). Apart from evolution being a powerful mechanism to select the most helpful learning mechanisms, learning during life can also help evolution to select the most adaptive traits. The individuals that are closer to the optimal solution are also the ones that will reach that solution faster through learning. This way, learning helps to discriminate the individuals which are closer to the solution, even when being near the solution, by itself, does not increase the measure of fitness of the individual. The fact that these individuals are the ones preferred also implies that there will be a gradual genetic assimilation of the features learned during life, even though learning does not have the capacity to directly modify the genotype. This indirect genetic assimilation of learned traits, defined as Baldwin's effect (Baldwin, 1896), which can lead to faster and more efficient evolution, has been supported by scientific evidence (Floreano and Urzelai, 1998).

Learning is also important to provide adaptation for local and relatively fast environmental changes that cannot be captured by the evolution process (Nolfi and Parisi, 1996).

Some learning techniques were mentioned in Section 2.3.1 in the context of specific architectures but, in fact, there are many architectures specifically designed for learning. Reinforcement-learning is the most common technique for learning in the domain of robotics and is therefore treated separately in the next section.

There are other learning techniques, like for instance the ones based on classical conditioning as defined by Pavlov (1927). An example of this is the robotic application of an unsupervised learning mechanism reported by Verschure et al. (1992). Their approach does not use external reinforcement and relies solely on the *a priori* stimulus-response associations. Learning here consists of the association of new stimuli, called the conditioned stimulus, with the behavioural responses to other, unconditioned, stimuli which occur simultaneously. One disadvantage of this approach is that it requires a high degree of architectural complexity in terms of the number of connections (Grossberg, 1971), because it imposes direct connections between all the unconditioned and conditioned stimuli and this can become very expensive with a high number of stimuli. Another problem with this approach is that learning cannot be explained only in terms of stimulus substitution (Mowrer, 1960), learning also involves substitution of behaviours which become inappropriate — a feature that is not modelled in classical conditioning. The two factor position defended by Mowrer overcomes the dichotomy between stimuli association (sign learning) and behaviour substitution (solution learning) by attributing a fundamental role to emotions. According to his view, stimuli are primarily associated with emotions which then drive the behaviour associations. This view substantiates the reinforcement learning approach if emotions are used as reinforcement.

2.4.2 Reinforcement learning

The short review that follows is not supposed to be an exhaustive survey (*e.g.*, Sutton and Barto, 1998; Kaelbling et al., 1996, offer more complete surveys) and comprises only a few examples.

Reinforcement-learning is a technique that allows an agent to adapt to its environment through the development of a policy, which determines which action it should take in each environmental state in order to maximise reinforcement. Depending on whether the reinforcement is computed internally or attributed by an external entity this can be considered unsupervised learning or not.

Reinforcement defines the desirability of a state and can be expressed both in terms of rewards and punishments. These are usually formalised in terms of the positive and negative values, respectively, of a reinforcement function that attributes a value to each learning iteration. This value can also be zero meaning that no reward or punishment was attributed and that evaluation is neutral.

The *a priori* domain knowledge incorporated by the designer in the learning system is minimal and is mostly encapsulated in the reinforcement function. This can be a limitation as some tasks might be difficult to describe in terms of rewards and punishments. The design of a reward function in robotic domains can be a problem when there are multiple goals and immediate reinforcement is not always available. In this case, it is often impossible to have a direct translation to a traditional monolithic reward function (Mataric, 1994).

An alternative reinforcement learning mechanism proposed by Bozinovski (1982) takes its inspiration from emotions. It starts with *a priori* associations of pleasant and/or unpleasant emotional states with specific context states. These emotional states will then be propagated through the rest of the robot state space while the robot explores its environment by trying out the different actions available. On account of being equipped with an initial rudimentary policy, presumably provided by genetics, the agent does not need any extra reinforcement and its learning will rely solely on reward propagation.

In opposition to other techniques (*e.g.*, Maes and Brooks, 1990; Maes, 1992; Nehmzow, 1994) reinforcement learning assumes the existence of delayed reinforcement. The reinforcement can be the consequence of a sequence of actions instead of a single action. This is important if the robot has to perform elaborate behaviour and possibly receive negative reinforcement in the course of achieving its task, because otherwise the robot will not have the necessary look-ahead to overcome the deterrents that it finds in the way of accomplishing its task. This means that reinforcement-learning algorithms usually have some form of credit assignment propagation so that value can be attributed to the states that lead to the goal state which produces reward.

The reinforcement learning algorithms in general are usually restricted to Markov decision processes, *i.e.* they assume that each environmental state can be entirely identified by the input representation defined by the designer as they do not explicitly deal with hidden state. An example of the hidden state is to have to decide upon the contents of a closed box without any input other than the vision of the closed box itself. If, for instance, the fact that the previous action had been to put a specific object inside the box was taken as a discriminatory element, then the hidden state problem would disappear. Hidden state is a problem which is usually present in robotic applications, because robots in general have very limited sensory capabilities making the differentiation between distinct states difficult.

Another issue present in the case of autonomous robot applications is that unlike traditional reinforcement learning tasks, the task of an autonomous robot is mainly one of continuously executing a task for as long as necessary in opposition to successfully completing a task and finishing. The goal-oriented nature of the problems usually used to test reinforcement learning techniques is not really applicable to autonomous agents. The autonomous robot should have multiple homeostatic goals that have to be prioritised according to circumstances and should not simply finish when it reaches a goal state. Another major difference in reinforcement learning applied to autonomous agents is that the distinction between a learning phase and a performing phase has to be eliminated, because an autonomous robot is supposed to continuously adapt to its environment.

Q-learning (Watkins, 1989) is the usually preferred reinforcement-learning technique because it provides good experimental results in terms of learning speed.

Although reinforcement-learning agents can be quite reactive and decide in real time the next action to take, their learning is quite slow particularly if the task is very complex. Slowness is usually pointed out as the major problem of reinforcement-learning techniques and is a particularly serious problem in robot domains where the life expectancy of the robot is usually short.

For more complex tasks skill decomposition is advisable as it can reduce significantly the learning time or even making the task feasible. Researchers report that a monolithic approach can fail to solve the long-term temporal credit assignment (Mahadevan and Connell, 1992). One of the reasons pointed out is the loss in accuracy of the propagation of credit assignment with long action sequences (Lin, 1992).

By task decomposition, the robot can learn behaviours that tackle each task individually and then learn the high-level coordination of the behavioural solutions found (examples in Lin, 1993). This requires the introduction of domain specific knowledge that might not be very easy to obtain and might limit the robot's final performance. Furthermore, task decomposition is a non-trivial problem, namely designing the subtasks' reinforcement functions may be hard (Mahadevan and Connell, 1992).

Behavioural decomposition usually consists of learning some predefined behaviours in a first phase and then finding the high-level coordination of these behaviours. Although the behaviours themselves are often learned successfully (Mahadevan and Connell, 1992; Lin, 1993), behaviour coordination is much more difficult and is usually hardwired to some extent (Mahadevan and Connell, 1992; Lin, 1993; Mataric, 1994).

One problem in particular which is quite difficult and task dependent is deciding when to change behaviour. This is not a problem in traditional reinforcement learning where agents live in grid worlds and state transition is perfectly determined. However, in robotics, agent states change asynchronously in response to internal and external events and actions take variable amounts of time to execute (Mataric, 1994). As a solution to this problem, some researchers extend the duration of the current action according to some domain specific conditions of goal achievement or applicability of the action. Others will interrupt the action when there is a change in the input state (Rodriguez and Muller, 1995; Asada, 1996). Rodriguez and Muller (1995) argue that new decisions should only be taken when there is a change in the input state, on the basis that otherwise the choice is uniquely determined by the current state of knowledge. However, this may not be a very straightforward solution when the robot is equipped with multiple continuous sensors that are vulnerable to noise.

Generalisation over the input space can also be a useful technique to accelerate the learning process (Lin, 1993). One of the major problems responsible for the slowness of reinforcement learning is the slow iterative process of spreading the rewards and punishments through the input space, which can be greatly minimised if the algorithm has added mechanisms to spread the reinforcements to similar input states.

One solution is to use neural networks to learn the utility values of each action (Lin, 1993). This way similar inputs are automatically updated when the network is being trained for the current input. Apart from accelerating the learning process, it also minimises the memory space needed to store the policy, which is often stored in the form of a look-up table with one value for each action and sensor state combination. This system, which has still other methods to overcome the slowness of the learning process, is described in detail in the next section as it is very similar to the one used in the experimental work reported in this thesis.

Another reinforcement learning algorithm using neural networks is the complementary reinforcement back-propagation algorithm (Ackley and Littman, 1990). In this case the output units of the networks encode the value of the action to be chosen in binary format. The values of the bits that represent the action to be taken are determined probabilistically from the activation value of each of the output nodes of the network. There are several examples of robotic application of this algorithm (Meeden et al., 1993; Kitano, 1995). This approach has an advantage over the one proposed by Lin (1993) of accommodating a greater number of possible actions with equivalent neural networks.

A different solution to the generalisation of input problem consists of the use of a Kohonen network to build a self-organising map of the sensory domain by exploration (Kröse and Eecen, 1994). In this approach, the neighbourhood relations between different states were imposed by the elementary actions.

The G Algorithm is yet another solution to the generalisation of input problem (Chapman and Kaelbling, 1991). The Q-table is represented by a tree that ramifies on the binary inputs. The tree is constructed as the agent explores its environment and groups the input space according to reinforcement. A split of the input space is made whenever a bit of the input space is determined to be statistically relevant in terms of the immediate reinforcement or discounted future reinforcement. The tree must be constructed before the learning action value phase. Another example of a tree-based algorithm (McCallum, 1996) addresses both the problems of input generalisation and hidden state, by adding information about previous states when there is a need to discriminate between otherwise indistinguishable input states.

One of the disadvantages of Q-learning is that it usually reduces the number of actions available to the agent to a small set. Having such a small and non-continuous space of possible actions is not very satisfactory in terms of achieving robotic autonomy. However, more important than allowing the robot more freedom of movement is to allow the robot freedom of choice. In reinforcement learning, the action the robot takes at each point is not predefined by the designer, but is selected by the robot according to what it has learned so far. Furthermore, if the set of actions is varied enough, the robot can still have a large repertoire of different behaviours by sequentially selecting the appropriate actions at each time step.

Some researchers have found that, in complex domains, straightforward reinforcement learning converges to local minima instead of learning a good policy (Lin, 1991; Chapman and Kaelbling, 1990). This problem can be overcome in part by an external teacher that shows the robot how to obtain reward (*e.g.*, Lin, 1993), by indicating the relevant actions that can be taken in the environment or forcing it out of local minima.

The problem of local minima is strongly influenced by the exploration *vs.* exploitation strategy selected. When learning, the agent has to trade-off between acting to get more information about the world and acting on the information it already has to get more reinforcement. If the agent does not actively explore its environment then it can easily become stuck in local minima.

The simplest solution to the exploration *vs.* exploitation problem is to use the ϵ-greedy strategy (defined in Sutton and Barto, 1998). This consists in taking the best ranked action most of the time and making a random action selection with a small probability ϵ. Usually, ϵ takes the value of 0.1. A more reasonable approach to the problem is to keep track of how much knowledge the agent has gathered in each context so that it can select to explore sub-optimal behaviours only in the situations where it has not tried them before. An example is the interval estimation algorithm (Kaelbling, 1990), that explores only if it has insufficient information.

The evaluation of the performance of the reinforcement-learning controllers can also be a difficult problem (Wyatt et al., 1998). Researchers are often tempted to use the reinforcement received by the controller to analyse how the performance of the agent is improving in time. In fact, an increase of reinforcement value is usually associated with an improvement of performance of the agent in its task. However, exceptions can be found even for well-designed reinforcement functions. Usually, the reinforcement function is internally computed by the agent which means that it is subject to the limitations of its perception. Inaccurate perception can make the reinforcement function misleading in the evaluation of the robot's true performance level (Wyatt et al., 1998). Nevertheless, an evaluation provided by an external observer is typically correct and can be useful to point out possible deficiencies of the internal evaluation. Moreover, because there is often a stochastic element associated with the learning process — introduced by the exploration algorithm, for example — a one-trial test of the controller can be misleading by showing a one-off performance instead of the expected performance of the learning algorithm.

2.5 Selected Architecture

The basic learning controllers used in the experiments reported in this dissertation are very similar to the learning architecture proposed by Lin (1993). This architecture is the main topic of the current section. To begin with, a short description of Lin's architecture is given. This is followed by a short discussion of its advantages and disadvantages. Finally, the domain-specific mechanisms of this architecture, which will be filled in later in this dissertation, are highlighted.

2.5.1 Description

A sketch of the architecture presented in Lin (1993) is shown in Figure 2.1. As was stated before, the main feature that characterises this architecture is that it solves the input generalisation problem by using neural networks to learn the utility function, one network per action.

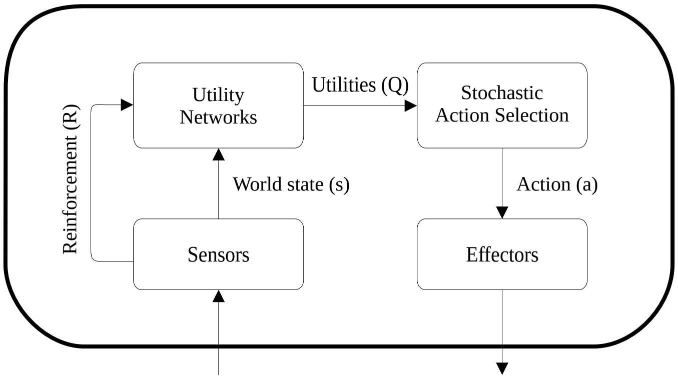


Figure 2.1: Lin's learning architecture.

This approach employs feed-forward neural networks with one hidden layer that use the following symmetrical activation function (a scaled hyperbolic tangent):

$$S(x) = \frac{1}{1+e^{-x}} - 0.5 \tag{2.1}$$

For training the neural networks it uses the back-propagation algorithm with a learning rate of 0.3 and a momentum of 0.9. Although this is not a very good network training method when compared with other batch-oriented training methods, it allows the incremental learning required for on-line learning.

The approach is based on the Q-learning algorithm for policy acquisition. The input of the neural-networks consists of the world state and the single output of each neural-network models the following function for one of the actions a:

$$\text{util}(s_n, a) = R_{n+1} + \gamma \ \text{eval}(s_{n+1}) \tag{2.2}$$

This function represents the expected discounted cumulative reinforcement that an agent will receive after executing action a in response to the world state s_n. The immediate reinforcement received in the next state (s_{n+1}) is R_{n+1}. The utility of the state s_{n+1}, or eval(s_{n+1}), is its expected discounted cumulative reinforcement if the optimal policy is followed by the agent. The value γ is the discount factor which is set to 0.9.

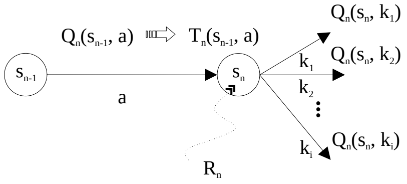


Figure 2.2: Learning iteration of the reinforcement-learning algorithm.

In each learning step of the algorithm (see Figure 2.2), the neural-network associated with the last action a taken is updated for the previous state s_{n-1}. An iteration of the back-propagation algorithm is made using as target value $T_n(s_{n-1}, a)$. The calculation of this target value depends on the current estimative $Q_n(s_n, k)$ for each action k provided by the outputs of the networks when using as input the current state (s_n).

$$T_n(s_{n-1}, a) = R_n + \gamma \ \max\{Q_n(s_n, k) \mid k \in \text{actions}\} \tag{2.3}$$

For action selection, Lin uses probabilistic action selection based on the Boltzmann-Gibbs distribution. The probability of selecting action a, with temperature T is:

$$P_n(s_n, a) = \frac{e^{\frac{Q_n(s_n,a)}{T}}}{\sum\limits_{k \in \text{actions}} e^{\frac{Q_n(s_n,k)}{T}}} \tag{2.4}$$

The temperature value is increased if the robot is within the same small area for a long period of time and reduced to zero in the testing phases, *i.e.* during the tests the action with the highest utility value is always selected.

Lin (1993) proposes three other different neural-network-based solutions that provide the robot with some memory to deal with hidden state. The solutions use recurrent networks or time windows as the input of normal networks. He also uses some extra techniques to enhance and accelerate the learning process:

Experience replay — replay a sequence of experiences in temporally backward order to speed up the credit assignment problem.

Action Model — have a model of the world that permits the agent to experience the consequence of its actions without having to try them out in the real world.

Teaching — Guide the robot through significant exploration.

These different network-based solutions and extra techniques were not used in the work carried out for this thesis.

In the experiments reported by Lin (1993) the robot's task is decomposed into three simple behaviours: wall-following, going through doors and docking on the charger. For learning these behaviours, the learning algorithm has sixteen different available actions to chose from and twenty-four sonar and light sensors defined as network inputs. The behaviours are learned separately with success. The learning is done in simulation, but Lin (1991) reports only small drops in performance when the controllers are tested directly in a real robot without any further adaptation.

In a second stage, after the simple behaviours have been learned, Lin uses the same learning algorithm to learn the coordination of these behaviours, *i.e.* he considers the behaviours themselves as the system's available actions. However, in order to learn the behaviour coordination successfully, he has to introduce simplifications into the learning task. To start with, each behaviour is associated with pre-defined conditions of activation. For example, the behaviour of door-passing can only be selected if a door is nearby. Furthermore, the introduction of a persistence rule proved essential for good results. This rule ensures that the same behaviour is kept until the goal of the behaviour has been achieved or a previously inapplicable behaviour becomes applicable.

2.5.2 Pros and cons

Since the focus of this research is emotions' influence in control and not control itself, in the selection of the learning architecture the simpler techniques that have been proven successful in the past were preferred. This philosophy was carried out even in the selection of the architecture's various arbitrary parameters that were chosen without much regard for optimal behaviour. For this reason the selection is far from perfect and presents several disadvantages:

- The learning abilities of the robot are not sufficiently sophisticated to allow great degrees of autonomy. Namely, the agent is essentially reactive and cannot deal with hidden states. Furthermore, a high degree of autonomy would probably require some form of autonomous decomposition of behaviour.

- The solution for dealing with generalisation over the input state has some problems. The neural networks have a tendency to be overwhelmed by the large quantity of training data provided by on-line learning and forget the rare relevant experiences. Filtering the available data in such a way that relevant training data has more weight in the learning process can help to prevent this problem.

- The agent has available only a restricted number of discrete actions, which may limit its behavioural capabilities. In this architecture, the selection of more or less general actions and inputs determines how specialised the agent is.

- The exploration *vs.* exploitation solution is very simple, yet, it has some advantages. First, it does not force the division between a learning phase and a performance phase, which is obviously undesirable for autonomous learning, because it allows more or less exploration dependent on the previous differences in performance registered for the different actions. And secondly, it does not implicitly assume an optimal action at each point, allowing for a more flexible policy.

- The fact that the policy acquisition is indistinguishable from world modelling has the disadvantage to require new policy acquisition from scratch every time the goal of the agent is changed.
- The navigation abilities provided by the architecture are poor because the agent does not have any notion of its location in space.

In traditional experiments with reinforcement-learning architectures the agent does not have homeostatic goals as required for autonomous agents, yet no problems were found in using this architecture in the pursuit of that kind of goal.

The positive points of the selected architecture are that it endows the agent with fast reactions, while still allowing it to learn its policy to act in the environment. This policy can be quite flexible as long as an adequate action set is chosen. The generalisation over the input state allows a greater richness of sensory input than usual and provides an acceleration of the learning process.

2.5.3 Open specifications

The selection of the learning architecture described previously left some domain-dependent details unspecified:

- reinforcement function;
- action set;
- state input;
- state transition;
- Meta-control variable values:
  - back-propagation learning rate;
  - action selection temperature.

In the design of the final autonomous agent all these open specifications must be filled in *a priori*. This does not mean that the specifications have to be rigid, but simply that the agent should not receive further external assistance once it starts its learning task. For example, the learning parameters should not be changed by an external entity after the learning process has started.

The reinforcement function must specify the correct behaviour of the agent by giving it rewards when it is performing well and punishments when its behaviour is inadequate. For this reason it implicitly specifies the agents' goals or motivations or its task. In autonomous agents, the selection of the reinforcement must take into consideration the fact that these implicit goals should be of a homeostatic nature.

The definition of the input space implicitly informs the robot which elements of its environment are important for achieving its task. The generalisation mechanisms provided by the neural networks allow the agent to discriminate which inputs are more adequate for the selection of its behaviour, but ultimately the designer has to define correctly all the possible inputs the agent might need.

Unfortunately the output space is reduced to a finite number of actions. This means that the action set should be selected with care, allowing enough flexibility of movement, namely in giving the robot enough freedom of movement to perform its task correctly. The actions can either be very primitive or consist of more elaborate behaviours. This difference in control strategy is actually what differentiates the experimental Chapters 4 and 5 from each other.

The state transitions are very important because they specify when the agent should evaluate the previously selected action and select a new one. A state transition is usually defined by a change of input state which simply consists of a change of the input values when discrete input is used. If continuous and noisy sensors are used to define the input space then the definition of state transition is not so simple. The problem becomes more difficult if behaviours are used instead of actions making a state transition at every step inappropriate. The definition of the state transition in this case is often associated with domain-specific conditions, yet this seems to represent an unnecessary stipulation of arbitrary rules that restrain the agent's flexibility. A solution based on determining significant changes in the input state seems more adequate. The problem of state-transition determination is closely related to the difficulties in determining when to change behaviour found in sequential behaviour decomposition. The duration of behaviours must be long enough to allow them to manifest themselves, and short enough so that they do not become inappropriate (due to changing circumstances) long before being interrupted.

The learning parameters mentioned above are often changed during learning to enhance the agent's abilities. This can be useful but should not be done by an external agent. In particular, the learning and exploration should not be stopped by an external entity that decides when the agent is sufficiently competent at its task. The agent itself should determine the value of these parameters as far as possible.

All these specifications left open by the learning architecture invoke the need for some kind of motivational system that can exert some form of meta-control over the learning algorithm. Later in this thesis, it is discussed and empirically explored how emotions can fulfill this role. For instance, emotions are usually associated with reinforcement and will therefore be used as its main source during the experiments. Furthermore, emotions can help to define the occurrence of state transitions when behaviours are used as the elementary actions of the learning architecture. Finally, emotions can also be used for varying on-line the value of the meta-control variables.

Chapter 3

Natural and Artificial Emotions

3.1 Introduction

In their quest for true intelligence, people usually have a Cartesian approach that regards emotions as a hindrance carried over from their early evolutionary development, at odds with their aspiration to high rationality. Psychologists, too, tend to concur with this popular view of emotions as useless or even disruptive to rationality (Toda, 1993). Interestingly, such natural distrust towards emotions can be substantiated if we take the opposite view, *i.e.* that emotions are indeed central to reasoning. Several reasons for the disruption caused by emotions, which are a direct consequence of considering emotions essential, are pointed out throughout this chapter. In fact, this view that emotions are an integral part of rational behaviour is receiving increasing support from brain research studies (LeDoux, 1998; Damásio, 1994; Cytowic, 1993).

Emotions play an important role in our lives, influencing our every day life decisions. As Goleman (1995) defends, the power of the emotional mind in everyday decision-making is greater than the power of the rational mind. He convincingly argues that it is more advantageous for success in life for humans to have a good emotional development than a high intelligence quotient.

Studies show that human decisions are not always rational (Grossberg and Gutowski, 1987). Pure logic is not enough and shows serious faults when used to model human intelligence in Artificial Intelligence systems (Dreyfus, 1992). Furthermore, emotions have been suggested in the field of Artificial Intelligence as the ultimate source of intelligence that might provide robots with the autonomy they need (Toda, 1994). Doubts have even been posed on whether machines can exhibit intelligent behaviour without emotions (Minsky, 1986; Charland, 1995).

Next in this chapter, several views of the influence of emotions in cognition are presented in terms of different cognitive mechanisms like memory, attention and reasoning. This will be followed by a description of the different emotions' functionalities in natural systems that can be and have been transfered to the artificial systems' domain. The chapter will finish with the proposal and analysis of the emotion model that was used in the experimental work carried out for this thesis.

3.2 Natural Emotions

3.2.1 Emotions and memory

One of the basic reasoning processes that is influenced by emotions is memory (see Blaney, 1986, for an extensive review on the subject). Blaney (1986) presents two alternative ways in which emotions can influence memory:

Mood dependence[1] — What one remembers during a given mood is determined in part by what one learned previously in that mood. Affective valence of the material, *i.e.* the type of emotions associated with material itself, is irrelevant.

Mood congruence — Some material, by virtue of its affective content, is more likely to be stored and/or recalled when one is a congruent mood. Concordance between mood at exposure and at recall is not required or relevant.

Others (Schwartz and Reisberg, 1991) claim that emotions independent of valence can contribute to better memorisation, *i.e.* that the events that are the most vividly remembered are also the most emotional or even traumatic, while emotionally neutral events are easily forgotten. Studies of the human brain support the view that emotions might be responsible for enhancing memorisation (LeDoux, 1998).

[1] State dependence in the original work.

In his experiments, Bower (1981) showed mood-dependence effects and claims that through their influence on memory, emotions have the power of biasing our decisions. Depending on their current mood, people are more likely to recall events that are congruent with that mood. These will make the probabilities of possible outcomes for each choice available at any one time subjective. Under these conditions, it is clear that the combination of the utilities of prospective outcomes and their probabilities for each choice will not lead to the selection of an objective optimal choice. When people are happy they are also more optimistic, because they raise the estimate of positive future events and reduce estimate of negative future events (Bower and Cohen, 1982). If, on the contrary, people are sad they will selectively remember more negative events which in extreme cases, can contribute to the vicious cycle of deepening depression (Blaney, 1986).

Later experiments (Bower and Mayer, 1985) demonstrated that mood-dependence effects are unreliable phenomena in laboratory experiments and found mood-congruence effects instead. Nevertheless, it is usually believed that an event is remembered best when people are in a situation or state similar to the one when the learning took place (LeDoux, 1998).

3.2.2 Emotions and attention

The mood-dependent recall can also be seen as an adaptive trait that allows the individual to recall only events that occurred previously in similar contexts and not to be distracted by irrelevant information. In general this can be seen as advantageous.

Many emotions theorists agree that emotions are most helpful for focusing attention on the relevant features of the problem at hand (LeDoux, 1998; De Sousa, 1987; Tomkins, 1984; Plutchick, 1984; Scherer, 1984; Panksepp, 1982) and, in particular, for determining the salience of the perceptual information (Cytowic, 1993). However, this can also provide a way for certain emotions to disturb the thinking process, because it makes it more difficult to pay attention to all aspects of a complex problem. In particular, long-term consequences can be ignored (Loewenstein, 1996) which for a wild-life environment is adequate but can be a serious disadvantage in a highly organised technological society like our recent man-made society, where the repercussions of decisions can spread out over a wide space and are very slow in dying out (Toda, 1982).

Emotions are also often pointed to as essential mechanisms for autonomous agents with multiple goals and limited resources in uncertain environments (Oatley, 1987; Frijda and Swagerman, 1987; Moffat et al., 1993). Their role is associated with the process of interrupting the agent's ongoing activities to deal with new and unexpected situations that need to be attended to (Sloman and Croucher, 1981; Simon, 1967) while protecting the resource-limited activities from unnecessary interruption and computation (Wright, 1994). These interruptions are particularly important when urgency of response is essential for survival, but in extreme, mostly pathological, cases can also be disruptive when the generated interruptions become frequent and inappropriate or undesired.

Apart from switching attention away from the task at hand, emotions are also usually held responsible for bringing to conscious awareness the emotion-inducing event and preparing the motor system for a reaction (Ortony et al., 1988).

3.2.3 Emotions and reasoning

One of the simplest ways emotions are considered to influence reasoning is by providing an evaluation value of the subjects' situation. It is often assumed that human decision making consists of the maximization of positive emotions and minimisation of negative emotions (*e.g.*, Tomkins, 1984).

Recently the role of emotions has been enlarged. Some researchers have proposed that the human brain is divided in two major independent and interacting systems: an affective and a cognitive one (Zajonc et al., 1982; Damásio, 1994; LeDoux, 1998). Both systems are responsible for behaviour, and their intensive cooperation attributes a primary role to emotions in reasoning.

On the one hand, to the emotional mind is attributed the responsibility for the faster responses by providing the system with an efficient mechanism to spring into action without pausing to think and only attend to the most striking aspects of its perception. On the other hand, the cognitive system makes a more extensive and careful evaluation of the situation and might eventually decide to bring the emotional response to a stop (LeDoux, 1997).

Furthermore, recent neurophysiological research suggests that our thinking is not so detached and ungrounded as we might believe and that emotions also assist the cognitive system. According to this research, with the help of the emotions, the feelings provided by our body play an important role in reasoning. This is the central claim of the somatic-marker hypothesis[2] (Damásio, 1994).

Damásio makes a clear distinction between the concepts of emotion and feeling. Feeling designates the process of monitoring the body. Feelings offer us the cognition of our visceral and musculoskeletal state. Emotion is a combination of a mental evaluative process with dispositional responses to that process, mostly toward the body proper but also toward the brain itself. According to Damásio, all emotions generate feelings, but only some feelings generate emotions. If feelings are associated with emotions then the body signals will move from the background to the foreground of our attention.

Somatic markers are special instances of body feelings, generated by emotions, which are acquired by experience based on internal preference systems and external events and which help to predict future outcomes of certain scenarios. They will force attention on the negative or positive outcome of certain options that can be immediately defeated, leaving fewer alternatives, or can be immediately followed. Through the estimation of long term-costs and benefits, the somatic markers provide humans with a reasoning system that is free from many of the faults of formal logic, namely the need for much computational and memory power for having every option thoroughly evaluated.

Damásio provides compelling evidence for his hypothesis, by showing examples of how emotionally impaired people have major problems making decisions. However, the boldness of his hypothesis has also created some skepticism. Sloman (1998) claims that the evidence only shows that global central mechanisms are necessary to ensure that the more specific mechanisms are deployed correctly; and that these mechanisms used for redirecting attention, and therefore essential to intelligence, may also be necessary for emotion production. He goes on to say that Damásio's heuristic control can occur without emotional mechanisms, because problems with massive search can easily be solved by humans with a context-addressable memory of slightly generalised special

[2] Marker because it marks an internal mental image and somatic because it is marked through body feelings.

cases.

Recently, Damásio's group has produced further results (Bechara et al., 1997) that provide stronger support for the somatic-marker hypothesis. Their experiments show that people reach the right decisions before the cognitive system has access to the necessary data to make informed decisions. This suggests that there is an independent process which is quickly attributing value to each decision. Apart from providing biases that assist the reasoning system in a cooperative manner, the emotional system is also credited with generating the overt recall of the pertinent facts necessary for the cognitive evaluation (Bechara et al., 1997). This way the emotion system contributes to the efficiency of the decision process.

3.3 Artificial Emotions

In this thesis, the approach followed towards emotions is an engineering approach (Wehrle, 1998). The primary criterion is one of performance of the robot, more specifically the enhancement of its autonomy, and not to improve our knowledge about the nature of emotions themselves, although the effective use of emotions might hopefully contribute some clues to their understanding.

As such, the aim of this work will not be to try to replicate the experience of human emotions as reported by the individuals' subjective cognitive observations, but to try to capture the underlying mechanisms which have an adaptive value that can be transposed to artificial creatures. Some properties of emotions that might be useful to an autonomous artificial creature are:

- Source of motivation, where motivation means anything that controls the focus of attention and orients the current reasoning of the agent. Emotions have been considered a fundamental source of motivation in psychology (*e.g.*, Beck, 1983) and have been used as a source of motivation in artificial creatures (Morignot and Hayes-Roth, 1995).

- Control of attention. Emotions influence perception by focusing the agent's attention on the most relevant features to solve its immediate problem. In particular, they have been attributed the role of interrupting the agent from what it is doing when new problems arise that need to be attended to (Sloman et al., 1994; Beaudoin and Sloman, 1993, describe an application within a nurse-maid scenario).

- Source of reinforcement. Emotions are usually associated with either pleasant or unpleasant feelings that can act as reinforcement. This allows emotions to motivate the agent to approach or avoid certain emotional scenarios. This is the most usual role attributed to emotions in the functionality of an artificial agent[3] (*e.g.*, Wright, 1996; Albus, 1990, or McCauley and Franklin (1998) in the domain of Pandemonium Theory).

- Emotion dependent memory. Bower and Cohen (1982) proposed a blackboard control system to model mood dependency. In their system, the subject's mood when learning is associated with what is learned. Later, moods act as selective filters in the retrieval process, admitting retrieval of events stored in memory that were originally learned in moods that are congruent with the current mood. Mood-congruence effects have also been modelled (Araujo, 1994) but using a system composed by two independent but interactive neural-network subsystems, one cognitive and one affective. The reinforcement-learning system developed by El-Nasr et al. (1998) models emotion dependent recall by making the agent more or less optimistic when it is respectively more happy or sad.

- Assistance in reasoning. Based on the ideas of Damásio (1994), Ventura et al. (1998) propose an emotion-based agent that simultaneously processes stimuli by an affective and a cognitive system. The agent's affective system quickly attains perceptual images that are used to directly access the cognitive images relevant for the cognitive system's deliberation.

- Behaviour tendencies or even stereotyped responses are usually associated with particular emotional scenarios. These built-in responses allow for appropriate behaviour to be automatically triggered in emergency situations, avoiding spending

[3] There are also some researchers (*e.g.*, Michaud et al., 1996; Shibata et al., 1996) who give emotions the somehow more sophisticated role of monitoring the robot's performance so that the robot's plans or actions can be changed if necessary.

unavailable time on elaborate reasoning. A typical example is the fear emotion where the source of fear is quickly located and avoidance behaviour is immediately activated. In the architecture proposed by Botelho and Coelho (1997), emotions are associated with simple procedure responses whose execution directs the agent to the identification of the emotion's cause so that immediate action can be taken.

- Physiological arousal of the body. A strong emotion is usually associated with a general release of energy in anticipation of demanding action response. The importance of this feature in biological systems is clear: it provides the way to mobilise extra energy to cope with emergency situations in a complex chemical entity. However, the translation of this feature to an artificial system is not clear, because in general artificial systems are not endowed with different states for overall performance. Nevertheless emotions can be used to modulate simple system parameters (Cañamero, 1997; Bates et al., 1992a), *e.g.* level of behavioural activity or speed, that are directly relevant to the overall performance of the system.

There are many other properties left out, the more pertinent being those of a social nature, which were left out on purpose. It is clear that emotions play an essential role in social interaction. The expression of emotions allows the individuals to transmit to others messages that are often crucial to their survival and therefore have great adaptiveness value (Darwin, 1965). This is a very interesting dimension of emotions that has received some attention in Artificial Intelligence research. The expression of emotions can be useful in several domains. It can:

- Enable artificial creatures to generate empathy emotions in people, by creating an illusion of life necessary for believable characters (Bates, 1994). This is particularly important in entertainment oriented systems.

- Regulate the intensity of the interaction between a learning robot and a teacher (Breazeal, 1998). The robot's emotional reactions can provide cues to whether it is being over-stimulated or getting bored. Taking those into consideration, the teacher can maintain a suitable learning environment that will enhance the learning performance of the robot.

- Make artificial systems more responsive to human emotions and thus more user friendly by implementing mechanisms to recognise human emotional expressions. This can be advantageous in both entertainment and educational applications (see Picard, 1995, 1997, for numerous suggestions). An example is the simulation of empathy feelings in computer interfaces in order to help the relief of frustration in human users (Klein, 1996).

- Allow for new communication mechanisms between artificial creatures. For instance, Shibata et al. (1996) use emotions as a communication mechanism that allows the robot to report to others its internal state, or more specifically, its level of task achievement.

- Help establishing and securing commitments between social agents, so that artificial agents can benefit from interaction and cooperation with others without being totally open to exploitation by enemies and profiteers (Aubé, 1998a).

These ideas are beyond the scope of the work reported here and will not be explored further in the current work, which is concerned with more basic mechanisms of simple survival by a solitary agent.

Some people argue that emotions are mostly important in the realm of social interaction and that it is in this dimension that they serve a real purpose. Some go as far as to argue that only social emotions are truly emotions (Aubé, 1998b). And it is quite true that in humans emotions of a social nature are among the most numerous, complex and refined. Nevertheless, it is also true that the complexity of social interaction present in human societies is quite recent in an evolutionary time scale (Papez, 1937) and that basic emotional mechanisms and their brain structures are much older. This by itself suggests that there are some basic emotions on top of which social emotions develop. Instead of denying that those innate emotions *are* emotions, one can instead name them primary emotions (Damásio, 1994) in contradistinction to the more sophisticated secondary emotions. These primary emotions are usually associated with basic survival instincts. As such, they often look misplaced in highly structured and artificial human societies (Toda, 1982, 1993), where social emotions are often more useful.

Another issue studied in Artificial Intelligence is how emotions can help in the creative process. Artificial systems are typically very predictable, because they follow rigid sets of rules or commands that do to not leave much room for the generation of new spontaneous behaviour. This is annoying in entertainment applications, but it is particularly serious in applications where creativity is essential, for instance musical composition. Some solutions to this problem have been proposed that resort to emotions (*e.g.* Riecken, 1998). These rely on the fact that memory retrieval is a key activity for the free associations necessary in creative work, and that memory retrieval itself is largely dependent upon emotions. Even the automatic generation of musical performance can profit from emotions, by adding emotional expressiveness taken from the performance of other music by humans exhibiting the intended mood (Arcos et al., 1998).

3.4 Proposed Model

3.4.1 Presentation

A large subset of theories of emotions is based on elaborate cognitive appraisal theories (*e.g.*, Lazarus, 1982; Power and Dalgleish, 1997; Ortony et al., 1988) that stress the role of conscious reasoning in the generation and definition of emotions, in spite of emotions also being aroused by crude subconscious experiences involving simple information processing without the need for high level reasoning processes (Zajonc, 1984; Izard, 1993).

Following the psychologists' main stream, most Artificial Intelligence models of emotions are based on an analytic and symbolic approach (Sloman et al., 1994; Frijda and Swagerman, 1987; Dyer, 1987; Pfeifer, 1982; Pfeifer and Nicholas, 1985; Bates et al., 1992b) that tries to endow the model with the full complexity of human emotions as perceived from an observer's point of view. However, in both ontological development and evolution the full richness of emotions is only achieved at a final stage. In the early stages of these processes only certain basic emotions are present and, presumably later, other more complex emotions develop on top of these.

In opposition to the traditional approach, a synthetic bottom-up approach based on the animat approach (Wilson, 1991) was preferred for the current work. This made the existing models inadequate, because they are over-designed and too complex (Pfeifer, 1994), leaving no other alternative than designing yet another emotion model.

Recently, models have been suggested that also follow a bottom-up approach (Velásquez, 1998; Cañamero, 1997; Foliot and Michel, 1998; Wright, 1996) and it is interesting to see that they often agree with the present work in the treatment given to the most relevant issues. The problem with reproducing most of these models is that they usually provide so little architectural specification that they allow almost total freedom of implementation. Furthermore, the evaluation of their practical implementations is often difficult, because in general they are presented as an end result, *i.e.* the adaptiveness value of the presence of emotions is not evaluated, but only presented as fact. In these conditions, unless an objective and accurate description of the end product is given, only its direct observation can make any kind of evaluation possible.

The most significant emotion features that the designed model tries to capture are:

- Emotions have valence, *i.e.*, they provide a positive or negative hedonic value.
- Emotions have some persistence in time, *i.e.* sudden unrealistic swings between different emotions should not be allowed, particularly when the emotions in question differ a lot.
- The occurrence of a certain emotion depends not only on direct sensory input, but also on the agent's recent emotional history.
- Emotions colour perception in that what is perceived is biased by the current emotional state.
- Emotional state can be neutral or dominated by an emotion. This implies the existence of a mechanism to decide which emotion, if any, is dominant at any one time.

The model that was developed — Figure 3.1 — is based on four basic emotions ($\mathcal{E}$):

Happiness, Sadness, Fear and Anger. These emotions were selected because they are believed to be the most universally expressed emotions along with Disgust (Ekman, 1992) and are adequate and useful for the robot–environment interaction afforded by the experiments. Others might prove too sophisticated or out of place. For instance, there seems to be no situation where it is appropriate for the robot to feel disgust. However, if, for instance, toxic food were added to the environment, disgust would become useful to keep the robot away from it.

$$\mathcal{E} = \{\textsc{Happiness},\ \textsc{Sadness},\ \textsc{Fear},\ \textsc{Anger}\}^{4} \tag{3.1}$$

The emotions chosen are also usually included in the definitions of basic or primary emotions[5] (see, for example Shaver et al., 1987; Power and Dalgleish, 1997; Goleman, 1995), which is a good indicator of their relevance and need. Other emotions, like love and hate, which some authors like to suggest as primary emotions, were not included because they do not seem very basic[6] and the present work does not have, for the moment, any social aims.

The model determines the intensity of each emotion based on the robot's current internal feelings ($\mathcal{F}$). The intensity of each emotion is calculated through simple linear weighted dependencies from feelings. The nature of the feelings depends on the robot and its task, but might, for example, include Hunger, Pain and Temperature. The set used in the first experiments is given in Equation 3.2.

$$\mathcal{F} = \{\text{Hunger, Pain, Restlessness, Temperature, Eating}\} \tag{3.2}$$

Furthermore, the emotion state also influences the robot's feelings, or body state. The body reactions that give rise to an emotion are also the ones aroused by the emotion. This way, each emotion tries to influence the body state in such a way that the resulting

[4] A different typeface is used for the model's emotions to distinguish them from natural emotions.

[5] It should be noted that the paradigm of primary emotions is not undisputed, yet most of the arguments against it are marginal to the present usage. These arguments refer to the plausibility of translating all emotions in terms of graduations of primary emotions. The point here is that these emotions are more universal and fundamental than others and therefore more adequate to animats with low reasoning capabilities in a simplified environment.

[6] There are even arguments against considering them as emotions (*e.g.*, Sloman, 1987).

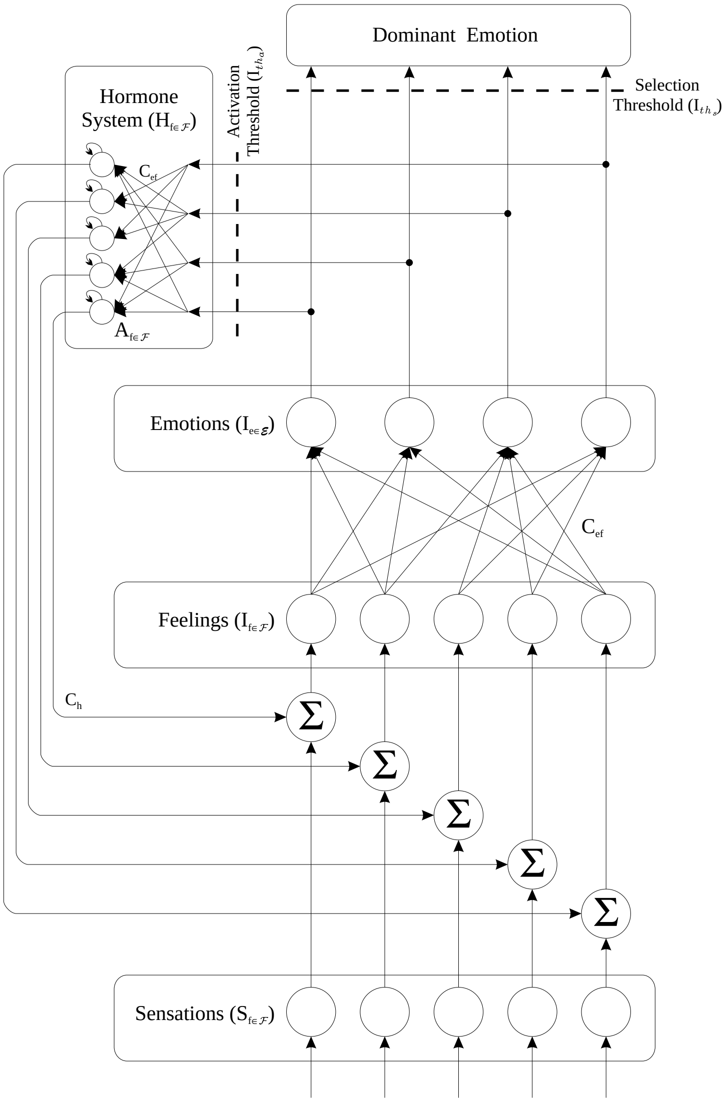

Dominant Emotion
Hormone System ($H_{f\in\mathcal{F}}$)
Activation Threshold (I_{th_a})
Selection Threshold (I_{th_s})
C_{ef}
$A_{f\in\mathcal{F}}$
Emotions ($I_{e\in\mathcal{E}}$)
C_{ef}
Feelings ($I_{f\in\mathcal{F}}$)
C_h
Σ
Σ
Σ
Σ
Σ
Sensations ($S_{f\in\mathcal{F}}$)

Figure 3.1: Emotions model.

body state matches the state that gives rise to that particular emotion. An emotion only influences the body if its intensity value is significantly large, *i.e.* its value is above an activation threshold. In this case, the emotion is considered active.

The emotions influence the body through a hormone system, by producing appropriate hormones. The hormone system in the model is a very simplified one. It consists of having one hormone associated with each feeling. A feeling intensity is not a value directly obtained from the value of the body sensation that gives rise to it, but from the sum of the sensation and hormone value. The hormone values can be (positively or negatively) high enough to totally hide the real sensations from the robot's perception of its body. The hormone quantities produced by each emotion are directly related to its intensity and its dependencies on the associated feelings. The stronger the dependency on a certain feeling, the greater quantity of the associated hormone is produced by an emotion.

On the one hand, the hormone mechanism introduces a sort of fight between the emotions to gain control over the body which is ultimately what selects which emotion will be dominant. On the other hand, the robot feelings are not only dependent on its sensations but are also dependent on its emotional state, *i.e.* the intensity of its emotions.

A formal description of the model's functions is given by Equations 3.3 to 3.8. The function $\mathrm{Th}_{[b_-,b_+]}(x)$ was simply needed to confine values within an interval $[b_-, b_+]$.

$$\mathrm{Th}_{[b_-,b_+]}(x) = \begin{cases} b_- & \text{if } x < b_- \\ b_+ & \text{if } x > b_+ \\ x & \text{otherwise} \end{cases} \tag{3.3}$$

Equation 3.4 shows how the intensity value of emotion e at step n (I_{e_n}) is calculated from the intensity of the feelings (I_{f_n}) at that step. This calculation involves an emotion bias (B_e) and coupling coefficients (C_{ef}) between the emotion e and the feelings f.

$$\forall e \in \mathcal{E}, \forall n \in \mathbb{N}, \qquad I_{e_n} = \mathrm{Th}_{[0,1]}(B_e + \sum_{f \in \mathcal{F}} (C_{ef} I_{f_n})) \tag{3.4}$$

The calculation of the feeling's intensity has to take into account both the influences provided by the hormone system (H_{f_n}), which are dependent on a coefficient parameter (C_h), and the value of the respective sensation (S_{f_n}). The sensations' values are directly derived from the sensory data. The hormone values are responsible for the memory of the emotion system, and depend both on their previous values and the emotion influences (A_{f_n}). Note that these emotion influences are calculated using the same coupling coefficients (C_{ef}) that were used to calculate the emotions themselves. Emotions only influence the hormone values if their intensity is above the activation threshold (I_{th_a}). To calculate the value of the hormones (H_{f_n}), two different system parameters are used, the attack gain (α_{up}) and the decay gain (α_{dn}). The first one is used when the emotions and their influences are increasing and the other when the emotion's intensities are fading away. In general, the attack gain is much higher than the decay gain. This way the decay of emotions is slow while the emergence of new emotions is much faster. The values of these parameters and all the other used in the model are in Appendix A (which also points out some value restrictions).

$\forall f \in \mathcal{F}, \forall n \in \mathbb{N},$

$$I_{f_n} = \mathrm{Th}_{[0,1]}(C_h H_{f_n} + S_{f_n}) \tag{3.5}$$

$$H_{f_n} = \begin{cases} 0 & \text{if } n = 1 \\ \alpha_n H_{f_n} + (1-\alpha_n) A_{f_{n-1}} & \text{if } n > 1 \end{cases} \tag{3.6}$$

$$A_{f_n} = \sum_{e \in \mathcal{E}:\ I_{e_n} > I_{th_a}} C_{ef} I_{e_n} \tag{3.7}$$

$$\alpha_n = \begin{cases} \alpha_{up} & \text{if } |A_{f_n}| > |H_{f_n}| \\ \alpha_{dn} & \text{otherwise} \end{cases} \tag{3.8}$$

The hormones' values can increase quite rapidly, allowing for the quick build up of a new emotional state, and decrease slowly allowing for the persistence of an emotional state even when the cause that gave rise to it is gone — another of the characteristic features of emotions.

Figure 3.2 shows the response of an emotion e to a sensation on which it has a dependency (C_{ef}) of 0.8 weight. This dependency is actually indirect, through the respective feeling f. Assuming that the hormone feedback is initially zero, then when the sensation value (S_f) is 1.0, the emotion intensity (I_e) is 0.8 which is the highest value possible in this example. The influence of the hormone (H_f) is only noticeable after the sensation returns to value zero. Before that, the feeling intensity (I_f) is saturated by the stimulus itself. When the stimulus disappears, the emotion intensity has a sudden drop in value because it becomes dependent solely on the total value of hormone (H_f) that accumulated while the sensation was on. The values of hormone and emotion gradually decay to zero without the presence of the sensation. When the emotion intensity decays to values below the activation threshold, the emotion's influence on the hormone ceases and the values' decay rate increases.

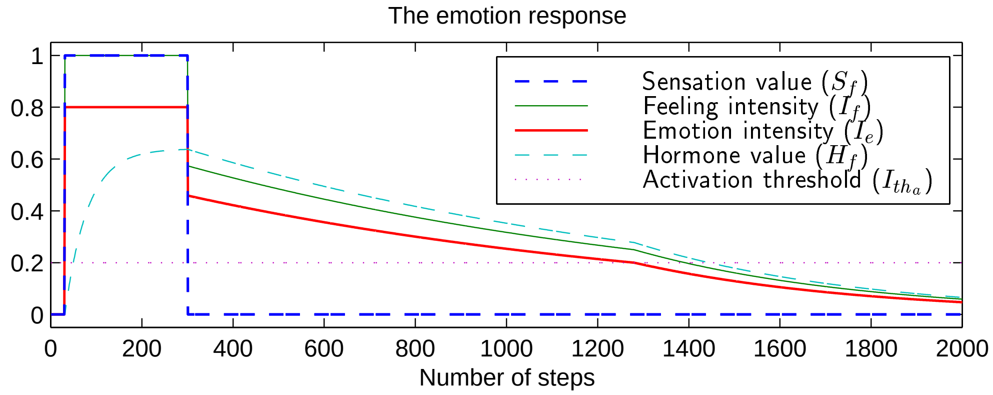


Figure 3.2: Emotional response to a sensation.

As a concrete example of the dynamics of the model in terms of robot-environment interactions consider the situation of the robot colliding with an obstacle. The collision itself produces a pain sensation that will be captured by the pain feeling. Assuming that FEAR has a strong dependency on pain[7], then the FEAR intensity will rise. If this intensity is high enough to make the FEAR emotion active then FEAR will produce hormones. In particular, the hormone associated with pain will quickly build up during the collision. This will make the FEAR emotion grow stronger and possibly overtake other existing emotions. When the robot finally manages to cease the collision, it will still have pain not because the pain sensation is still there, but because the hormone associated with pain has a high value. So the FEAR emotion will persist while the hormone gradually decreases in value. This means that while the robot is gaining distance from the obstacle, the FEAR will still be there. Nevertheless, it will usually fade away as soon as a short distance is gained and the risk of further collisions has diminished.

It should be noted that the time scales involved in the persistence of an emotion after the stimulus is gone, particularly when in the presence of a new stimulus that favours another emotion, are quite small. This allows for what is perceived as quick changes of emotions, in opposition to the much slower process of changes in mood. One can only talk of moods when talking of the residual hormone values that might exist in the system and are not strong enough to stimulate the existence of a dominant emotion.

[7] Although this dependency is used in the experiments, aversive stimulation such as pain is more usually connected to anger in humans (*e.g.*, Izard, 1993).

That would be consistent with the theory that moods are differentiated from emotions in terms of level of arousal (Panksepp, 1995). These residual hormone values can act as moods in the sense that they might favour the appearance of certain emotions, but the short time scales involved in the persistence of the residual values in the model probably make even this interpretation uncomfortable.

The dominant emotion is the one with the highest intensity, unless no emotion intensity exceeds a selection threshold[8](I_{th_s}). In this case, there will not be a dominant emotion and emotional state will be neutral.

Emotions were divided into two categories: positive and negative. The ones that are considered "pleasant" are positive (only HAPPINESS, in the set of emotions used), the others are considered negative. This way a value judgement can easily be obtained from the emotion model by considering the intensity of the current dominant emotion and whether it is positive or negative.

In summary, the model of emotions described provides not only an emotional state, based on simple feelings, that is coherent with the current situation, but also influences the body perception.

Side issues associated with emotions as moods and temperaments were not directly built into the architecture and are only exhibited as a by-product. Different temperaments, for instance, can be achieved by having different emotion dependencies on feelings or changing other parameters of the system.

3.4.2 Discussion

Like many other psychological terms (*e.g.* intelligence, consciousness), emotion is difficult to define and the existing emotion models employ mostly working definitions that tend to conflict with each other. There are even those who defend that emotions are emergent properties of complete agents and should not be engineered in the agent (Pfeifer, 1994).

On the one hand, emotions are essentially a private internal experience not subject

[8] This threshold is independent of the activation threshold, but should probably not be lower to ensure that the dominant emotions are always active.

to direct observation by others than the individual experiencing them, making proper scientific analysis extremely difficult. A behaviourist approach in particular would eliminate the emotions themselves. On the other hand, emotions are intrinsically related to other psychological processes (*e.g.* cognition) and the artificial separation from them created by the traditional scientific approach together with the artificiality of the experimental setups often hide away the true nature of real emotional experiences (Kaiser and Wehrle, 1996).

Research on the emotions field (James, 1890) started by emphasising the role of physiological arousal and emotional behaviour as primary and considering the awareness of the emotional state as the perception of these responses to the situation.

In opposition, recent emotions models usually take for granted that cognition has a fundamental part in the mechanism of emotions, namely that the phenomenon of rational appraisal of the stimuli is essential (*e.g.*, Lazarus, 1982). Most of the them use as evidence for their position the experiment reported by Schachter (1964) which gives some evidence for the need of cognition to label body arousal with particular emotions. However, there are some fundamental problems with this experiment (De Sousa, 1987; Zajonc, 1984). One is that it relies mostly on verbal reports of the subjects and some deceit has even been discovered in their reports after the experiment was finished. Second, it relies on very simplified arousal mechanisms and meanwhile research has shown that emotional arousal is much more differentiated than was previously thought.

It is well known that the reasons people give for their actions are not necessarily the real reasons. This has been particularly well demonstrated by experiments with patients who have the right and left brain hemispheres disconnected, in which the patient would with one of the hemispheres invent *a posteriori* an arbitrary reason for an action commanded by the other hemisphere for a totally different reason (Gazzaniga and LeDoux, 1978). In particular, these experiments showed that emotional outcomes can be transmitted from one hemisphere to the other without the knowledge of their causes being transmitted, demonstrating a dissociation between the emotional reaction and its cause. The mechanisms that lead us to do what we do, and in particular to our emotional reactions, are not necessarily knowable to the conscious self, which can rationalise them to give us the delusion that we act rationally (LeDoux, 1998; Cytowic, 1993). So apart from the deceit involved in reporting an experience to a third party there is also an element of self-deceit by the rational mind.

Emotions in particular are often associated with situations of time pressure where rational decision making similar to the one traditional Artificial Intelligence space search tries to mimic is inadequate. This suggests that emotion mechanisms should rely to some extent on simple associations of stimuli.

This latter view was accepted as important in definition of emotions and of utility in robotics applications when designing the proposed model of emotions. The decision of taking this stance reflects a background in the field of behaviour-oriented Artificial Intelligence, where similar issues are discussed under different denominations.

While most of the computational models of emotions rely on distinct entities that are labelled after human emotions, many researchers, particularly those looking into bottom-up approaches, would prefer the total dismissal of emotion labels. This is a valid approach in that emotions' categorisation is unnecessary to their existence and it can even be argued that the categorisation process is only done at a conscious, and therefore higher, level than the one required by the initial stages of a bottom-up system. The problem is how to take into consideration the different distinctions between emotions. In particular, an approach that reduces emotions to a simple unidimensional pleasure/displeasure vector (*e.g.*, Kitano, 1995; Foliot and Michel, 1998) loses much of the richness provided by emotions.

There are basically two views for the process of categorising emotions with different labels (Wehrle, 1998): emotions can either be considered emergent labels for the evaluation of prototypical situations or events (modal emotions) or evolutionarily achieved response programs (basic emotions). Either way their richness cannot be reduced to a one dimensional vector (Ekman, 1992).

Nevertheless, using existing emotion labels is not always an elegant solution. At times the need for the emotions to be in tune with the agent-environment interaction will make their meaning farfetched from their human counterpart (Wehrle, 1998) . However, labelling them often allows a quick grasp of what they stand for.

From a more practical view, using different emotions can be useful by providing a way to separate the different problems the robot is confronted with into different categories. This will, for instance, permit a modular multi-dimensional reinforcement function when emotions are used as reinforcement (which can be useful, for example, to allow the agent to concentrate on danger and ignore hunger when under threat); or allow for each problem to be solved separately, if emotions are used to divide the problem space into smaller sub-problems.

The proposed model is full of simplifications and ignores many of the features expressed in current definitions of emotions. Furthermore, because the robot's environment is very simple, emotions themselves will also be very simplified — simplified, perhaps, to the point where their distinction from simpler mechanisms as drives or motivation systems becomes diffuse. However, features that are characteristic of emotions alone (*e.g.* persistence and valence) were reproduced to give them more authenticity.

Emotions have evolved from rigid adaptive systems as reflexes and physiological drives, but are more flexible mechanisms because they involve an appraisal of significance of the events in terms of the survival of the individual and action tendencies instead of a direct coupling from events to action (Staller and Petta, 1998). In the model presented here this flexibility is achieved in that events are not directly transformed in actions but are subjectively evaluated as emotion value. Instead of proposing rigid behavioural solutions, emotions provide guidance for behaviour by attributing this value as reinforcement to the performance of the robot's behaviour in terms of its final goals.

One of the simplifications consists in the fact that the model only incorporates simple linear dependencies of feelings in the definition of emotion arousal. This has some limitations. For instance, the distress caused by hunger should be much more noticeable when hunger reaches dangerous levels (Balkenius, 1995) which suggests that the dependence between the two should perhaps be exponential. Moreover, the dependency should probably not be monotonic. The distress level should possibly rise if the agent is consuming too much food. In general, stimuli are not minimised or maximised but kept within comfort values. Nevertheless, this process that we perceive as homeostasis, *i.e.* keeping a value within bounds, is often made with the aid of certain environmental conditions (Bolles, 1980). Furthermore, emotions are modelled as simple response to events, without any anticipatory power attached to them. However, unexpectedness can affect emotion intensity (Ortony et al., 1988) and certain emotions can be associated with the notion of expected reward (Balkenius, 1995). For example, anger is often triggered when an expected reward is not obtained and fear can be seen as a reaction to an expected negative reward.

The hormone system developed is also very simplified and does not try to mimic biological hormone systems and the naming might be misleading by suggesting different functions than those modelled in the system. Hormone discharges are usually associated with transformations in the functioning of the nervous system induced by emotions, but rather at the level of behavioral output (Kravitz, 1988) than at the level of perception. Nevertheless, emotions are responsible for moving certain body sensations from the background to the foreground of our attention (Damásio, 1994). Moreover, there is evidence to suggest that sensations are not produced only by stimuli but also by brain processes. Melzack (1997) defends that sensory input only modulates the experience of the body generated by the brain, they do not directly cause it. Pain is referred by Melzack as a demonstrative example: only if a local anaesthetic is delivered to a person in time to prevent the early pain response does the later pain totally disappear.

There is no reason to claim that the developed model provides the robot with the ability to feel emotions in the sense the humans do. To start with, the body plays a crucial role in human emotional experience (LeDoux, 1998). A robot's underlying composition is very different from human physiology and the sensors of its physical state that might define its emotional feelings would also have to be very different (Picard, 1997). Furthermore, its lack of consciousness (Frijda and Swagerman, 1987; Ortony et al., 1988) together with the fact that its emotions are far from the complexity of true emotions as experienced by humans makes such an assumption ludicrous. In reality, it was considered more important to design emotions that could be afforded by the robot-environment interaction than to equip the robot with human-like emotions (Cañamero, 1998). However, language will be used that might, implicitly, attribute emotional feelings to the robot. This kind of language is used only because it is more practical and concise. Nevertheless, a different typeface was used for the agent's “emotions”.

3.4.3 Application

The model of emotions behaves appropriately when tested on the robot, in the sense that the robot consistently displays plausible contextual emotional states during the process of interacting with the environment. Furthermore, because its emotions are grounded in subjective body “feelings”, and not direct sensory input or “sensations”, it manages to avoid sudden changes of emotional state, from one extreme emotion to a completely different one. The more different the two emotions are, the more difficult it is to change from one to the other. The physiological arousal caused by emotions was repeatedly left out of cognitive theories of emotions, because it was not considered cognitively interesting, yet without it emotions lack their characteristic inertia (Moffat et al., 1993). Nevertheless, recent artificial emotion models based in a sub-symbolic approach do often try to model this feature (Picard, 1997; Velásquez, 1998; Breazeal, 1998).

The developed model does not endow the robot with the feeling of emotions, in the sense that it has a conscious and subjective experience of emotions (Frijda and Swagerman, 1987), but more importantly it endows it with an emotional state that can be used to affect its behaviour.

In order to evaluate the functional role of emotions in reasoning, the emotional state should be used for the actual control of a complete agent, determining its behaviour (Albus, 1990; Wright, 1996; Moffat et al., 1993). Furthermore, it is important to show empirically that endowing the robot with emotions has adaptive value by comparing the developed emotional robot with other non-emotional robots. Although emotions research in biological systems can be a source of inspiration to guide robot design, it is not by itself a valid proof of the adaptive value of artificial emotions for artificial systems (Cañamero, 1998). In the next chapters, examples will be given of its use in robot experiments. In particular, emotions will be used to fill in much of the specifications left open by the selected learning architecture described in the previous chapter.

Chapter 4

Action-Based Control

4.1 Introduction

In this chapter, a description is given of a first attempt at integrating an emotional system with the control of an autonomous robot. To start with, emotions were allowed to influence control by providing an evaluation of the context.

To investigate the validity of this approach, several experiments were carried out using a robot simulation. The robot was given a simple survival task that requires learning. A reinforcement learning controller was developed to solve the task. This controller makes use of well-known techniques: a Q-learning algorithm to learn its policy and neural networks for storing the utility values.

Unfortunately, experiments showed that, contrary to expectations, the emotion-based evaluation was inadequate as a reinforcement signal for policy acquisition by Q-learning. The problem was investigated and further experiments were done to find other ways in which emotions could be more helpful to the action-based controller. The use of emotions as modulators of learning system parameters proved much more fruitful.

A detailed description of the experimental setup is presented in the next section, Section 4.2, covering the robot's task, emotional system, controller, and experimental evaluation. This is followed by a report of the experiments done and the results obtained in Section 4.3. Apart from the main experiments, other experiments were done to determine why the emotion reinforcement was unsuccessful and to explore other ways in which emotions can influence control. The results for these experiments can be found in Sections 4.4 and 4.6, respectively. In between these two sections, the results achieved until that point and the problems found with emotion-dependent reinforcement are summarised. The conclusions reached at that point are later summarised, together with the conclusions for the alternative emotion roles experiments, in the section at end of this chapter. For further implementation details consult Appendix C.

4.2 Experimental Setup

4.2.1 Robot, environment and task

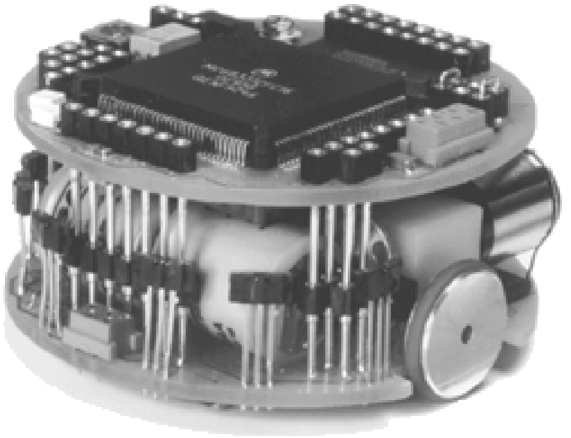
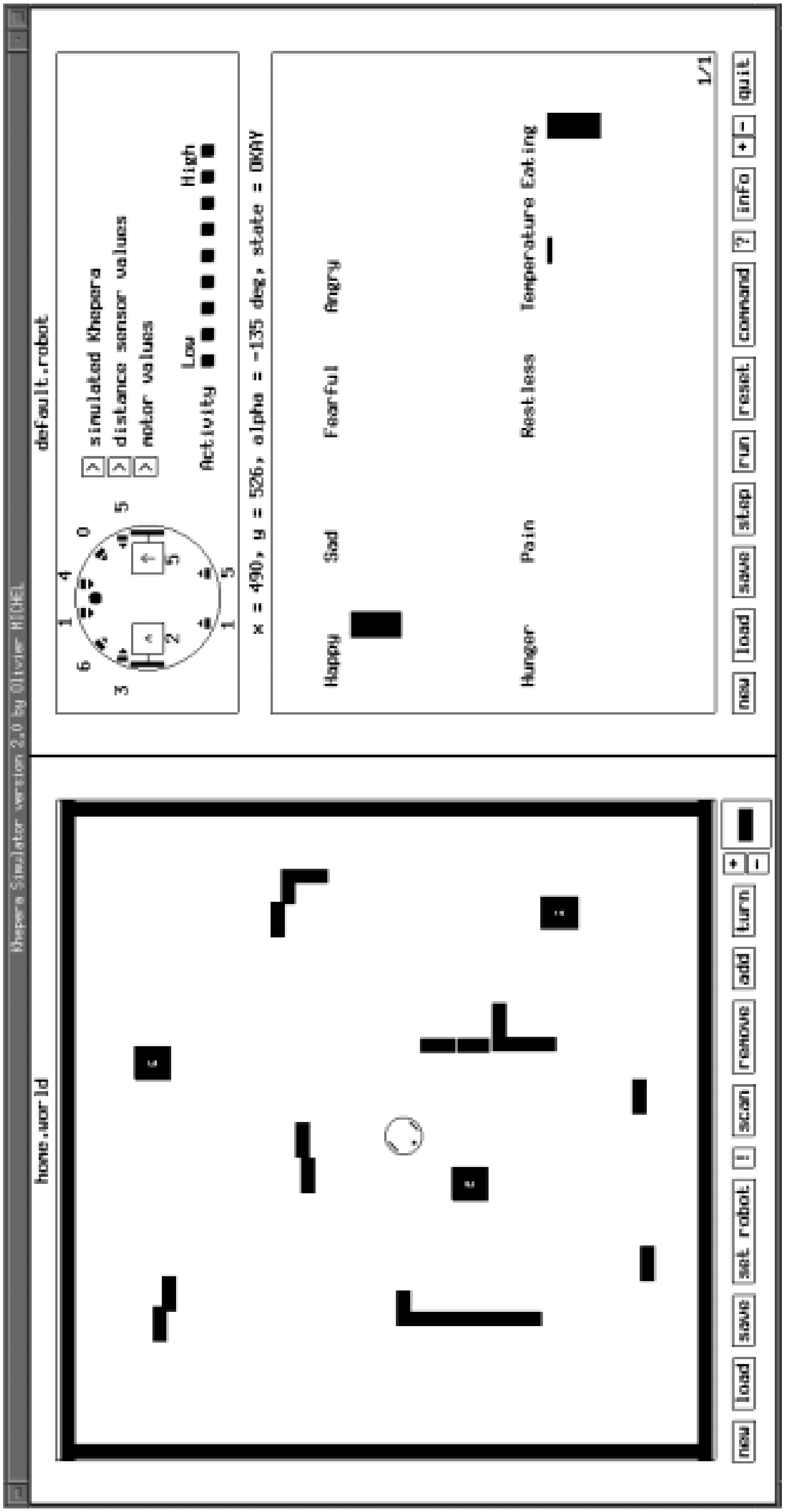


Figure 4.1: The Khepera robot.

All the experiments were carried out in a simulator (Michel, 1996) of a Khepera robot (Mondada et al., 1994) — a small robot with a left and a right wheel motor, and eight infrared sensors that allow it to detect object proximity and ambient light. Six of the sensors are located at the front of the robot and two at the rear. Figure 4.1 shows the original robot. The experiments were done with the simulated robot within the environment shown by Figure 4.2, which is a closed environment with some walls and three lights surrounded by bricks[1]. Figure 4.3 gives an idea of the sensor capabilities of the simulated robot. Figure 4.3(b) shows the values for the infrared distance sensors

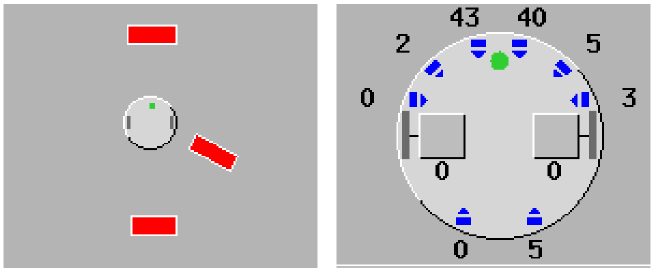
[1] The lights had to be surrounded by bricks to avoid the robot becoming permanently stuck in their concavities. The lights can still be perceived by the robot as the bricks are transparent to light.



x = 490, y = 526, alpha = -135 deg, state = OKAY

Figure 4.2: The simulated robot and its environment.

obtained for situation 4.3(a). The maximum distance sensor range is in between the distance to the front brick, which is barely detected, and that to the rear brick, which is not detected at all. The third brick, on the right side, shows how obstacles can be very close to the robot without being detected.



(a) Robot in environment.

(b) Sensor values.

Figure 4.3: The infrared sensor readings of proximity for situation 4.3(a) is given in 4.3(b). It should be noticed that these readings can vary between 0 and 1023 and that the very low values, *e.g.* 5, are due to noise.

The ultimate goal of the research reported in this dissertation is to develop a fully autonomous real robot. This was one reason why self-sufficiency was considered a useful property to include in the system. Another reason for this choice was that it is easier to conceptually ground emotions in the context of an animal-like creature with self-maintenance needs. Simulated feeding needs were therefore added to the robot. The robot is always losing energy: the more it uses its motors the more energy is used up. It can recover its energy from light. More exactly, the amount of energy that the robot acquires at each step depends on whether enough light is being received by the two front sensors and on how much light is being received by those sensors. The main reason for having lights as food sources is to allow the robot to distinguish its food sources with its poor perception capabilities. Apart from feeding itself by standing next to the lights, the robot is supposed to wander around and avoid walls.

4.2.2 Emotion system

An emotion system was developed based on the emotion model presented previously in Chapter 3.4 and using the following feelings ($\mathcal{F}$): Hunger, Pain, Temperature, Restlessness and Eating. The sensations that give rise to these feelings are[2]:

Hunger — is directly related to its current energy deficit.

Pain — is active if the robot is bumping into obstacles.

Temperature — depends upon the usage of the motors; as long as high velocity is being demanded of the motors, the temperature will rise[3].

Restlessness — increases if the robot does not move.

Eating — depends on the amount of energy the robot is acquiring at the moment. Its value is high when the hunger sensation is decreasing.

The values of the emotions' dependencies on feelings and biases (see Table 4.1) were carefully chosen by hand to provide adequate emotions for the possible body states. The process of selecting these values consisted in first deciding which combination of feelings should lead to an emotional reaction taking into consideration the robot task, and then selecting some initial dependencies accordingly. These were afterwards corrected if the observation of the robot's emotional reactions while running showed any unexpected deficiencies. This did not involve many adjustments and mostly consisted in balancing the different emotions so that the right emotion would be dominant in each specific emotional context. Some initial tentative dependencies had the drawback of allowing the saturation of the emotional system but simple restrictions on the dependencies values were found that eliminated this problem (details in Appendix A). The emotions are such that:

- The robot is **happy** if there is nothing wrong with the present situation. It will be particularly HAPPY if it has been using its motors a lot or is in the process of getting new energy at the moment.
- If the robot is restless, has very low energy and it is not acquiring energy, then its state will be SAD.

[2] Further details in Appendix C.1.

[3] The real robot's velocity does not matter, in fact; the robot can be demanding high speed from its motors while heading motionless against a wall.

- If the robot bumps into the walls then the pain will make it FEARFUL.
- If the robot is hungry, restless and with pain it will get ANGRY.

	Hunger	Pain	Restlessness	Temperature	Eating	Bias
HAPPINESS	-0.2	-0.3	-0.2	0.2	0.7	0.1
SADNESS	0.7	0.0	0.5	0.0	-0.4	0.0
FEAR	-0.4	0.8	-0.2	0.15	0.0	0.0
ANGER	0.2	0.2	0.3	-0.2	0.0	0.0

Table 4.1: The emotions' dependencies on feelings.

4.2.3 Basic controller

The role of the basic learning controller is to produce actions that maximise the expected evaluation received. To achieve this purpose the controller can select one of six possible discrete actions which are specified in detail in Appendix C.3:

- move slowly forward;
- move fast forward;
- turn left;
- turn right;
- stop;
- move slowly backwards with a slight twist to the right.

The controller — Figure 4.4 — implements a Q-learning algorithm using neural networks very similar to the one reported by Lin (1992), which was presented in Section 2.5.1. It will be defined next in terms of two separate modules:

Associative Memory Module — This plastic module associates the sensor readings and feelings with the current expected value of each of the actions that the robot can take.

Action Selection Module — Based on the information provided by the previous module, this module makes a stochastic selection of the action to take at each step.

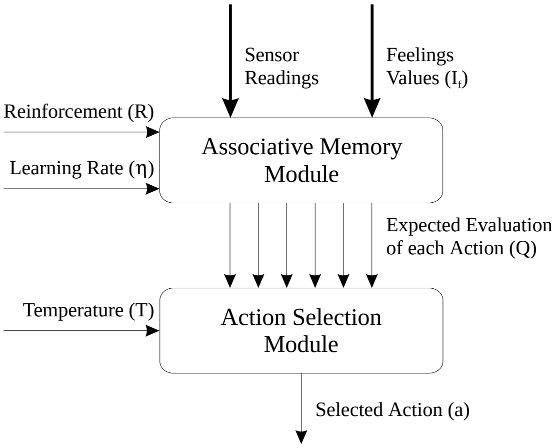


Figure 4.4: Basic controller for action-based control.

Associative Memory Module

The associative memory consists of six neural networks that each try to predict the outcome of selecting each one of the six available actions. Each network is a three layer feed-forward network with:

- 22 input units: one for each distance(8) and light(8) sensor[4], one for each feeling(5) and a bias;
- 5 hidden units;
- 1 output unit that represents the expected outcome if the action associated with this net is selected in the situation represented by the input units.

The activation functions used were the hyperbolic tangent[5] in the hidden layer and the identity function in the output layer. This allows the output nodes of the neural networks to have values outside the interval between minus one and one. The weights between the hidden layer and the output layer are initialised with random values, and the weights between the input layer and the hidden layer are set to zero. This way all the networks will provide an initial neutral evaluation. The learning algorithm used to

[4] The values of IR sensors were converted to values varying between zero and one, with one representing maximum intensity, before being given as input to the networks.

[5] More specifically: $\tanh(\beta x) = \frac{1-e^{-2\beta x}}{1+e^{-2\beta x}}$, $\beta = 0.25$.

train the networks was back-propagation (see, for example, Hertz et al., 1991, for a full description).

First attempts that used the networks to associate the received evaluation, *i.e.* the reinforcement R, with the network inputs were a failure because the robot's learning was very poor. Learning from delayed rewards with Q-learning (Watkins, 1989) proved to be much more successful. The networks were used to learn utility functions that model $util(s_n, a)$:

$$\text{util}(s_n, a) = R_{n+1} + \gamma \ \text{eval}(s_{n+1}) \tag{4.1}$$

The discount factor (γ) was set to 0.9. The function eval(s_{n+1}) is the expected cumulative discounted reinforcement starting from the state s_{n+1} reached by doing action a in state s_n. The value R_{n+1} is the immediate reinforcement in iteration $n+1$. For each iteration, the target value $T_n(s_{n-1}, a)$ will be given to the network whose action was used in the previous iteration:

$$T_n(s_{n-1}, a) = R_n + \gamma \ \max\{Q_n(s_n, k) \mid k \in \text{actions}\} \tag{4.2}$$

After an action a has been evaluated its network state for situation s_{n-1} is saved. The network state is defined by the current value of each one of its units, *i.e.* by the input values, the hidden units values and the output values of the network. The new estimative of the utility value ($Q_n(s_n, k)$) of each action k for the new state s_n is calculated. The maximum is obtained and used in the previous formula to update the weights of the network associated with action a and the previous situation. Just before learning by back-propagation takes place, the network's saved state for situation s_{n-1} is restored. After the learning has taken place, the utility value for action a is recalculated for state s_n.

This way, apart from updating the network's prediction with the experience provided by the last action taken, new predictions are calculated for the present situation. Those will be used by the Action Selection Module, described below, to decide which action to take next.

Action Selection Module

The utility values provided by the associative memory are used for the stochastic selection of the next action to take. The higher the value provided by the associated net, the higher the probability of an action to be selected.

The function used to calculate the probability of each action is based on the Boltzmann-Gibbs distribution. For a selection temperature[6] T, the probability of selecting action a is:

$$P_n(s_n, a) = \frac{e^{\frac{Q_n(s_n,a)}{T}}}{\sum\limits_{k \in \text{actions}} e^{\frac{Q_n(s_n,k)}{T}}} \tag{4.3}$$

The selection of a new action is not made every cycle; there is a certain inertia of the current action that is directly correlated with its probability. The reason for this is to have a more coherent behaviour. Otherwise, the robot would spend most of its time trembling, because it would be selecting different actions at each step.

An action is only evaluated, and eventually a new one selected, every second step, unless there is a significant change in the emotional state, *i.e.*, a change from one of the following states to another:

- a positive emotion is dominant;
- a negative emotion is dominant;
- no emotion is dominant.

Even if an evaluation takes place, the probability of not choosing an action based on the above criteria (Equation 4.3), if a is the currently selected action, is:

$$P_n(\text{No Selection}) = \sqrt[10]{P_n(s_n, a)} \tag{4.4}$$

This way the probability of an action being selected at a given step is extended to a probability of its being consecutively selected in the following next ten steps.

[6] The selection temperature is not related at all to the robot's temperature feeling.

Both the mechanisms described attempt to give a more coherent behaviour to the robot, yet care was taken not to do this at the expense of:

- Preventing the controller from taking notice of sudden changes in the emotional state;
- Giving preference to the previous action independently of how well rated that action is;
- Giving preference to the previous action even when the conditions have changed significantly, making it inappropriate to do so.

Summary of one control iteration

```
% Action $a$ was taken previously in state $s_{n-1}$
% State $s_n$ reached and reinforcement $R_n$ received

PreviousState $\leftarrow$ network[$a$].state;

For $k \in$ Actions do
    network[$k$].update($s_n$);
    $Q_n(s_n, k) \leftarrow$ network[$k$].output;
end;

$T_n(s_{n-1}, a) \leftarrow R_n + \gamma \max\{Q_n(s_n, k) \mid k \in \text{actions}\}$;
network[$a$].state $\leftarrow$ PreviousState;
network[$a$].learn($T_t(s_{n-1}, a)$);
network[$a$].update($s_n$);
$Q_t(s_n, a) \leftarrow$ network[$a$].output;

if there is a change in emotional state or
   random number $\in [0, 1) > \sqrt[10]{P_n(s_n, a)}$ then
    Select new action using the probabilities provided by Equation 4.3;
else
    Select action a again;
end;
```

4.2.4 Experimental procedure

All experiments consisted in having the robot learn for two thousand steps followed by an evaluation of its performance, for another two thousand steps, with learning turned off. In total, the robot takes one hundred and twenty thousand learning steps and sixty one evaluation tests, one test after each learning period plus one extra test before any learning takes place. Tests were made transparent to the experiment: when continuing with its learning the robot's state is restored to the state just after its last learning step and previous to the test.

The robot's evaluation was based on the reinforcement values it received in its testing period. There were two evaluations, each based on a different reinforcement function. One was the mean of the **emotion-dependent reinforcement** values and the other was the mean of the **sensation-dependent reinforcement** values obtained during its test period (Sections 4.3.1 and 4.3.2 give a detailed description of the two reinforcement functions). The higher the value, the better is the evaluation. Good robot behaviour (*i.e.* task-adapted behaviour) is usually associated with positive reinforcement and bad behaviour with negative reinforcement. Nevertheless, positive reinforcement is usually sporadic so mean reinforcements are not expected to take very high values. Qualitative evaluations made by an external observer are also reported which show this association of higher reinforcement values with better overall performance of the robot in its task.

For each experiment, this whole procedure was performed fifty times so that an average of the evaluations over several trials could be obtained. Each trial had a new robot with all state values reset and placed in a randomly selected starting position. There are twenty possible starting positions, shown in Figure 4.5, that were chosen to maximise the differences in starting conditions, but were otherwise arbitrary.

The experimental data shown in the result graphs are the means of the reinforcement values obtained during the sixty-one testing phases in each of the fifty runs. The error bars show the 95% confidence intervals[7].

The robot was designed to learn continuously, as any autonomous robot should, and therefore it might seem strange to have a distinction between a learning phase and

[7] See Appendix B for detailed calculations.

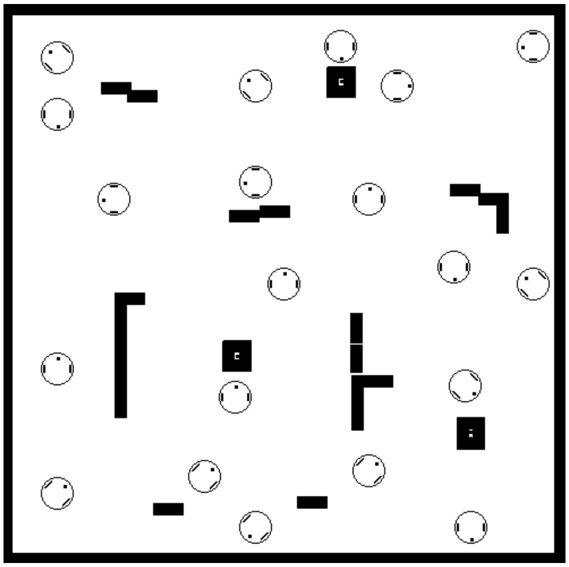


Figure 4.5: The robot's starting positions.

a performance phase. The idea behind having a testing phase with no learning is that each step in a test represents a snapshot situation that the controller has to deal with. If the robot was allowed to learn while under evaluation the resulting evaluation would be the mean performance of consecutive controller learning stages and not the instantaneous evaluation of the controller's current learning stage.

4.3 Experimental Results

The purpose of the experiments reported in this section was to test whether *an emotion-based evaluation of the context is adequate as a reinforcement signal for policy acquisition by Q-learning*[8].

The results of an experiment that uses emotions as a source of reinforcement are given and compared to those of a control experiment that uses a more traditional reinforcement function based on raw sensations. The controllers used in each experiment are:

- **Emotion-driven Controller** — The basic learning controller using emotion-dependent reinforcement.

[8] This and other specific experimental hypotheses to be tested are highlighted in italics in the course of this dissertation.

- **Sensation-driven Controller** — The same basic learning controller but using sensation-dependent reinforcement instead.

To give a clear idea of the learning algorithm's performance the experimental results for two other controllers are also presented:

- **Random Controller** — Shows how the basic controller would perform if it did not learn at all.
- **Hand-crafted Controller** — Shows how well a competent controller can perform in practice.

The results for each of the four different controllers will be given next, one by one.

4.3.1 Emotion-driven controller: Emotion-dependent reinforcement

The first controller tested, the emotion-driven controller, uses an emotion-dependent reinforcement ($R_n = R_{e_n}$) which is defined in Equation 4.6. The reinforcement magnitude was set to be the intensity of the current dominant emotion or zero if there was no dominant emotion. If the dominant emotion was negative then its positive intensity value would be negated.

$$\forall e \in \mathcal{E}, \quad \text{sign(e)} = \begin{cases} 1 & \text{if } e \text{ is positive} \\ -1 & \text{if } e \text{ is negative} \end{cases} \tag{4.5}$$

$$R_{e_n} = \begin{cases} 0 & \text{if } \forall e \in \mathcal{E}, I_{e_n} < I_{th_s} \\ I_{e_n}\text{sign}(e) \text{ where } e = \arg\max_{e\in\mathcal{E}}(I_{e_n}) & \text{otherwise} \end{cases} \tag{4.6}$$

Experimental results are shown in Figure 4.6. The right graph shows the values for the reinforcement function used in this experiment. The left graph shows the reinforcement function based on direct sensations which was only calculated and shown for direct comparison with the results of the controller presented next.

The initial analysis of the results suggested the existence of two quite differentiated populations, one that manages to learn the task and another that does not. For this

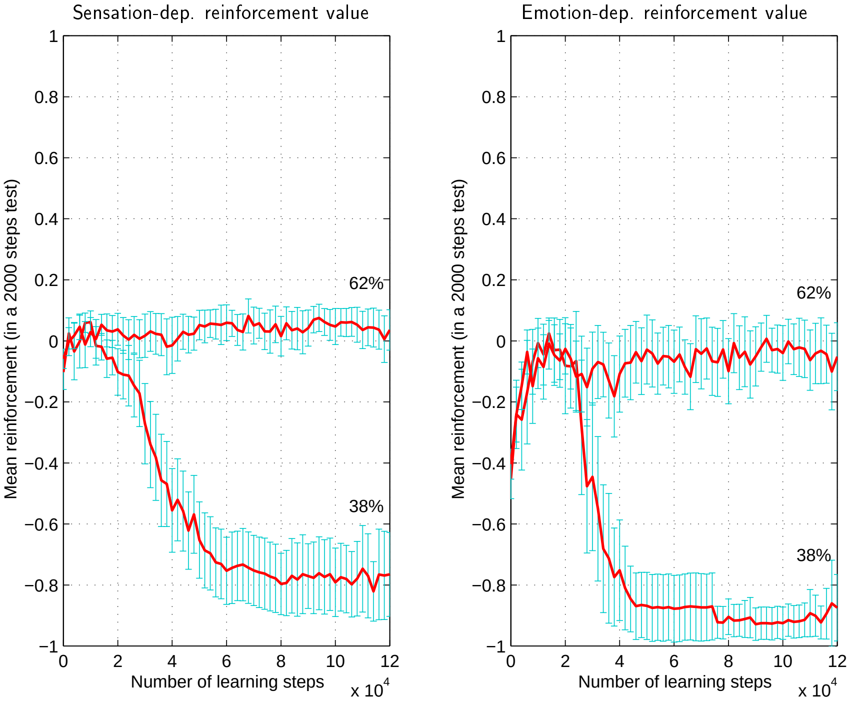


Figure 4.6: Emotion-driven controller: Reinforcement values registered while the robot was learning with emotion-dependent reinforcement (right graph). The sensation-dependent reinforcement values (left graph) were calculated for comparison with other experiments.

reason, in the presentation of the results, two populations were distinguished based on the emotion-dependent reinforcement obtained at the last ten evaluation points. The trials that had quite negative reinforcement for these testing points formed one population which amounted to 38% of the total. The remaining trials were included in the “successful” population.

It should be noticed that the robot’s adaptation task must be achieved in a limited amount of time. If the robot takes too long to adapt, the reinforcement will lose meaning and the task will become impossible. The reason for this is that if the robot does not learn to feed itself, it will get increasingly hungry. It will eventually arrive at a state where it will keep getting low reinforcement on account of its hunger, independently of what it does. This is why two very different populations can co-exist: only one managed to learn the task before being dominated by hunger.

An alternative partition of populations was made based on the robot’s final behaviour. This new partition is consistent with the previous in that good behaviour is usually associated with good reinforcement and bad behaviour with low reinforcement. A total of 58% of the robots managed to converge to a suitable behaviour, namely circling near a light or just wandering near a light in such a way that they receive plenty of light and never get hungry. Two of these robots made use of the fast-forward action, with different levels of success, to achieve a higher reinforcement through the increase in temperature. However, the remaining 42% end up behaving in a totally inappropriate way (*e.g.* bumping into walls or lights). If the robot ends circling in an open space getting increasingly hungry because it was not near any light, its behaviour was also considered inappropriate. This would happen frequently enough to hint that the circling behaviour near a light was not a robust behaviour, but a sort of *accidental* behaviour. In fact, if the robot were moved away from the light to a open space, it would just remain with its circling behaviour as if nothing had happened. In time, after hunger begins to be noticeable, it will learn to behave differently. However, this change of behaviour will be mostly due to new learning and not to any previous learning.

4.3.2 Sensation-driven controller: Sensation-dependent reinforcement

At this point, the use of emotions to provide reinforcement was re-evaluated. The reason for the poor results obtained in the previous experiment appears to be that the controller is not receiving the kind of reinforcement it needs. The controller needs a good evaluation of the situation as it stands at the moment and not the mixed evaluation of present and recent past situations that the emotions provide.

The reinforcement provided by emotions can thus be quite misleading. Even when a good action selection is made, the robot may still receive negative reinforcement (and vice versa). An example will make this problem clear. Imagine that the robot bumps into a wall. It will feel pain and therefore become FEARFUL. During the time that it is close to the wall it will be FEARFUL but even if it finally manages to go away, by taking a move-backwards action for example, it will still receive negative reinforcement because the FEAR emotion will persist for a while even when the wall is out of reach. So, although the FEAR intensity will get smaller, it will still be providing inappropriate negative reinforcement.

It looks as if *the reason for the emotion-dependent evaluation failure is the recurrent* *and lateral influences of emotions in the model.* To test this hypothesis, a control test was made that consisted in using a controller, the sensation-driven controller, with a putatively more adequate reinforcement signal ($R_n = R_{s_n}$). This new signal was based on the previous one but without any temporal or lateral side effects. The sensations were used instead of the feelings to calculate the value of each emotion and the highest of these values was selected to be the reinforcement value. Figure 4.7 illustrates this modification and Equation 4.7 shows the resulting reinforcement function. As in the previous experiment, the value of negative emotions was negated. This procedure provides a more traditional reinforcement that directly reflects the immediate situation.

$$R_{s_n} = \left(B_e + \sum_{f \in \mathcal{F}} (C_{ef} S_{f_n}) \right) \operatorname{sign}(e) \quad \text{where} \quad e = \arg\max_{e \in \mathcal{E}} \left(B_e + \sum_{f \in \mathcal{F}} (C_{ef} S_{f_n}) \right) \tag{4.7}$$

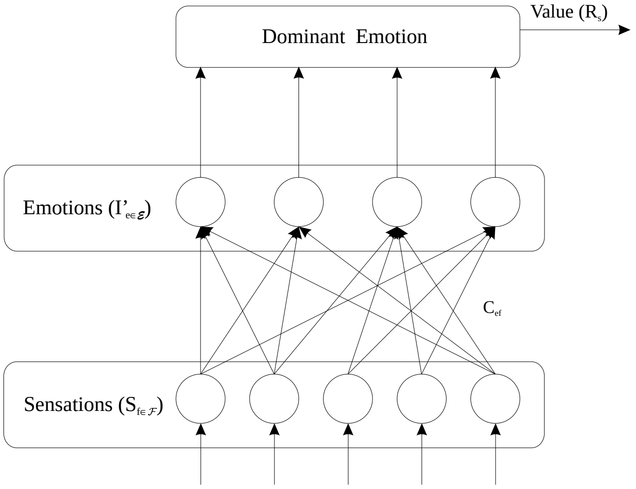

Value (R$_s$)
Dominant Emotion
Emotions (I'$_{e\in\mathcal{E}}$)
C$_{ef}$
Sensations (S$_{f\in\mathcal{F}}$)

Figure 4.7: Truncated emotion model used to obtain the sensation-dependent reinforcement.

The results obtained in this experiment are shown in Figure 4.8 in terms of the mean sensation-dependent reinforcement and the mean emotion-dependent reinforcement. The emotion-dependent reinforcement value graph is given for comparison with previous results while the sensation-dependent reinforcement is the reinforcement actually received by the robot. They are a considerable improvement on the results obtained with the previous experiment which used emotion-dependent reinforcement.

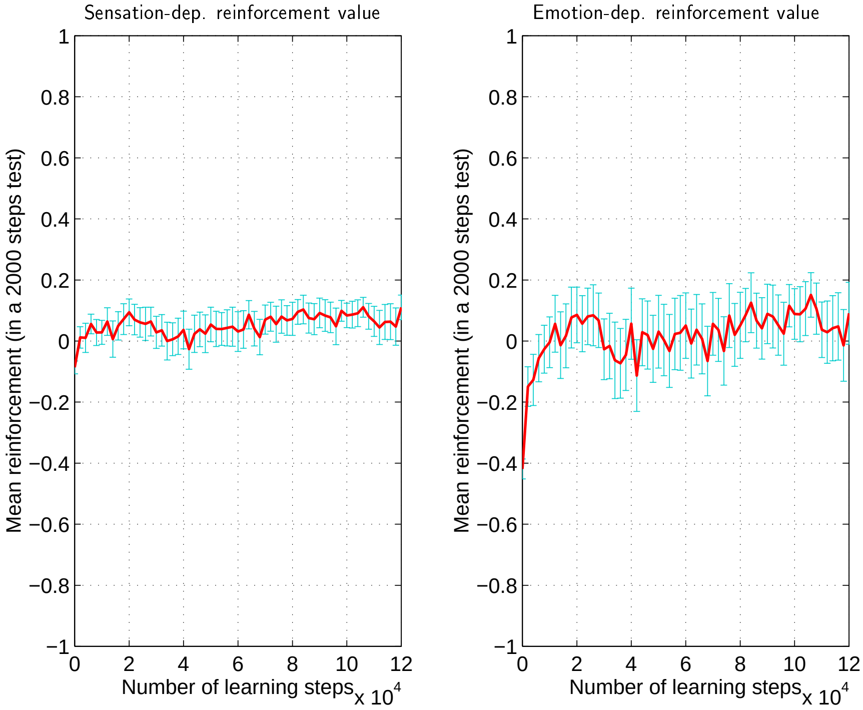


Figure 4.8: Sensation-driven controller: Reinforcement values registered while the robot was learning with sensation-dependent reinforcement (left graph). The emotion-dependent reinforcement values (right graph) are shown for comparison with other experiments.

Qualitatively, most of the final behaviours of the robot were quite successful. Many would converge to the circling behaviour near a light or wandering near a light. These behaviours tend to be much more robust than the ones of the previous experiment, in that in general there was not so much preference for just one action. Instead, the final behaviours would use a small subset of actions involving both forward and backward turning movements much more often, which made them withstand better being placed away from the light. Another group, about 40%, were wandering about using the fast-forward action a lot. This kind of behaviour gets them very good reinforcement, because this action raises the temperature, which is considered beneficial. However, because these robots do not always keep within reach of one light, their reinforcement is unstable. The reinforcement may suffer substantial drops, if the robot becomes significantly hungry because of being away from a light source for a while. Although this kind of behaviour has better reinforcement in general, it was only learned once in the emotion-dependent reinforcement experiment. There was another case in that experiment where the same sort of behaviour was found, but the robot would bump into lights and walls all the time. This led to the suspicion that this kind of behaviour under emotion-dependent reinforcement might degenerate into some sort of bumping behaviour. To test this hypothesis, a small experiment was run that consisted in starting the emotion-dependent reinforcement learning experiment with robots that had converged to this wandering behaviour. Two out of the ten robots tested converged to crashing behaviour. When the same experiment was done with sensation-dependent reinforcement all robots maintained their wandering behaviour.

In the evaluation of the behaviours just described, the restrictive short range of the robot's sensors and the fact that the inputs of the networks provide only a view of the current situation should be taken into consideration. So one cannot expect from the robot some kind of complex behaviour that depends on previous actions or sensings or some sort of global map of the environment.

4.3.3 Random controller: No learning

The results obtained with both emotion-dependent and sensation-dependent reinforcement do not seem very impressive in terms of final reinforcement obtained. Even the most successful robots do not seem to do much more than to maintain their average reinforcement. However, it should be clear that even just maintaining reinforcement is quite good. A controller selecting randomly between all available actions will actually have decreasing reinforcement, because of increasing hunger, throughout the entire experiment.

Figure 4.9 shows how the robot performs over time without any adaptation, just with a random action selection controller. The left graph of Figure 4.9 shows the values of the reinforcement function based on direct sensations and the right graph shows the reinforcement function based on emotions. The immediate sensations provide a steady and gradual decrease of reinforcement over time, that reflects the decrease in energy level, while emotions suffer a much more significant drop right at the start due to the recurrent nature of the emotions model.

The experimental setup provides temporal constraints that add complexity to the problem. As observed previously in the emotion-driven controller section, the time the robot

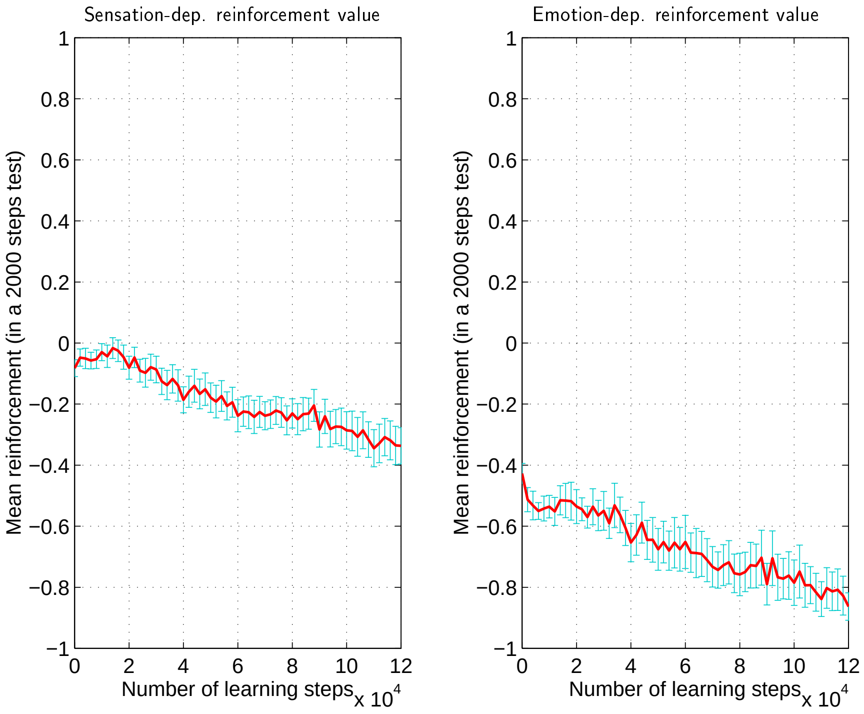


Figure 4.9: Random controller: Reinforcement values registered with a non-learning robot.

takes to learn its task is crucial to its successful adaptation. If it takes too long, the reinforcement will lose meaning before any adaptation can be achieved. It should be noted that this does not make the problem unsolvable, but is an added difficulty that is successfully overcome by the sensation-driven controller.

4.3.4 Hand-crafted controller: Competent initial state

The fact that the reinforcement received by the robots can theoretically reach the value of 1.0 is misleading in suggesting that a successful learning controller should, in time, reach and maintain such a reinforcement. In practice this is not possible, because maximum reinforcement can only be achieved during short periods of time widely separated from each other.

In order to have a better understanding of the level of performance of the learning controllers, a controller was designed to take full advantage of its environment and achieve high reinforcement, by carefully selecting the initial weights of the networks. There were two main reasons to hand-craft this controller:

- to determine how much reinforcement a reasonably successful behaviour might receive in practice;
- to check if a successful behaviour is stable under the learning algorithm.

To simplify the process of design of this controller, the set of actions from which the controller makes its selection was slightly modified. In the new set of actions, the backward movement is done in a straight line instead of with a twist to the right.

The designed behaviour consisted in having the robot, oriented towards a light, selecting between fast forward movement and backward movement, depending on how far from the light it was at each point. This would give rise to a kind of interleaved attraction-and-repulsion-to-light behaviour. The robot's reinforcement would be optimal because its temperature would reach its maximum value, the robot would eat a lot and would not have any hunger, pain or restlessness. The networks' initial weights were pre-defined in such a way that the result behaviour would be the one just described. Some minor settings of the weights were made in order to give it a little of avoiding behaviour, although this was not very successful: the avoidance behaviour the robot exhibited due to this last procedure was quite ineffective.

Three new experiments were made with the robot starting off with this human crafted behaviour, each one of them corresponding to one of the experiments reported previously: no learning, emotion-dependent learning and sensation-dependent learning. Figures 4.10, 4.11 and 4.12 show the results.

Without learning, this behaviour would end up receiving a mean emotion-dependent reinforcement of 0.54 and a mean sensation-dependent reinforcement of 0.30. When the robot was allowed to learn either with sensation-dependent or emotion-dependent reinforcement, it maintained the initial behaviour and kept similar reinforcement values, apart from a small increase in variance due to the exploration characteristic of learning.

In the experiments reported previously (Figures 4.6 and 4.8), the robots learning from emotion-dependent and sensation-dependent reinforcement would sometimes reach reinforcement levels similar to these. However, the algorithm does not always manage to

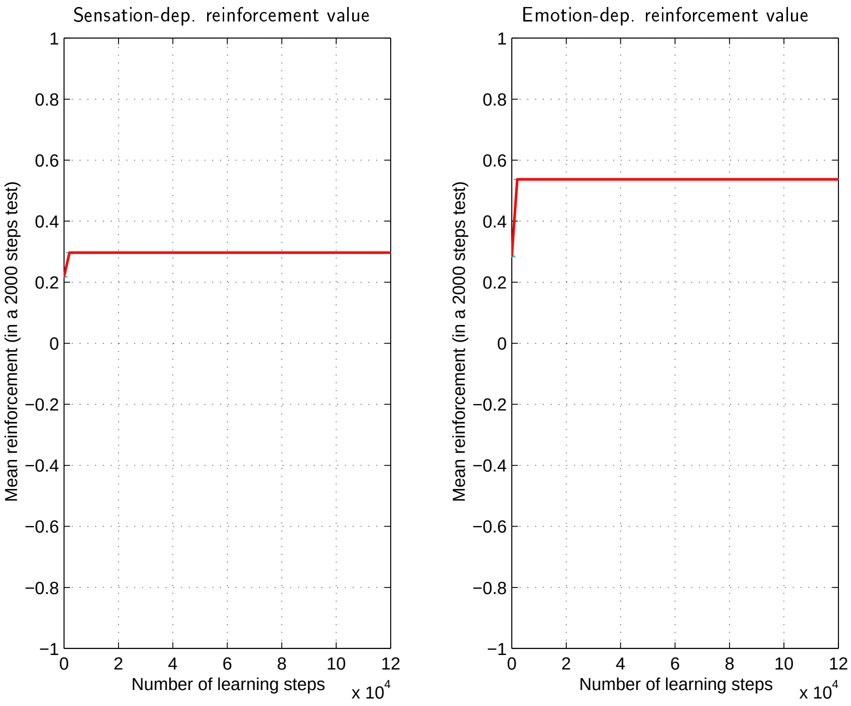


Figure 4.10: Hand-crafted controller with no learning.

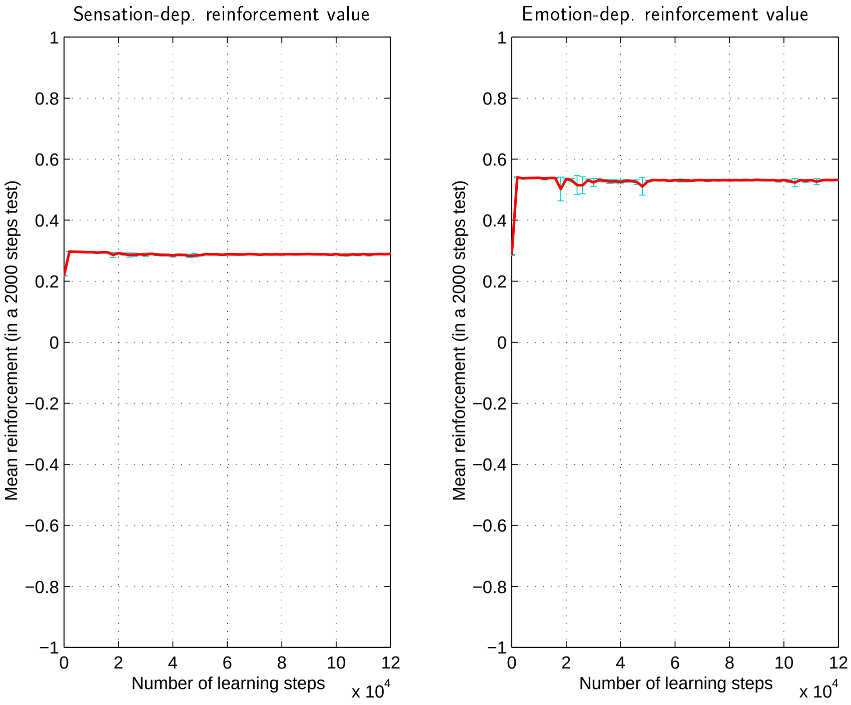


Figure 4.11: Hand-crafted controller learning with emotion-dependent reinforcement (right graph). The sensation-dependent values (left graph) are for comparison.

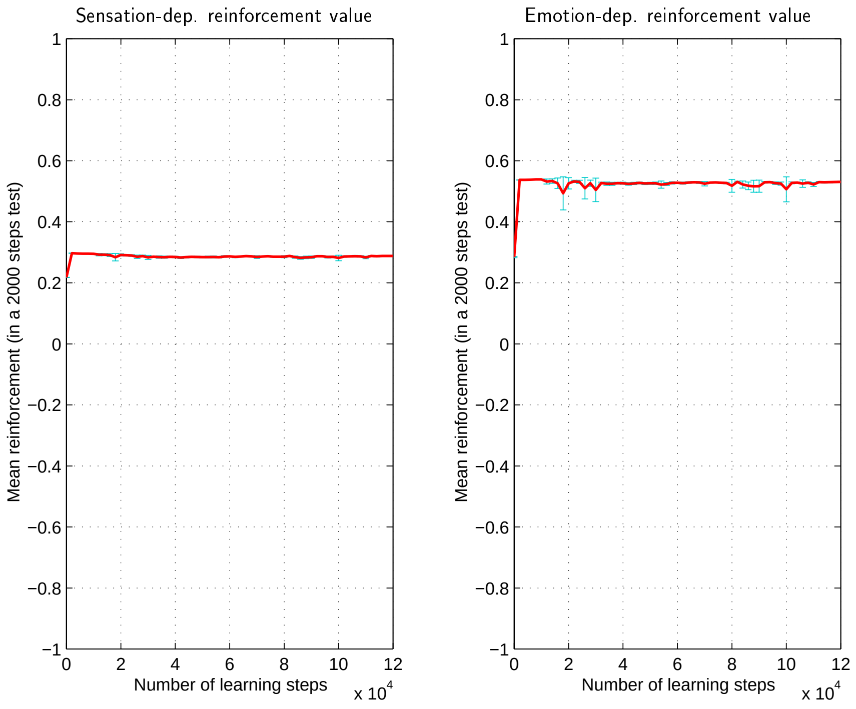


Figure 4.12: Hand-crafted controller learning with sensation-dependent reinforcement(left graph). The emotion-dependent values (right graph) are for comparison.

converge to such good solutions and in the case of some previous emotion-dependent reinforcement trials it converged to receiving very bad reinforcements. It should be noticed that the exact behaviour that was designed cannot be achieved with the set of actions normally used. Nevertheless, slightly more sophisticated behaviours were learned in these previous experiments that achieved the same kind of reinforcement for long periods of time. These learned behaviours were less stable, because it is more difficult for the controller to keep track of a light with an action set with non-invertible actions. This might suggest that if the robot was equipped with the new set of actions then its performance in the learning task would improve. However, experiments showed the opposite (Figures 4.13, 4.14 and 4.15 show the results of using the new set of actions with the first three controllers of this section). The reason for this worse performance is probably the fact that it is easier to avoid further encounters with an obstacle that appears in front of the vehicle if the backward movement is not done in a straight line. The results with this new set of actions agree, however, with those previously obtained in that emotions provide an inadequate reinforcement signal for this task.

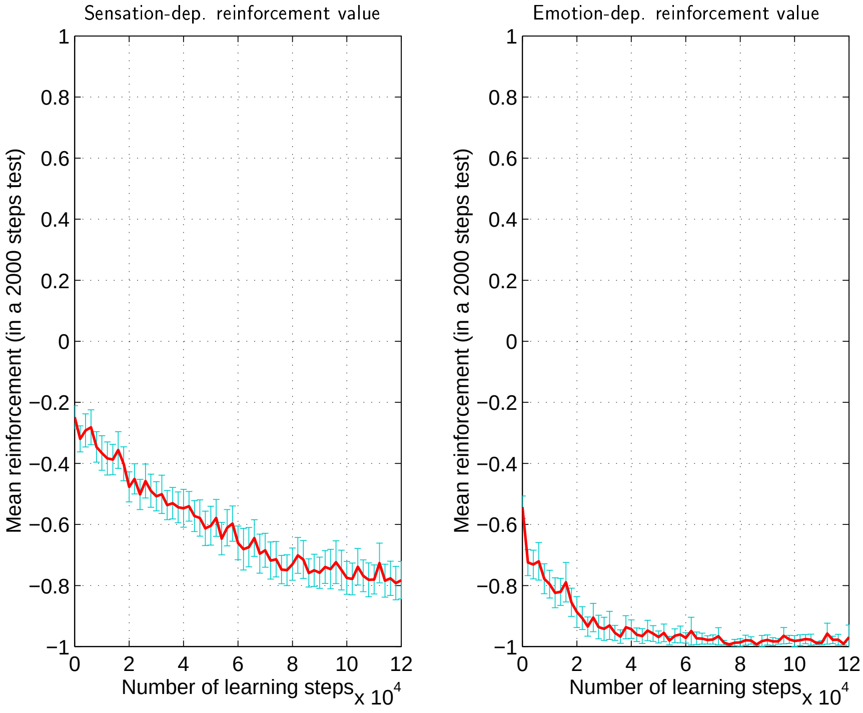


Figure 4.13: Random controller employing the second set of actions.

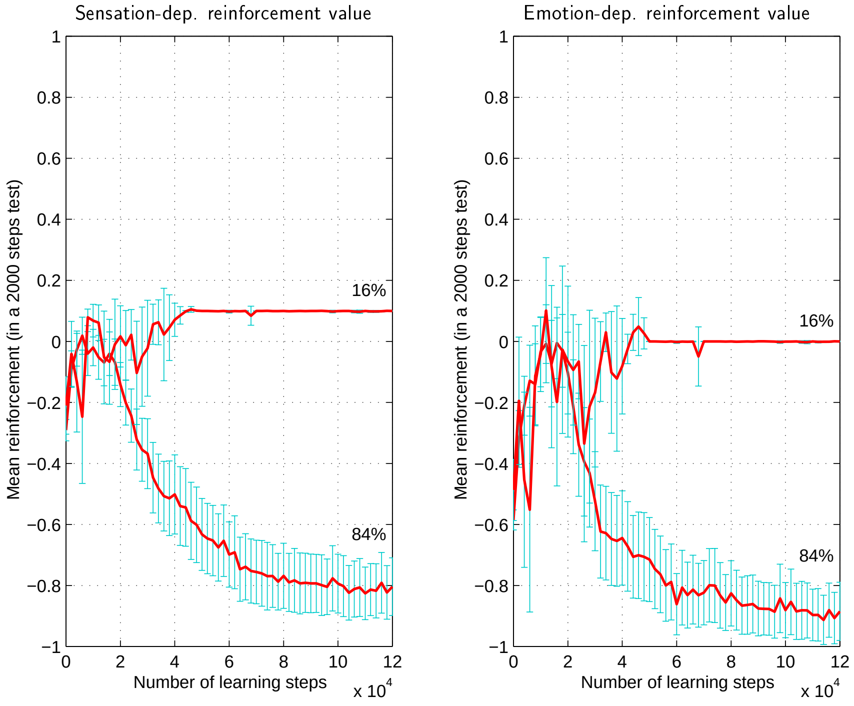


Figure 4.14: Emotion-driven controller employing the second set of actions. The reinforcement received by the controller is on the right graph and the sensation-dependent values (left graph) are for comparison.

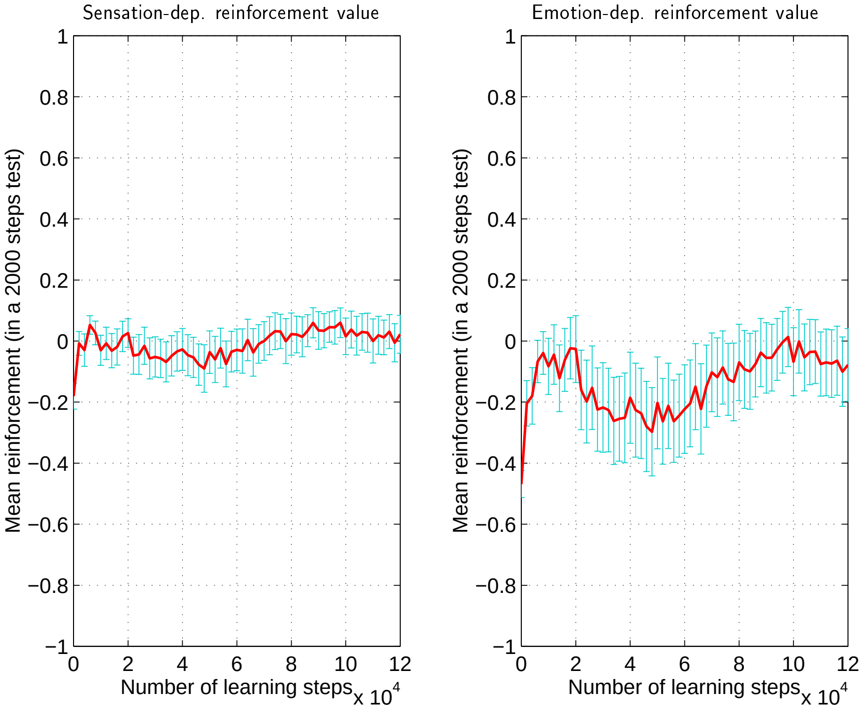


Figure 4.15: Sensation-driven controller employing the second set of actions. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

4.4 Further Experiments for Analysis of Results

Given the results presented, one has to conclude that the emotions were quite unsuccessful in providing a good reinforcement value, but still the question remains of whether its failure was not due to some hidden experimental feature. A number of possible causes were investigated experimentally. Alongside, two other issues were also explored: emotion influence on perception and learning during evaluations. The results obtained are presented next.

4.4.1 Reinforcement dependent upon rate of change

It was noticed previously that one of the problems of emotions might be that they continue giving a negative (or positive) reinforcement even when the situation is improving (or deteriorating). An attempt to minimise this problem was to have the reinforcement value be the difference between the previous and the current emotional value. In other words, the new reinforcement value (R') would reflect the improvement or worsening

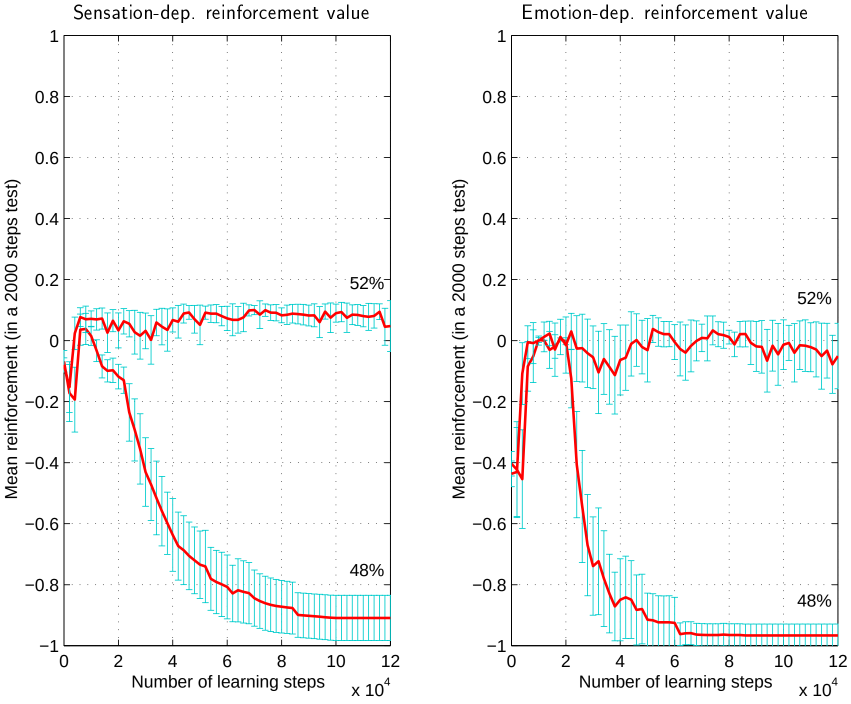


Figure 4.16: Emotion-driven controller with rate-of-change-dependent reinforcement. The reinforcement received by the controller is on the right graph and the sensation-dependent values (left graph) are for comparison.

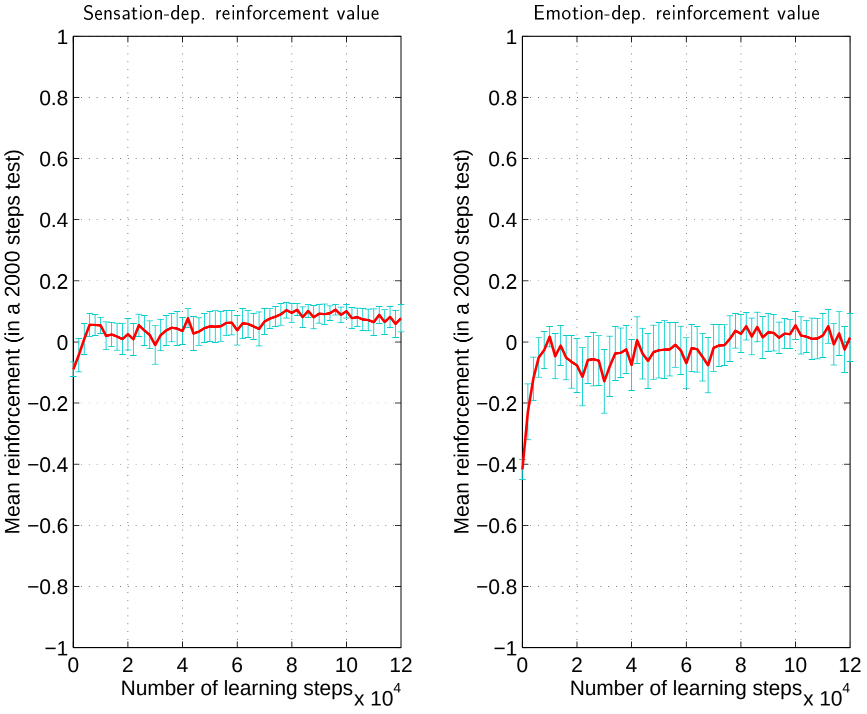


Figure 4.17: Sensation-driven controller with rate-of-change-dependent reinforcement. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

of the robot's situation.

$$R'_n = R_n - R_{n-1} \tag{4.8}$$

The value of this kind of reinforcement is in general smaller; for this reason, the selection temperature used by the Action Selection Module was decreased to a more suitable value of 0.02.

Figures 4.16 and 4.17 show the results for using rate of change for both the emotion-dependent and sensation-dependent reinforcements. Once again, the robot performs much better with the sensation-dependent reinforcement. The use of rate of change proved to be unsuccessful in improving learning with emotion-dependent reinforcement.

4.4.2 Emotion selection threshold in sensation-dependent reinforcement

The main unexplored difference between sensation-dependent and emotion-dependent reinforcement is the selection threshold (I_{th_s}) used in process of selecting a dominant emotion. If the intensity of the emotions is too small, then there will be no dominant emotion selected and the emotion-dependent reinforcement will be zero. This thresholding implies that the emotion-dependent reinforcement provides less information than the sensation-dependent reinforcement, because small emotion intensity values are discarded and replaced by zero. The selection threshold used in the previous experiments was 0.2; an experiment was run applying this same threshold to sensation-dependent reinforcement. No significant differences arise from the use of the threshold in the sensation-dependent reinforcement — the result graphs (see Figure 4.18) are similar to those previously obtained (see Figure 4.8). Apparently, the Q-learning mechanism seems to solve the threshold-added difficulty easily.

The results of this experiment show that the selection threshold by itself is not responsible for the low performance of the emotion-driven controller.

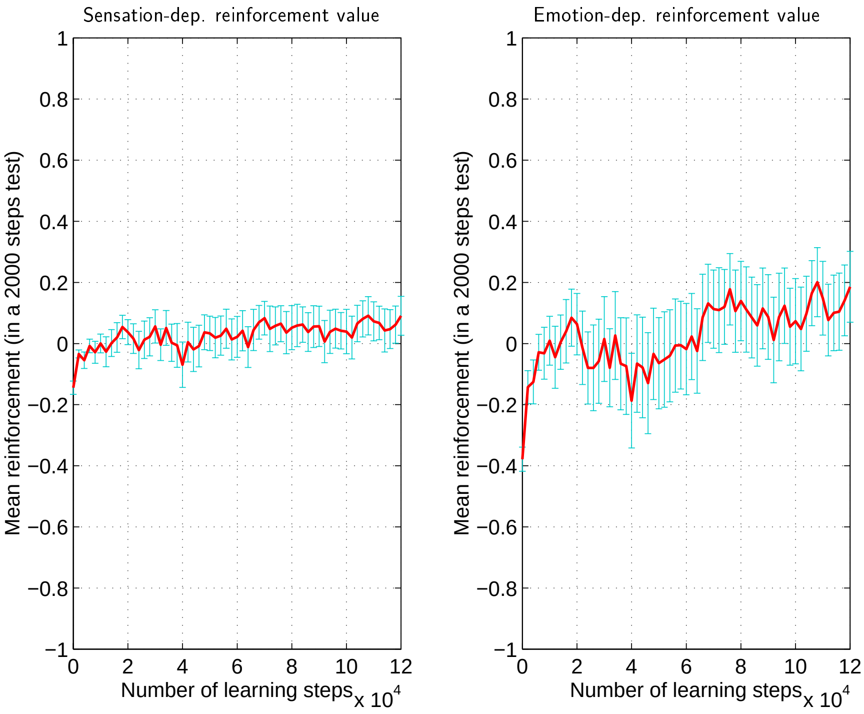


Figure 4.18: Sensation-driven controller with reinforcement subject to thresholding. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

4.4.3 Simple action selection

In the basic controller used in the experiments, extensions were made to the “vanilla” Q-learning algorithm in terms of the action selection mechanism. This algorithm is usually associated with an action selection at every step, but in order to have the current action changed less often, some mechanisms were added that prevent an action selection at every step by maintaining the current action. This could also have influenced the performance of the emotion-dependent reinforcement controllers.

A new set of experiments was made with traditional action selection at every step. The main differences found in the new results when compared with the previous ones are the following:

- as expected, the robot’s final behaviour is much more *hesitant*, *i.e.* less able to keep an action for a meaningful amount of time;

- new types of behaviours emerged (*e.g.* *circling* near a light using the fast-forward action in conjunction with other actions);

- the robot is more successful at avoiding collisions;
- in general, the area covered by the robot is smaller;
- energy maintenance improved substantially in the case of emotion-dependent reinforcement, and declined slightly in the case of the sensation-dependent reinforcement;
- the results of the experiment with no learning develop an increase in reinforcement, before the reinforcement begins to drop due to hunger (it was found that with a random action selection the temperature sensation has a strong tendency to rise, which makes the robot HAPPIER);
- the experiment with sensation-dependent reinforcement received similar reinforcements.
- the experiment with emotion-dependent reinforcement received much higher reinforcements, although it still performed considerably worse than the experiment with sensation-dependent reinforcement.

The use of a simpler action selection mechanism has advantages and disadvantages. Although the task becomes easier to learn (probably due to the reduction in bumping), the final behaviour is not very impressive.

These results might suggest that the problems of the "vanilla" controller could be easily overcome if the action selection temperature were lowered, yet this is not the case. The experiment was repeated with a temperature of 0.07 instead of the usual 0.1 and the results in terms of reinforcements were worse.

4.4.4 Different networks

Contrary to what might be suggested by the poor results, the networks used are actually somewhat over-complex for the task in hand, because using only one neuron in the hidden layer showed similar results (see Figure 4.19).

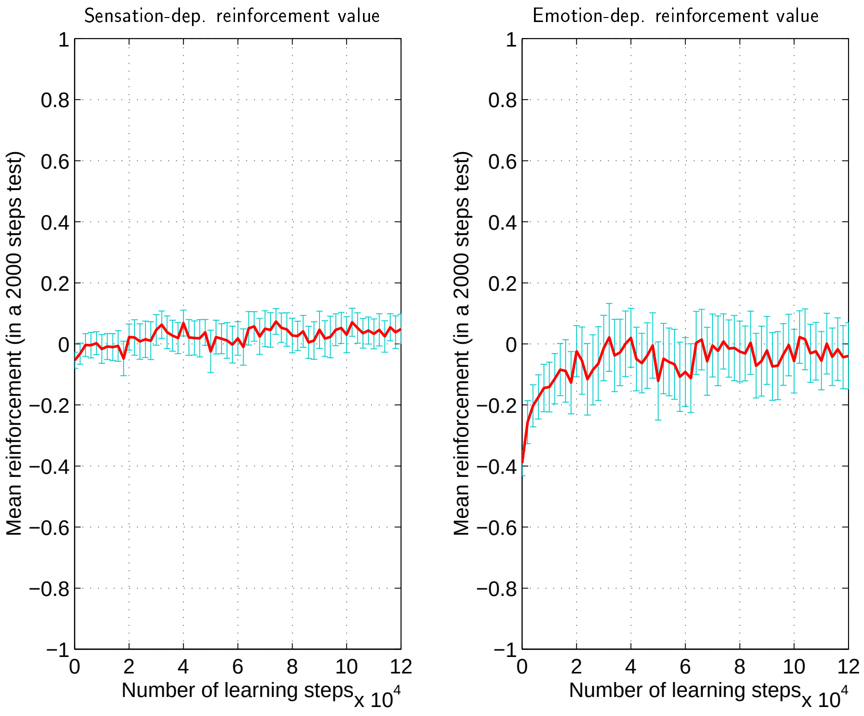


Figure 4.19: Sensation-driven controller using networks with only one hidden unit. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

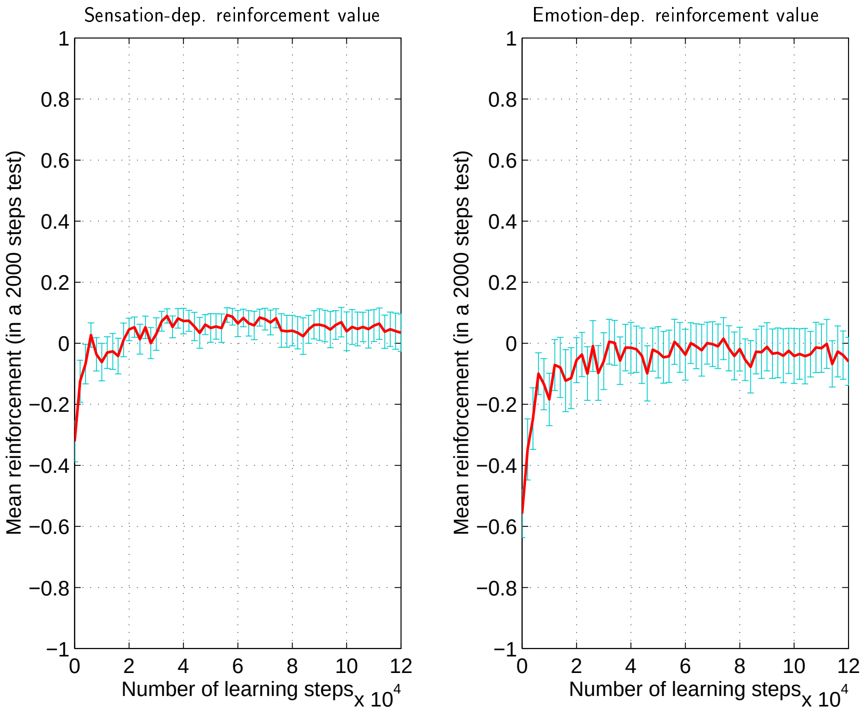


Figure 4.20: Sensation-driven controller using networks with initial random values in all weights. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

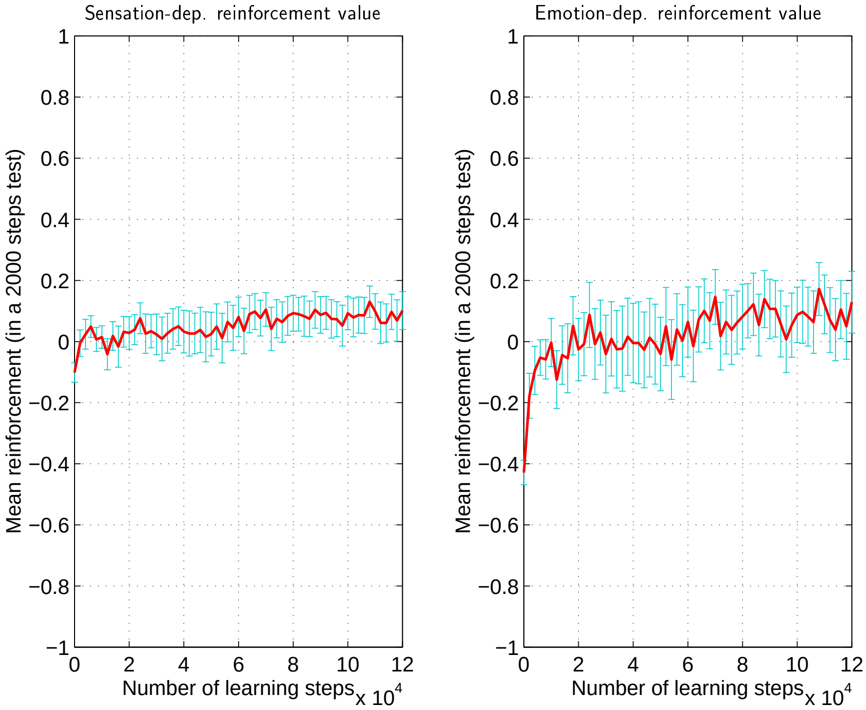


Figure 4.21: Sensation-driven controller with faithful perception. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

In opposition to traditional experiments with learning neural-networks, the weights between the input layer and the hidden layer were initially set to zero instead of randomised. Not using random initialisations for all the network weights might seem strange and prone to failure, but results of experiments that use random initialisations of all weights show that this has not negatively influenced the learning performance (compare Figures 4.20 and 4.8).

4.4.5 Non-emotional perception

The value of the feelings given as input to the neural networks are not the robot's raw sensations, but are influenced by the emotions through the hormone system: the robot has a false image of its body sensations. Does this influence the learning task? A brief analysis of the networks' weights showed that the feelings had an active role in influencing the controllers' preferences in the selection of actions.

Figure 4.21 shows the results for a learning experiment where the controller neural networks (see Figure 4.4) input sensations (S_f) instead of feelings (I_f). There were no significant changes in the reinforcement rewards.

Even if the robot learns with the networks' feelings inputs totally removed and only the sensor readings' inputs are kept (*i.e.* the feelings' input values are replaced by a constant number) the results are still similar to those previously reported: performance for both sensation-dependent and emotion-dependent reinforcement does not suffer any significant change.

4.4.6 Learning during tests

The robot's controller is designed to learn continuously, yet learning is turned off during the evaluation period[9]. The reason for this is simply to have an instantaneous evaluation of the same controller in two thousand different scenarios provided by the steps of the evaluation period. The problem is that these are not arbitrarily chosen random scenarios, but are the scenarios consecutively reached by the robot due to its action selections. Therefore, a test's individual evaluations are not always a fair sample of the evaluations the robot can get. The behaviour of the robot in the earlier steps of the evaluation period will bias its evaluation in later steps. Extreme behaviours may cause long runs of good or bad reinforcement. For instance, if the controller is not aware of certain features of the environment it might repeatedly perform a misplaced action that will keep it in this situation and the evaluation will be biased. For example, it might be running into a wall for the whole of a test. If the space localisation of each individual evaluation were made independent, the final evaluation would be improved by evaluations done in different places in space (*e.g.* near a light).

The main problem with any testing approach for this experiment is that each evaluation can not be dissociated from the robot's previous experiences and it is intimately related with the previous step situation both in terms of the robot's spatial location and internal state. There are few alternatives for how the test may be done, because the robot's location and internal state cannot be arbitrarily chosen. These are always the result of the robot's history.

[9] For details on the experimental procedure consult Section 4.2.4.

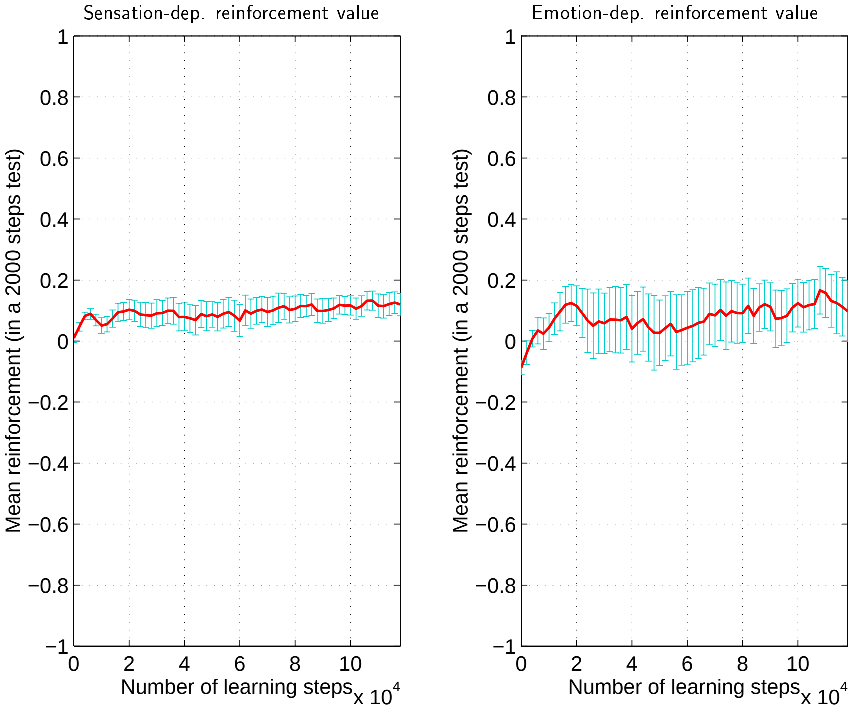


Figure 4.22: Sensation-driven controller learning during tests. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

One possible alternative would be to have several smaller learning test periods instead of one, at each testing phase. After each one of these test periods the robot state would be restored to the state prior to the test phase. Executing several tests starting from the same point would result in different evaluations due to the randomness of the controller. However, following this evaluation method would result in a significant increase in the complexity of the evaluation process that would probably not be justified by the evaluation improvements obtained.

A more radical alternative would be to allow learning during the test period. The previously described test problems would not happen if the robot were allowed to learn, because the robot would learn to go away from the obstacles. This alternative is also not devoid of faults. If the two thousand step test is kept, then it will not be same controller under test during the period, but a controller that is constantly changing. Furthermore, it will be impossible to differentiate what the controller has actually learned from what it learns and unlearns as situations change. However, this is probably a more adequate evaluation procedure, because the robot is supposed to learn continuously anyway; and possibly the evaluation procedure in use is actually providing a worse evaluation than the one deserved by the robot's performance.

To test whether performance would be better, an experiment was run that employed learning while evaluating the robot's performance. Results are shown in Figure 4.22. The robot seems to perform a bit better, although not significantly better, when learning all the time. It is also interesting to notice that the variance of the sensation-dependent reinforcements is much smaller, which can be easily explained by the evaluations not favouring extremes as much as before.

4.5 Summary

Table 4.2 presents a summary of the results obtained in the experiments carried out with the four different controllers both in terms of emotion-dependent and sensation-dependent reinforcement value. Results show that emotions were unsuccessful in providing a competitive reinforcement function when compared with a more traditional reinforcement function based on sensations. Although not presented here, results consistent with this finding were obtained even with slightly different emotion models (an example is given in Appendix C.5). The main difference between the reinforcement functions, and the only identified cause for the emotions' failure, was the existence of recurrent and lateral influences in the emotions model.

In Section 4.4, several other causes were tested, but no other suitable explanations were

Controllers		Section	Figure	Emotion-dependent reinforcement	Sensation-dependent reinforcement
Emotion-driven	group 1	§4.3.1	4.6	-0.04 ± 0.10	0.04 ± 0.06
	group 2	§4.3.1	4.6	-0.91 ± 0.07	-0.78 ± 0.12
Sensation-driven		§4.3.2	4.8	0.07 ± 0.09	0.07 ± 0.05
Random		§4.3.3	4.9	-0.80 ± 0.07	-0.32 ± 0.06
Hand-crafted	no learning	§4.3.4	4.10	0.54 ± 0.00	0.30 ± 0.00
	with learning	§4.3.4	4.11 & 4.12	0.53 ± 0.01	0.29 ± 0.00

Table 4.2: Comparison of the emotion-driven, the sensation-driven, the random and the hand-crafted controllers. The means of the emotion-dependent and sensation-dependent reinforcement values and their 95% confidence interval obtained in the last ten testing points of the trials are presented.

found. In each case, no significant change in results was observed when the possible cause was eliminated or controlled for. In particular:

- The use of rate-of-change-dependent reinforcement instead of absolute value reinforcement does not affect results — emotion-dependent reinforcement still performs worse than sensation-dependent reinforcement.
- The emotion selection threshold is not responsible for the differences in performance between the emotion-dependent and the sensation-dependent reinforcement. The same threshold can be applied to the sensation-dependent reinforcement with no detrimental effect.
- The use of traditional action selection at every step produces equivalent results. The main differences found in those results when compared with the ones shown here are that it is easier for the agent to reach higher reinforcement values, but the final behaviour observed by external visual inspection is less impressive. As expected, the robot's final behaviour is much more *hesitant* and in general, the area covered by the robot is smaller. However, the robot seems more successful at avoiding collisions which probably makes the learning task easier.
- Networks with different numbers of hidden units and networks with all their weights initialised with random values were also tested and found to make no significant difference.

Another issue relevant to the model used is whether using feelings or sensations for the robot's perception makes a difference in terms of its final performance. In general, the values given as input to the neural networks are not the robot's raw sensations, but feelings that are influenced by the emotions through the hormone system. In the particular task tested, the influence of emotions on perception is unnoticeable in terms of final behaviour.

As a side issue, a different evaluation mechanism was also attempted. The question of whether the robot should be allowed to learn while being evaluated was raised. Arguments for and against such procedure were presented. In practice, results show that there is not much difference in terms of final evaluation whether the robot is learning or not. This result will be used in next chapter to reduce the experiments' computational effort by evaluating the robot while it is learning its task.

4.6 Exploring Alternative Uses of Emotions

The experiments reported previously have repeatedly shown that using emotions as reinforcement in the present controller is detrimental. However, other uses of emotions might be more fruitful. The fact that emotions performed poorly as reinforcement value should not discourage their use in artificial systems: the importance of emotions in human reasoning is widely acknowledged and there are many other possible roles for them that should be considered.

In this section two alternative uses are suggested and tested. In both cases, emotions modulate the learning process instead of directly attributing value to the situations the robot experiences[10].

4.6.1 Emotions modulating learning rate

Human learning abilities are strongly dependent on the person's emotional state. For instance, strong emotions often give rise to vivid memories, while lack of emotion is often associated with disinterest and difficulties in learning (Schwartz and Reisberg, 1991). Along this line, experiments were made using emotions to modulate the learning rate. In these experiments, the robot would have a higher learning rate — directly proportional to the intensity of the current dominant emotion — if under a dominant emotion, be this positive or negative, than if there were no dominant emotion.

The use of reinforcement value to modulate learning rate is not new in the domain of robotic research, experiments using a similar modulation have been reported that showed an increase of learning performance in terms of robustness (Verschure et al., 1995).

This experiment explores the use of emotions as a sort of learning gain. This learning gain ($\mathcal{G}$) is used to influence the learning rate (η) in the following way:

[10] Sensation-dependent reinforcement will be used in these experiments.

$$\eta = 2\,\mathcal{G}(\text{Emotion})\,\eta_{default} \tag{4.9}$$

To obtain the normal results with fixed learning rate, we should have a constant learning gain ($\mathcal{G}$) of 0.5.

First of all and before starting the actual experiments, different values of learning rates were tried out. Figure 4.23 shows the results, including the use of the learning rate default value of 0.1 used by all previous experiments (same as in Figure 4.8).

Next some experiments were done using a variable learning rate dependent on the robot's emotional state. In these experiments (results in Figures 4.24 and 4.25) the learning gain was set to the intensity of the present emotion which varies between the selection threshold and 1, or zero if there were no dominant emotion, *i.e.*:

$$\mathcal{G}(Emotion) = \begin{cases} I_e & \text{if there is is a dominant emotion } e \\ 0 & \text{otherwise} \end{cases} \tag{4.10}$$

The learning rate itself was calculated through Equation 4.9 and took values between 0.04 and 0.2 or was zero. This means that if a dominant emotion is not present the robot will not learn at all. It is quite surprising how the robot still manages to maintain its high rate of success in the learning task (although the learning is a bit slower) if one takes into consideration that neutral states are not being learned at all by the neural networks. If the weights of all network layers are initialised with random values (see Figure 4.25), some (64%) of the robots present severe difficulties in learning (although they do manage to learn in the end). This is the result of the initial preferences of the system not being neutral. If these initial preferences happen to favour the wrong kind of actions, the robot will first have to unlearn these. Since the robot is not learning all the time, this can result in a high performance cost.

It is possible to conclude that the robot does not need to waste computing time on learning if there is no emotion present. This will not compromise its performance, unless the robot's initial preferences are very misleading. Even in this case, the drop in performance is only temporary.

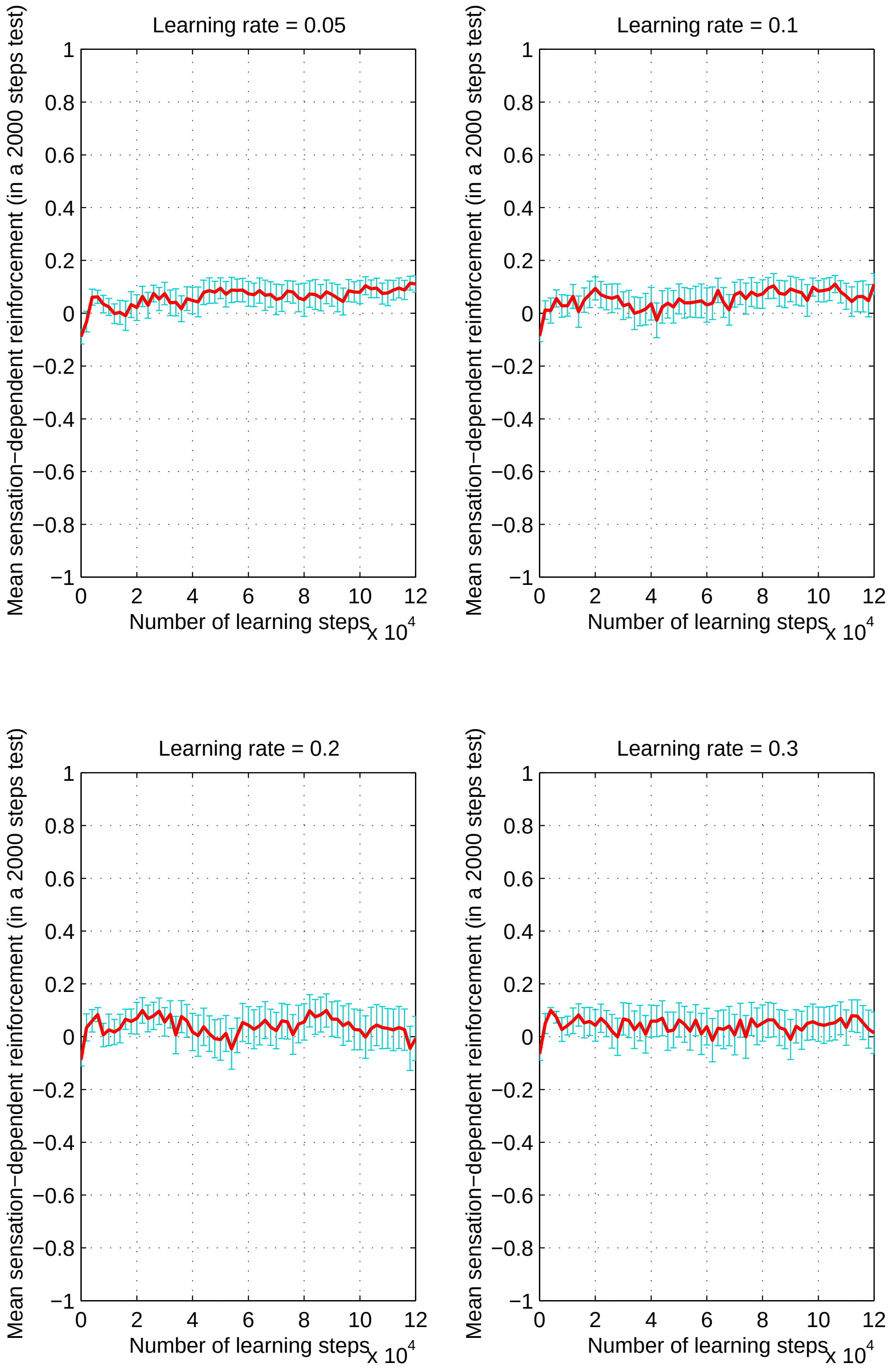


Figure 4.23: Different learning rates.

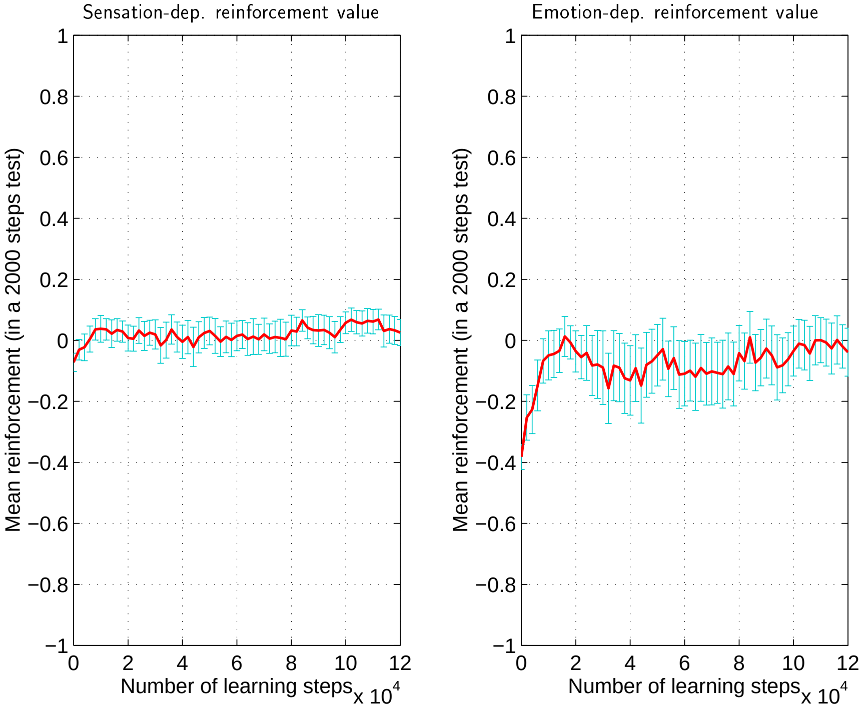


Figure 4.24: Learning rate dependent on current emotion. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

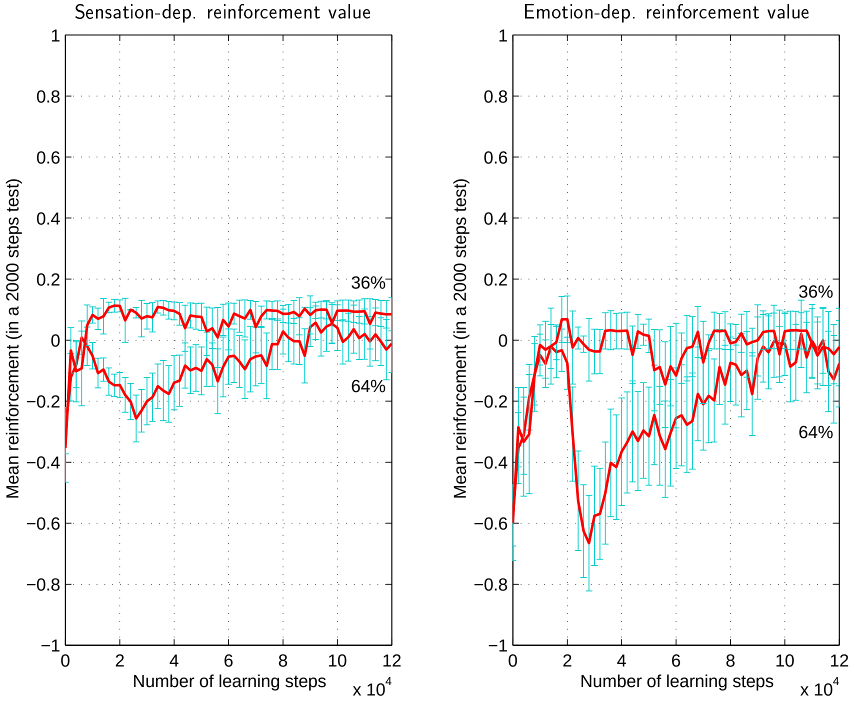


Figure 4.25: Learning rate dependent on current emotion. Random initial weights. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

4.6.2 Variable emotion-dependent selection temperature

Emotions were also used to modulate the action selection temperature with success. In these experiments, the fixed exploration versus exploitation ratio was upgraded to a more sophisticated selection algorithm that takes into consideration the deadlock situations in which the robot gets trapped. In these situations, there is an option that is by far better ranked than the others and therefore always gets selected although its practical utility turns out to be very low and it is not able to change the situation at all. The solution used to circumvent this problem was increasing the selection temperature when the robot was in a negative emotional state and thus triggering more exploration than usual.

To begin with, some experiments were run to discover how well the robot would perform with different fixed temperatures. See Figure 4.26 for results. Previous experiments used an action selection temperature of 0.1. Consult Figure 4.8 for comparison.

Next, emotions were used to modulate the exploration versus exploitation ratio, by directly influencing the temperature parameter of the action selection module. Two emotion-dependent functions ($\mathcal{F}_1$ and $\mathcal{F}_2$) were designed to yield values in the range 0.05 and 0.25. This is a suitable selection temperature range because it includes values for which the learning controller performs well, but is slightly extended towards the upper bound to allow more exploration in action selection.

$$\mathcal{F}_1(Emotion) = \begin{cases} I_e/4.0 & \text{if there is is a dominant emotion } e \\ 0.05 & \text{otherwise} \end{cases} \tag{4.11}$$

$$\mathcal{F}_2(Emotion) = \begin{cases} I_e/4.0 & \text{if there is is a negative dominant emotion } e \\ 0.05 & \text{otherwise} \end{cases} \tag{4.12}$$

The results for using the function $\mathcal{F}_1$ and $\mathcal{F}_2$ to determine selection temperature (T) are presented in Figures 4.27 and 4.28, respectively.

Apparently, the results for function $\mathcal{F}_2$ (see Figure 4.28) are an improvement over the results previously achieved. The use of this function allows the robot to explore new solutions when it is in a bad situation. The function $\mathcal{F}_1$, increases the temperature independently of whether the robot's emotional state is positive or negative. Disrupting

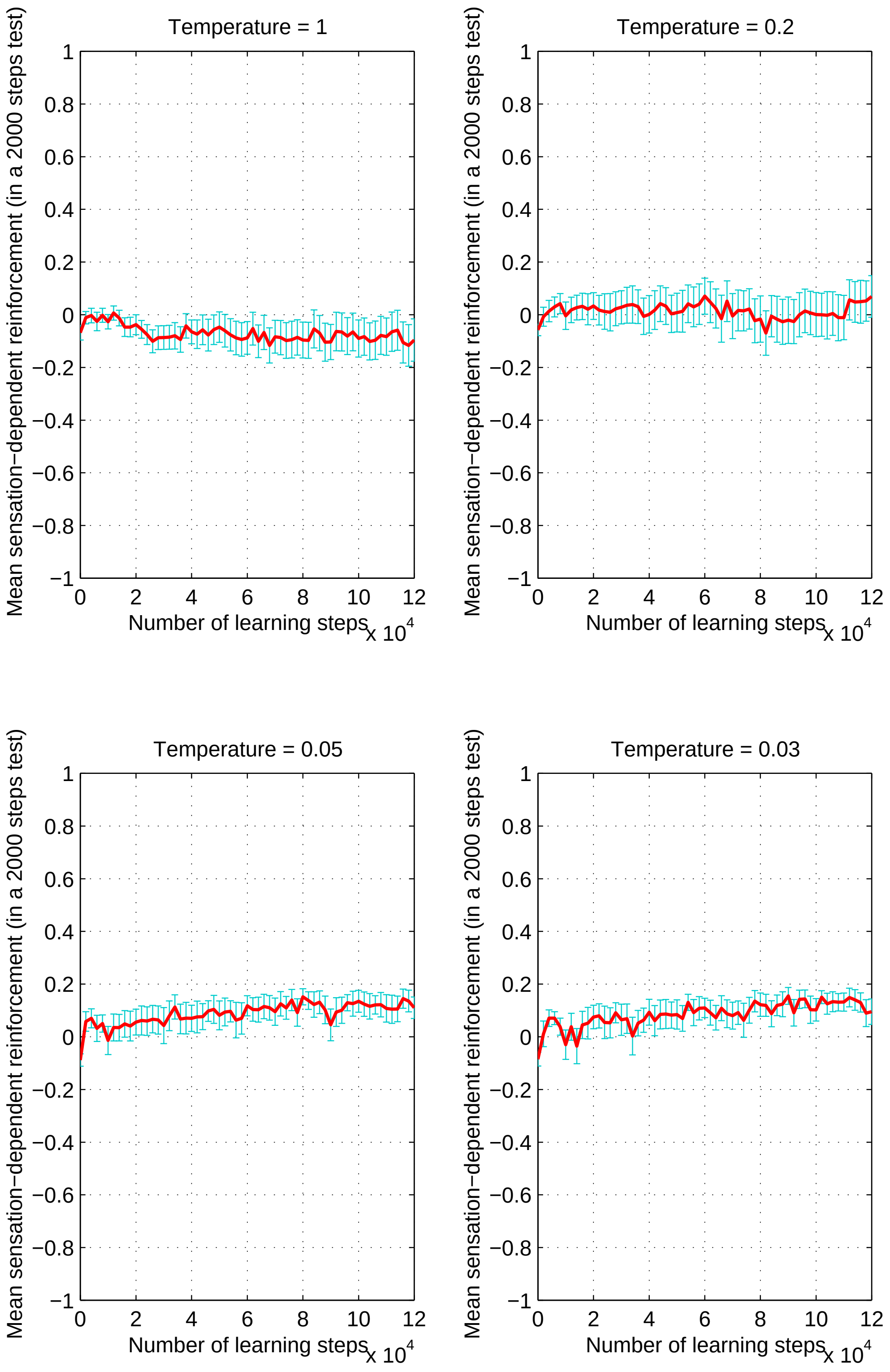


Figure 4.26: Different action selection temperatures.

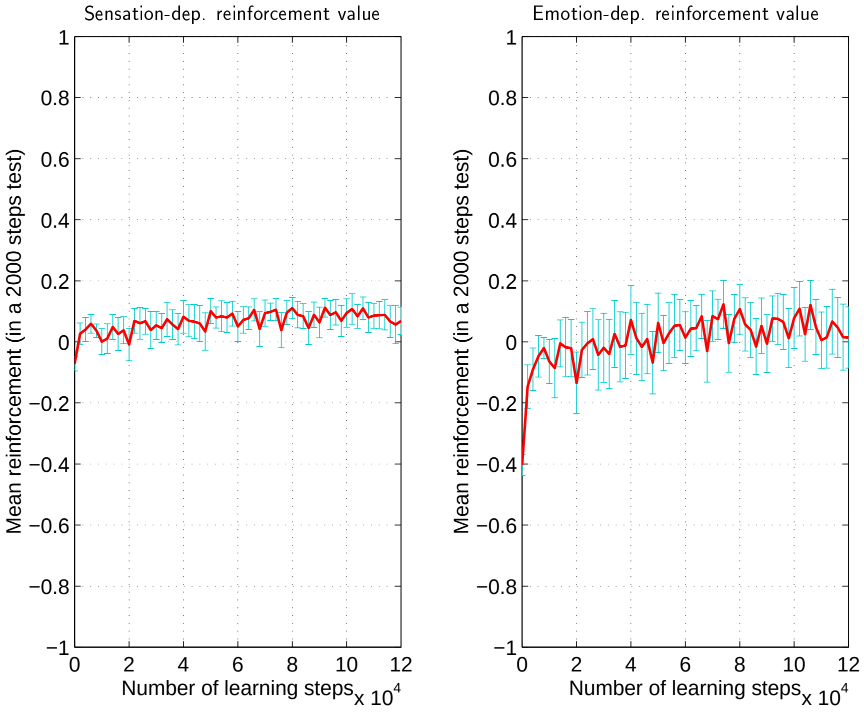


Figure 4.27: Selection temperature influenced by emotions: $\mathcal{F}_1$. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

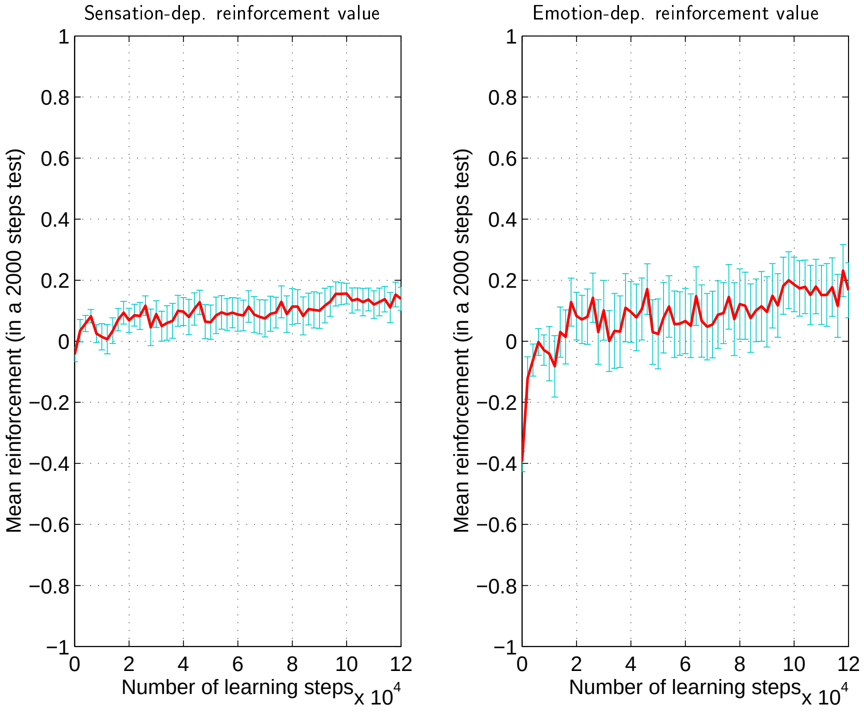


Figure 4.28: Selection temperature influenced by emotions: $\mathcal{F}_2$. The reinforcement received by the controller is on the left graph and the emotion-dependent values (right graph) are for comparison.

behaviour that is being successful does not seem to be such a good idea, although no significant changes can be found in using $\mathcal{F}_1$ (see Figure 4.27) when compared with the standard experiment (Figure 4.8). Table 4.3 summarises the results.

Sensation-driven controllers	Figure	Emotion-dependent reinforcement	Sensation-dependent reinforcement
Selection temperature = 0.1	4.8	0.07 ± 0.09	0.07 ± 0.05
Selection temperature = $\mathcal{F}_1$	4.27	0.05 ± 0.09	0.08 ± 0.05
Selection temperature = $\mathcal{F}_2$	4.28	0.17 ± 0.09	0.13 ± 0.04

Table 4.3: Comparison of experiments with different selection temperature. The table presents the means of reinforcement values and 95% confidence interval in the last ten testing points of each experiment.

4.7 Conclusions

Unfortunately, the experiments reported in this chapter failed to show that emotions can be used for reinforcement in robot learning. More importantly, the results do show that the role of emotions is more intricate than often assumed and that a simple approach to the use of emotions as context judgement values suitable for direct use as reinforcement is not very successful when a more than usually realistic emotion model is used. This suggests that more attention should be given to the role attributed to emotions in adaptation.

The emotions do not really provide a good evaluation of what is going on at any one moment, but are a sort of mixed evaluation the robot has acquired from its past experiences. This may be good for modulating its behaviour, but should not be taken at face value when trying to predict the outcome of each one of its primitive actions.

In Section 4.6 preliminary results were presented that suggest that emotions can be successfully used in modulating the learning rate and the exploration versus exploitation ratio. Although these alternative approaches to the use of emotions appeared to be successful in improving performance, no solid conclusions could be drawn with the data obtained, the problem being that the simpler sensation-dependent controllers are already quite good at their task, making it difficult to demonstrate clearly any significant improvement provided by more sophisticated controllers. For a thorough examination of these mechanisms the complexity of the robot's task must be increased, and that is the topic of the next chapter.

Chapter 5

Behaviour-Based Control

5.1 Introduction

The task described in the previous chapter was extended to provide a more challenging robot-environment interaction. This prompted an upgrade of the basic learning controller. The new controller is based on the action-based one described in Section 4.2.3, but the actions were replaced by behaviours that add extra competence to the controller. As is demonstrated in this chapter, the higher level of abstraction of this controller makes it more suitable for the use of emotions.

Once more, emotions were used to influence control, but this time with more success. Three possible forms of emotional influence were examined:

Control triggering — One of the most difficult problems faced when employing reinforcement learning techniques in robotics applications is to determine when a discrete state transition occurs. This transition can be triggered by some internal or external event and must be identified by the designer, because it determines when the controller needs to re-evaluate its previous decision and make a new one. An incorrect state transition design can be fatal to the success of the learning agent. In fact, this was the reason why it was found necessary to tackle this problem first.

Experiments were done to test whether emotions can successfully fulfill the role of determining state transitions. In practice, the learning controller was triggered whenever:

- there was a change of dominant emotion;
- the current dominant emotion intensity value was statistically different from the values recorded since a state transition was last made.

Reinforcement — In the initial experiments with behaviour-based control, emotion-dependent reinforcement was set aside and sensation-dependent reinforcement used instead[1]. The poor results obtained with the earlier task suggested that emotion-dependent reinforcement might compromise the experiments' results. Once the event-detection mechanism was settled, emotions were tested again as source of reinforcement.

Perception — Another mechanism re-evaluated was the influence of emotions on the robot's perception. When the robot learns associations between states and rewards through its neural networks, it is using feelings to represent state by using the feelings as network inputs. In the emotion model developed, feelings are influenced by emotions through the hormone system. So the represented robot state is emotion-dependent: the state which the robot learns to associate with rewards is actually being biased by emotions. It is being changed to be more compatible with the active emotions, thus making the relevant features of the environment more salient because those are usually the ones associated with emotional value. The question is, what is the impact of this on the robot's final performance.

To start with, the next section provides a detailed description of the testbed used for the experiments. A description is given of the extended task, emotion system and learning controller, stressing the differences from the previous experimental setup. In addition, this chapter's experiments benefited from a more elaborate experimental procedure that is also described within that section. Three sections follow, each reporting on experiments that explore one of the roles of emotion mentioned above. This is followed by a global analysis of the final emotional controller in Section 5.6, and some conclusions on the overall accomplishments of emotions in the last section of the chapter. Some of the specific implementation details of the experiments are relegated to Appendix D.

[1] Sections 4.3.1 and 4.3.2 provide a full description of each of these reinforcement functions.

5.2 Experimental Setup

5.2.1 Robot, environment and task

As before, the robot's task consists in collecting energy from food sources scattered throughout the environment, but the survival problem was made more difficult by making energy harder to obtain. This was accomplished by two different means:

- *The robot has to perform elaborate behaviour to receive energy.* To gain energy from a food source, the robot has to bump into it. This will make energy available for a short period of time. At the same time an odour will be released that can be sensed by the robot. It is important that the agent is able to discriminate this state through its sensors, because the agent can only get energy during this period. This energy is obtained by receiving high values of light in its rear light sensors, which means that the robot must quickly turn its back to the food source as soon as it senses that energy is available. To receive further energy the robot has to restart the whole process by hitting the light again so that a new time window of released energy is started.
- *The robot can only extract a limited amount of energy from each food source.* A food source can only release energy a few times before it is exhausted. In time, the food source will recover its ability to provide energy again, but meanwhile the robot is forced to look for other sources of energy in order to survive. The robot cannot be successful by relying on a single food source for energy, *i.e.* the time it takes for new energy to be available in a single food source is longer than the time it takes for the robot to use it.

When a food source has no energy, the light associated with it is turned off. This was done in order to avoid the robot staying around the same food source, even when that source has no more energy left. When the light is turned off, the food source becomes an obstacle like any other and the robot can look for a new food source with its light sensors again. A light is on when it has energy available to release and during the periods it is releasing energy. This last point is important, because otherwise the robot would not be able to extract the energy through its light sensors. Although it was felt necessary to use the mechanism of turning the lights off in the experiments to make the task a bit easier for the robot, experiments *a posteriori* with emotion-dependent event detection proved this mechanism superfluous. The robot would exhibit a slightly worse performance, but still managed to successfully learn the task.

The task can be translated into multiple goals: moving around the environment in order to find different food sources and, if a food source is found, extracting energy from it. Furthermore, the robot should not keep still in the same place for long durations of time or collide with obstacles.

All the experiments were carried out with the same Khepera simulated robot, but placed in the environment shown in Figure 5.2. There are a few exceptions in which the environment pictured in Figure 5.1 is used instead. This is a more demanding environment that is used to distinguish between controllers that exhibit similar performances in the normal environment. The new environments are more corridor-like than the previous, allowing the robot to travel from one light to another by wall following. The length of the corridors an agent must travel to go from one light to another measures the difficulty of the environment, because longer corridors demand more persistence from the robot.

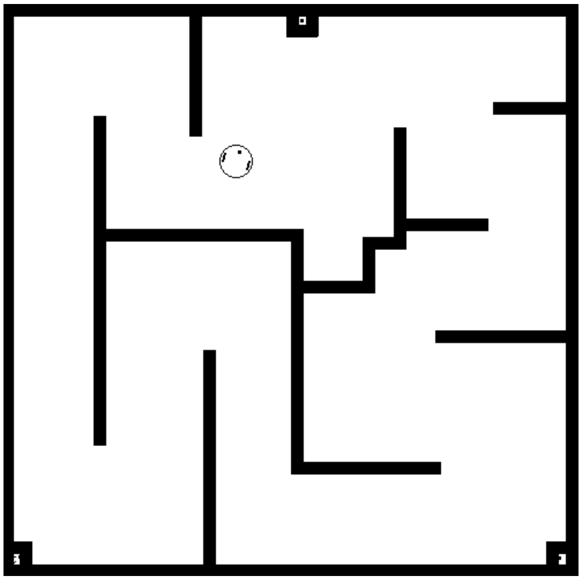
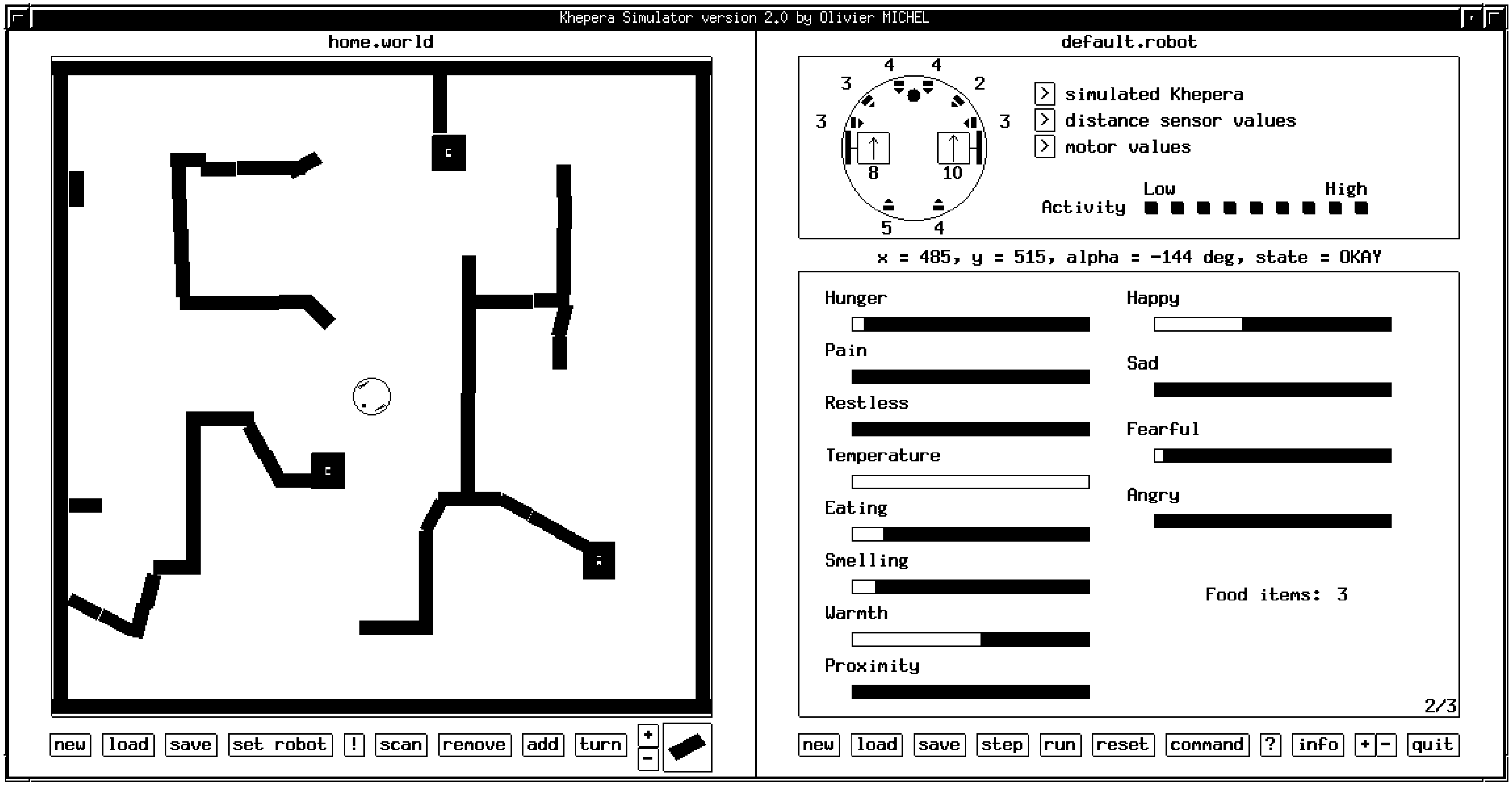


Figure 5.1: The robot in its more demanding environment.



```
x = 485, y = 515, alpha = -144 deg, state = OKAY
```

```
Food items: 3
```

Figure 5.2: The simulated robot and its normal environment.

5.2.2 Emotion system

A new instantiation of the emotions model was made. On account of the added complexity of the task, three new sensations were added to the emotion system: Smell, Warmth and Proximity.

$$\begin{aligned}\mathcal{F} \;=\; & \{\ \text{Hunger, Pain, Restlessness, Temperature, Eating, Smell, Warmth,} \\ & \text{Proximity}\ \} \qquad (5.1)\end{aligned}$$

In addition, slight changes were introduced in the calculation of the previously existing sensations. For details consult Appendix D.1. The sensations used were:

- **Hunger:** The robot's energy deficit;
- **Pain:** High if the robot is bumping into obstacles;
- **Restlessness:** Increases if the robot does not move and it is reset whenever a behaviour is selected;
- **Temperature:** Rises with high motor usage and returns to zero with low motor usage;
- **Eating:** High when the robot is acquiring energy;
- **Smell:** Active when there is energy available;
- **Warmth:** Directly dependent on the intensity of light perceived by the robot's light sensors;
- **Proximity:** Reflects the proximity of the nearest obstacle perceived by the distance sensors.

In order to have the robot's emotional state compatible with its new task, the emotions' dependencies on feelings are such that:

- The robot is HAPPY if there is nothing wrong with the present situation. It will be particularly HAPPY if it has been using its motors a lot or is in the process of getting new energy at the moment. Even just the smell of food can make it HAPPY.

- If the robot has very low energy and it is not acquiring energy, then its state will be SAD. It will be more SAD if it cannot sense any light.

- If the robot bumps into obstacles then the pain will make it FEARFUL. It will be less FEARFUL if it is hungry or restless.

- If the robot stays in the same place too long it will start to get restless. This will make it ANGRY. The ANGER will persist for as long as the robot does not move away or change its current action. A hungry robot will tend to be more ANGRY.

Table 5.1 presents the actual values for each of the emotion dependencies on feelings. Again finding the adequate dependencies values was a simple process of trial and error, requiring few adjustments. One example was the need to adjust the FEAR and HAPPINESS dependencies so that when the agent bumps into an obstacle around a light to obtain food, the HAPPINESS generated is larger than the FEAR generated by the pain. No emotion dependencies were created for the feeling of Proximity; this feeling is used only to determine state within the learning controller.

	Hunger	Pain	Restlessness	Temperature	Eating	Smelling	Warmth	Bias
Happiness	−0.2	−0.3	−0.2	0.2	0.4	0.3	0.0	0.1
Sadness	0.7	0.0	0.1	−0.2	−0.4	0.0	−0.2	−0.1
Fear	−0.2	0.7	−0.2	0.1	−0.2	−0.2	0.0	0.0
Anger	0.2	0.1	0.7	−0.2	−0.2	0.0	0.0	0.0

Table 5.1: The emotions' dependencies on feelings.

5.2.3 Basic controller

The main improvement that was introduced with the new learning controller — Figure 5.3 — was the replacement of the primitive actions by behaviours. Taking into account the current robot feelings, and the previously received evaluations, this controller tries to maximise the evaluation received by selecting between one of the three possible behaviours. These three primitive behaviours were hand-designed and consist of the following:

Avoid obstacles — Turn away from the nearest obstacle and move away from it. If the sensors cannot detect any obstacle nearby, then remain still.

Seek Light — Go in the direction of the nearest light. If no light can be seen, remain still.

Wall Following — If there is no wall in sight, move forwards at full speed. Once a wall is found, follow it. This behaviour by itself is not very reliable in that the robot can crash, *i.e.* become immobilized against a wall. The avoid-obstacles behaviour can easily help in these situations.

It was chosen to have the primitive behaviours hand-designed and learn only the harder task of behaviour coordination in the hope that emotions might be helpful in solving some of problems found at this level.

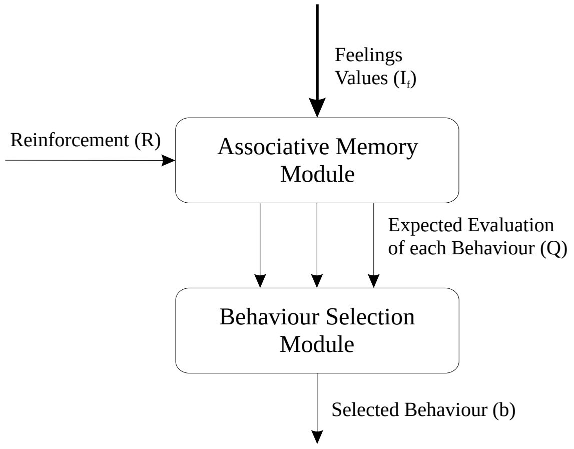

Feelings
Values (I_f)
Reinforcement (R)
Associative Memory Module
Expected Evaluation of each Behaviour (Q)
Behaviour Selection Module
Selected Behaviour (b)

Figure 5.3: Basic controller for behaviour-based control.

Apart from being behaviour-based, this controller is very similar to the previous, but has a few other differences that are highlighted next in the context of each of its modules: the Associative Memory Module and the Behaviour Selection Module.

Associative Memory Module

This plastic module uses three neural networks to associate the robot feelings with the current expected value of each of the three robot behaviours. These are three layer feed-forward neural networks, with the following characteristics:

- 9 input units, one for each feeling and a bias[2];
- 10 hidden units;
- 1 output unit that represents the expected outcome of the associated behaviour.

The neural networks initially used were not powerful enough to learn the solution for the new problem. The number of hidden units had to be increased. Tests showed that 10 hidden units allowed enough memory capacity without increasing too much the computation time for each learning iteration[3]. More important than avoiding too expensive computation times for the experiments is to avoid slow convergence of the learning algorithm which was proved previously, in Section 4.3, to have a detrimental effect on the success of the learning task. Furthermore, it was found that the linear function on the output activation function had to be replaced by the hyperbolic tangent, because the first performed very poorly. This way, both the hidden and output units have the same activation function. Replacing the output activation function by a hyperbolic tangent stipulated that the output values learned by the neural networks be bounded between -1 and 1. To circumvent that problem, the utility values given to the networks as target values were truncated to fit within that interval. This imposed some compression of the utility values, but no obvious problem was found from this in the experiments.

Behaviour Selection Module

Taking into account the value attributed to each behaviour by the previous module, this module makes a straightforward stochastic selection of the behaviour to execute

[2] The sensor readings are not necessary anymore because they are implicit in the new feelings.

[3] Tests following the procedure used to mimic the hand-crafted behaviour (described in Section 5.6.3) were made using 5, 6, 8, 10, 15 and 25 units. Best results were obtained for 10 and 15 units with very small differences for 8 or more units.

next based on the Boltzmann-Gibbs distribution. For a selection temperature[4] T, the probability of selecting behaviour b is:

$$P_n(s_n, b) = \frac{e^{\frac{Q_n(s_n,b)}{T}}}{\sum\limits_{k \in \text{behaviours}} e^{\frac{Q_n(s_n,k)}{T}}} \tag{5.2}$$

5.2.4 Experimental procedure

The previous evaluation procedure[5] had to be modified to cope with the new task. The learning period was extended to provide more time for knowledge acquisition to take place. This resulted in a huge increase in the processing time of the experiences that was redeemed in part by discarding the separate testing phases and evaluating the robot while it learned. The distinction made between a learning phase and a performance testing phase was thus eradicated. New evaluation measures were also introduced that allow a more thorough interpretation of the results.

Each experiment consisted in having thirty different robot trials of three million learning steps. In reality this duration could be made shorter, because the learning algorithm converges long before the end of these trials. The reason for the long runs was to make sure that the learning algorithm was stable and that the robot's performance would not suddenly drop, for instance. This is particularly important in the context of continuously learning agents. Nevertheless some of the preliminary experiments of Section 5.3 were made with shorter trials of only twelve hundred thousand steps.

In each trial, a new fully recharged robot with all state values reset was placed at a randomly selected starting position[6]. For evaluation purposes, the trial period was divided into sixty smaller periods of fifty thousand steps (or thirty periods of forty thousand steps, in the case of the shorter trials). For each of these periods the following statistics were taken:

[4] The selection temperature should not be confused with the temperature feeling.

[5] Details in Section 4.2.4.

[6] Any physical position, with a random orientation, in the environment that does not overlap any obstacle.

Emotion — mean of the reinforcement value provided by emotions, a measure of how positive the robot's emotional state is, which is equivalent to the emotion-dependent reinforcement measure taken in previous experiments;

Reinforcement — mean of the reinforcement obtained during all the steps which is equivalent to the previous measure if emotion-dependent reinforcement is used or to the sensation-dependent reinforcement measure taken in previous experiments if sensation-dependent reinforcement is used in the experiment;

Event reinforcement — mean of the reinforcement obtained only for the steps at which the learning controller was triggered;

Energy — mean energy level of the robot;

Distance — mean value of the Euclidean distance d, taken at one hundred steps intervals, between the opposing points of the rectangular extent that contains all the points the robot visited during the last interval,

$$d = \frac{1}{100}\sqrt{(x_{max} - x_{min})^2 + (y_{max} - y_{min})^2}$$

a measure of how much distance was covered by the robot[7];

Collisions — percentage of steps where a collision was detected;

Events — percentage of steps where the adaptive controller was triggered.

It should be noticed that while the reinforcement statistic is a good measure of overall performance, the event reinforcement reflects the actual reinforcement received by the adaptive controller.

In the graphs of the results, an average of the different statistics over the several trials is presented with error bars representing the 95% confidence interval[8].

[7] An iterative step-by-step distance measure would offer little information, because it would equally result in high values for situations where the robot is energetically moving in a very small region and situations where the robot quickly covers its entire environment. The hundred step interval was carefully chosen to capture the difference implicit in the previous situations and still be small enough to measure most of the robot's motion.

[8] The same as previously, consult Appendix B for details.

Also reported within this chapter are a few follow-up experiments that consisted of taking one of the final robots achieved by a normal experiment trial and testing it for a further hundred thousand steps. These experiments were done to examine certain behavioural details of the trained robots which are described during the presentation of results.

In Appendix D.2, a summary of the different settings used in individual experiments is presented together with the values of the various system parameters.

5.3 Experiments: Control Triggering

5.3.1 Introduction

In a robotic environment, a distinct state can be found at virtually every step. The perception of the world will always be at least slightly different from step to step due to noise. Nevertheless, making a re-evaluation of a behaviour-based system every step by performing an evaluation of the previous behaviour and selecting a new behaviour is not wise. It is both a computational waste and a hindrance to successfully learning the advantages of each of the behaviours. If the behaviour is evaluated and eventually replaced every step, then it will not have time to develop to its full potential and will be reduced to small individual actions that will look almost random. This will make it difficult for the learning system to make a correct evaluation of the possible achievements of the behaviours. On the other hand, if the behaviours are left running for too long, events may occur that will make them inappropriate for the new situation. The ideal would be to know when a significant change has occurred in the environment that makes a re-evaluation necessary.

Using emotions to trigger state transition seems reasonable, because emotions can provide a global summarised vision of the environment. Any important change in the environment is liable to be captured by changes in the emotional state.

Emotions are frequently pointed to as a source of interruption of behaviour (Sloman and Croucher, 1981; Simon, 1967) in the domain of more traditional symbolic Artificial Intelligence architectures. In general, it is considered that behaviour should be interrupted and eventually replaced whenever a strong emotion is felt. My added claim is that if the emotional intensity falls, then behaviour should also be changed, because the crisis that gave rise to the emotion has probably been solved. So state transition is triggered not only by sudden rises of emotional intensity but also by abrupt drops. Implicit in this approach is the fact that the emotion model being used is continuous and so does not provide a clear cut onset or termination of emotions, requiring that abrupt changes be detected instead.

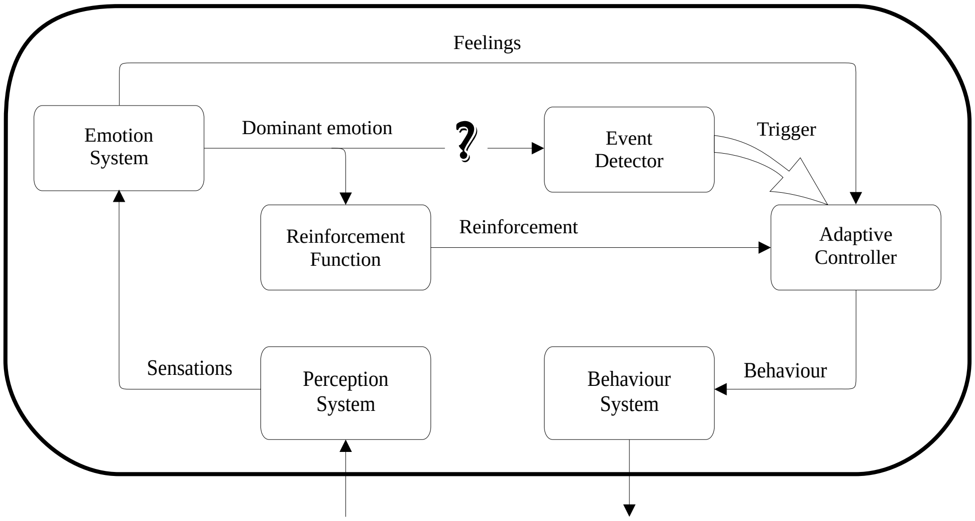


Figure 5.4: Emotions triggering state transition.

In order to test *whether emotions can successfully be used to trigger state transitions* (see Figure 5.4), two controllers were designed:

Event-triggered — Based on the ideas expounded above, a controller was designed that has state transitions triggered by the detection of significant changes in the emotional state. From the robot's point of the view, an event occurs whenever there is a significant change in emotional state, as this should reflect a relevant event in the robot-environment interaction.

Interval-triggered — A simple alternative to emotion-dependent event detection used for comparison. This controller triggers the adaptive controller at regular intervals. In particular, the inadequacy of establishing a state transition at every step is shown empirically.

The development and evaluation of these two controllers is the topic of the following two subsections. Next, a comparison between the two is made also taking into consideration the performance of both a competent and a random controller. Finally, experiments with a further increase in task difficulty are reported that make clear the advantages of emotion-triggered state transition.

5.3.2 Event-triggered controller

To test the hypothesis above, a controller with emotion-dependent event detection was designed. An event is detected whenever:

- there is a change of dominant emotion, including changes between emotional states and neutral emotional states (*i.e.* states with no dominant emotion);
- the current dominant emotion value is statistically different from the values recorded since a state transition was last made, *i.e.* if the difference between the new value and the mean of the previous values exceeds both a small tolerance threshold and ξ times the standard deviation of those previous values, where ξ is a constant (details below);
- A maximum limit of 10 000 steps is reached.

If an event occurs, then the adaptive controller is triggered: the previous behaviour is evaluated and a new behaviour is selected according to the new situation. Otherwise, the current behaviour is left running.

The calculation of the mean and the standard deviation of the emotion intensity takes into account all the steps between events. When a new event is detected, the restlessness feeling is reset and the emotional state is re-evaluated. This is the first state taken in the calculation of the two statistical variables. In the following steps, these variables are iteratively updated until an event is detected. It should be noticed that a new state can only be discriminated statistically after at least two states have been recorded.

A minimum difference for value discrimination was required, a tolerance threshold of 0.02, to disregard insignificant variations in intensity value. Otherwise, in situations

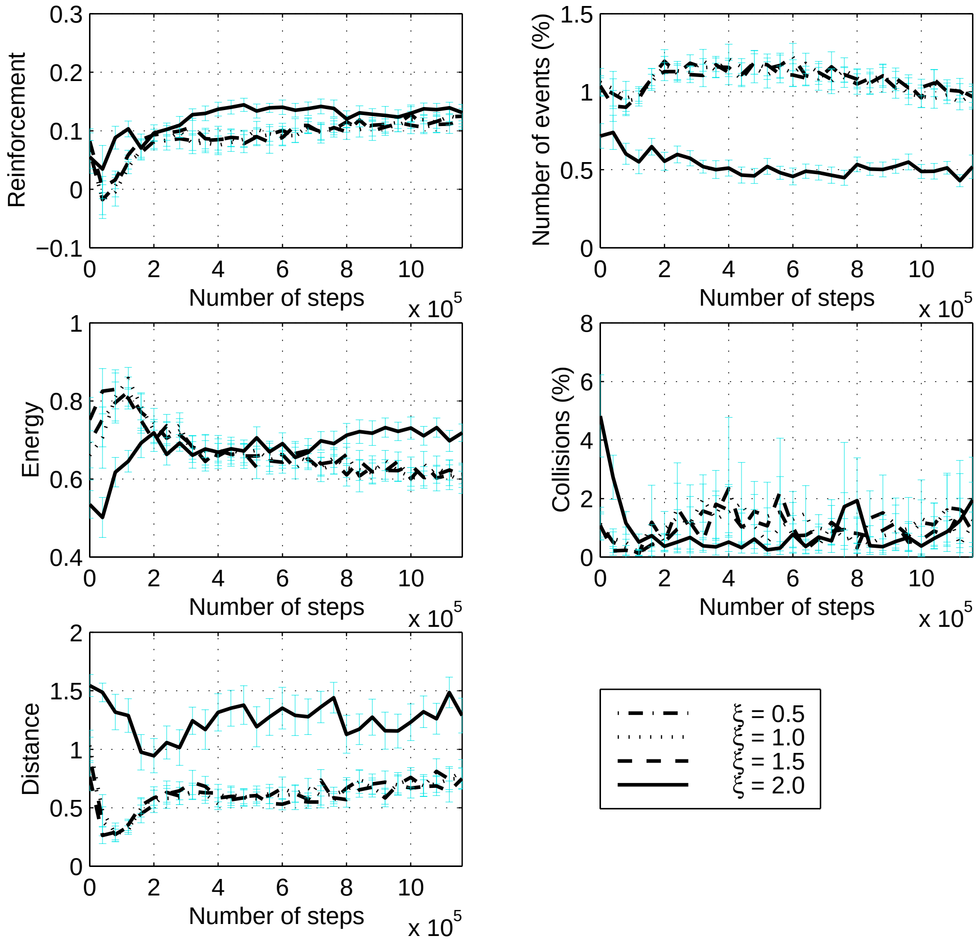


Figure 5.5: Event-triggered controller with different values of ξ and no limit on the maximum number of steps.

of very low standard deviation, imperceptible variations would be caught by the event detection mechanism.

The factor ξ is the key parameter of the event detection mechanism. Although an appropriate value was easily found, it was considered important to do a more extensive investigation of the possible values it could take, so several short experiments were done to test different values. Figures 5.5 and 5.6 show two iterations of this process.

For values of ξ below or equal to 2, the maximum limit of steps is actually not required. It was only for higher values that problems were found. The higher the value the larger the number of robots that would stop detecting events altogether. The problem is that if ξ is set too high then it becomes impossible for the event detection mechanism to

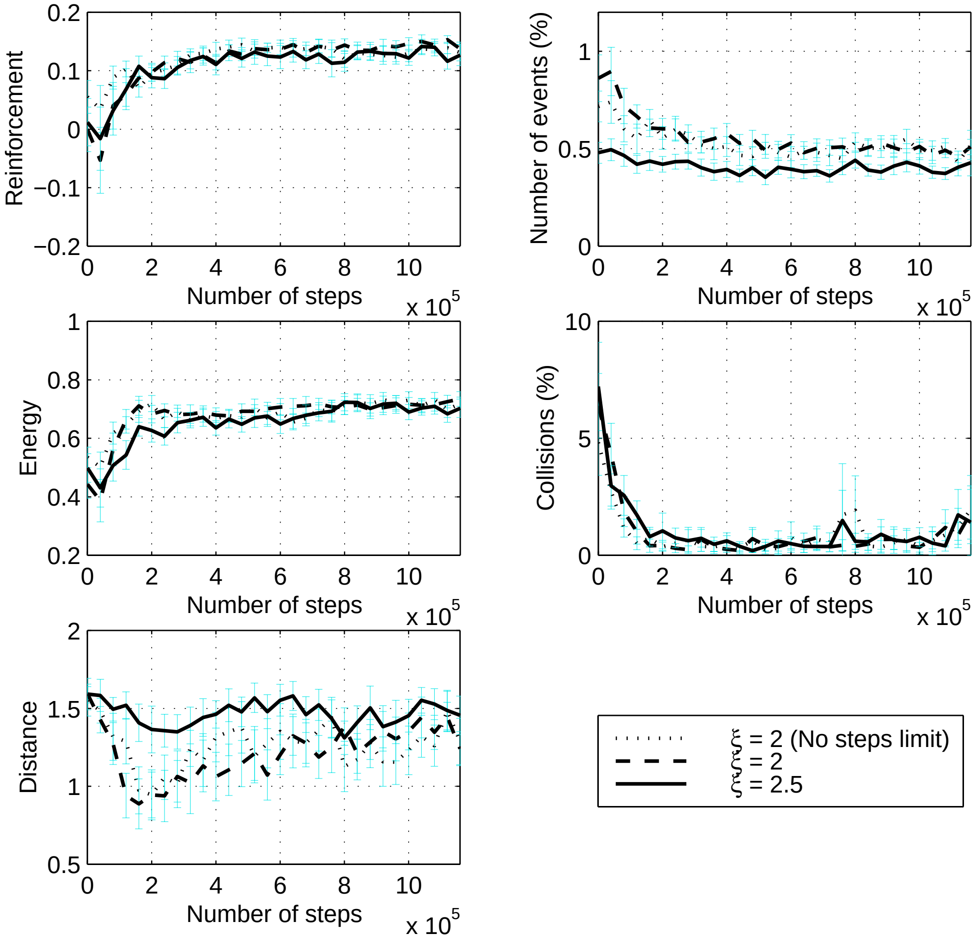


Figure 5.6: Event-triggered controller with different values of ξ.

discriminate between different intensities of the same emotion. In experimental trials where detection of events had ceased, robots were often found doing a wall-following trajectory in an advanced state of starvation. In this case, no new emotion was liable to pop up and the intensities of SADNESS felt by the robot were not different enough to trigger an event, even if the robot happened to pass by a light. The cycle could only be broken by forcing an event after a maximum step limit. This limit was chosen high enough to be the least intrusive possible, while still solving this problem. For instance, with ξ set to 2, the robot would rarely reach intervals between events larger than a thousand steps. In fact, results in Figure 5.6 show that using the maximum step limit or not with this value of ξ does not make any significant difference.

It should be clear at this point that triggering events by detection of a significant change in the intensity of the dominant emotion is essential for the system. Experiments showed that, even with a smaller maximum limit of 1000 steps, the system does not work properly if the only difference in emotional state taken into consideration is the change from one dominant emotion to another.

Figure 5.5 shows the results for different values of ξ without the use of a maximum step limit. The different performances are generally good: the robots manage to maintain a high energy value and a reduced number of collisions. The value of $\xi = 2$ was preferred over the other tested values because it has good performance with many fewer events.

The fact that 2 was the highest value suggested that still higher values should be tested, which required the introduction of the maximum step limit. Figure 5.6 shows the results obtained compared with the best obtained previously. The new results did not show either an improvement in performance or a substantial reduction in the number of events.

The conclusion reached was that either 2 or 2.5 was an adequate value for ξ and therefore the experiments in the next sections use the value 2 by default.

As Figure 5.7 shows, the discrepancy between the mean reinforcement and the event reinforcement a robot actually receives is quite significant for the event-triggered controller. The event reinforcement is worse because an event usually signals a situation where something went wrong and a new behaviour must be tried. To ensure that the mean reinforcement is not more adequate than the event reinforcement, a new test was done using as reinforcement the mean reinforcement obtained during the whole period the robot was executing the previous behaviour. Figure 5.8 demonstrates that there are no substantial differences in performance between the use of the two types of reinforcement. For simplicity event reinforcement will continue to be used.

5.3.3 Interval-triggered controller

As stated before, generating an evaluation and selection of a behaviour in every step is not fruitful. The initial experiments done with the behaviour-based controller did so and were an endless source of disappointment. In Figure 5.9, the results obtained are

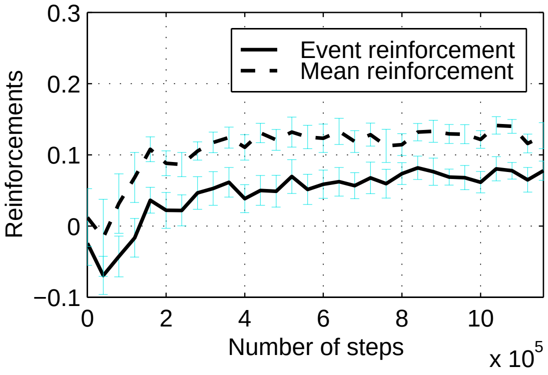


Figure 5.7: Event and mean reinforcements of the event-triggered controller (ξ set to 2.5).

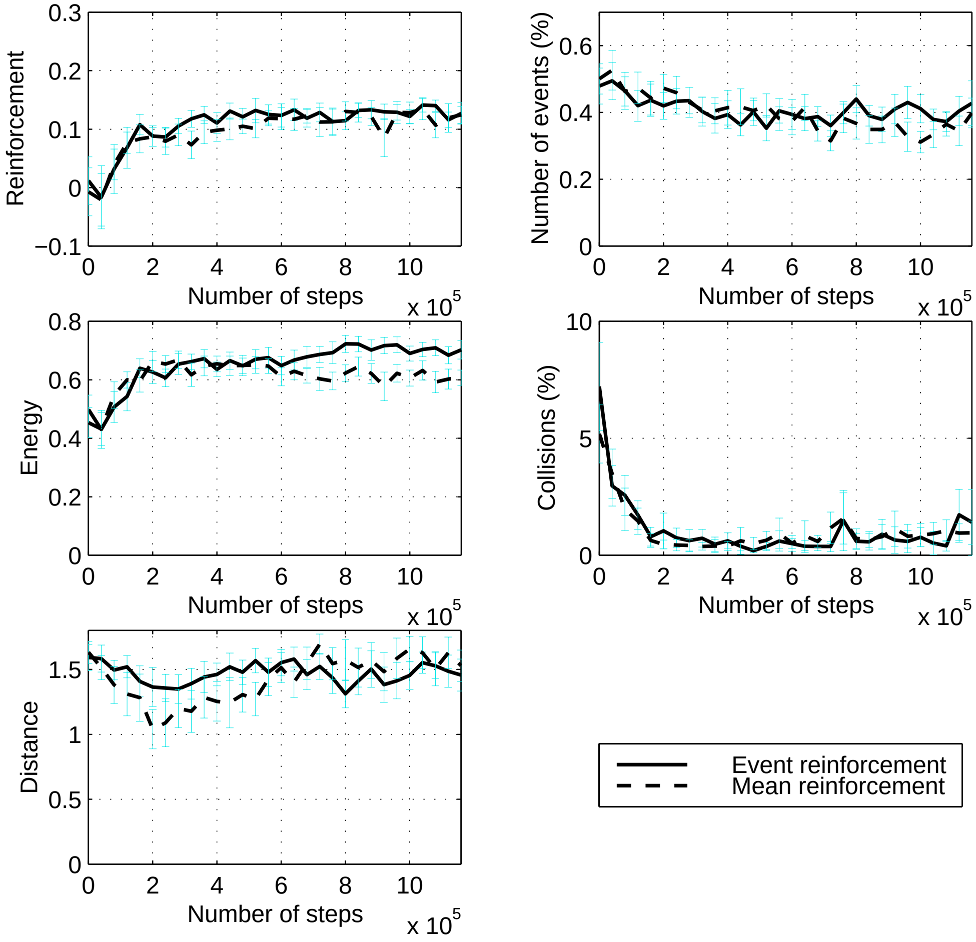


Figure 5.8: Event-triggered controller using mean reinforcement (ξ set to 2.5).

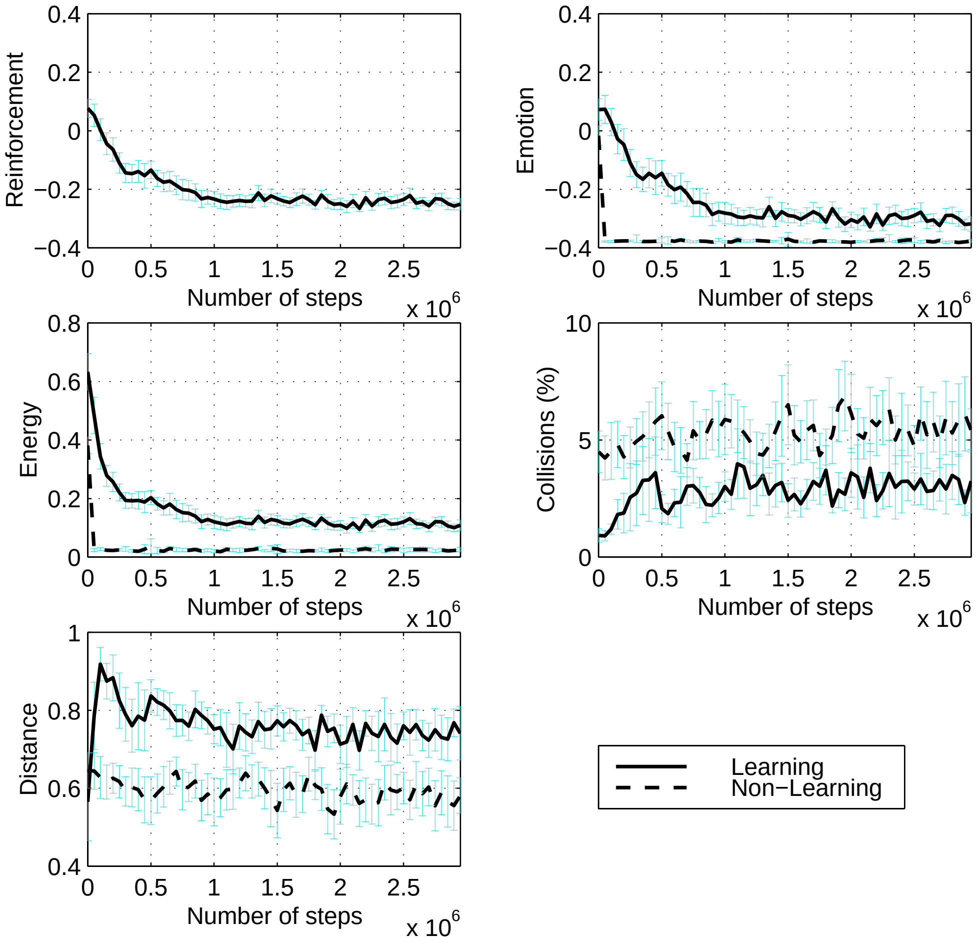


Figure 5.9: Step-triggered controller with and without learning.

shown and compared with the results for the same step-triggered controller without learning. Results show that the controller does not learn much: its performance is not very different from that generated by the random selection of behaviours exhibited by the non-learning controller.

An increase in the time interval between consecutive control iterations is imperative, but finding the right interval is not trivial and required extensive testing. On the one hand, small intervals do not allow a proper behaviour evaluation, leading to a poor overall learning performance. Under these conditions, the robot is unable to maintain its energy level. On the other hand, if the interval is too large, the number of collisions increases, because it takes longer for the robot to notice the obstacles it crashes into. If the interval is increased enough the robot will also become incapable of maintaining

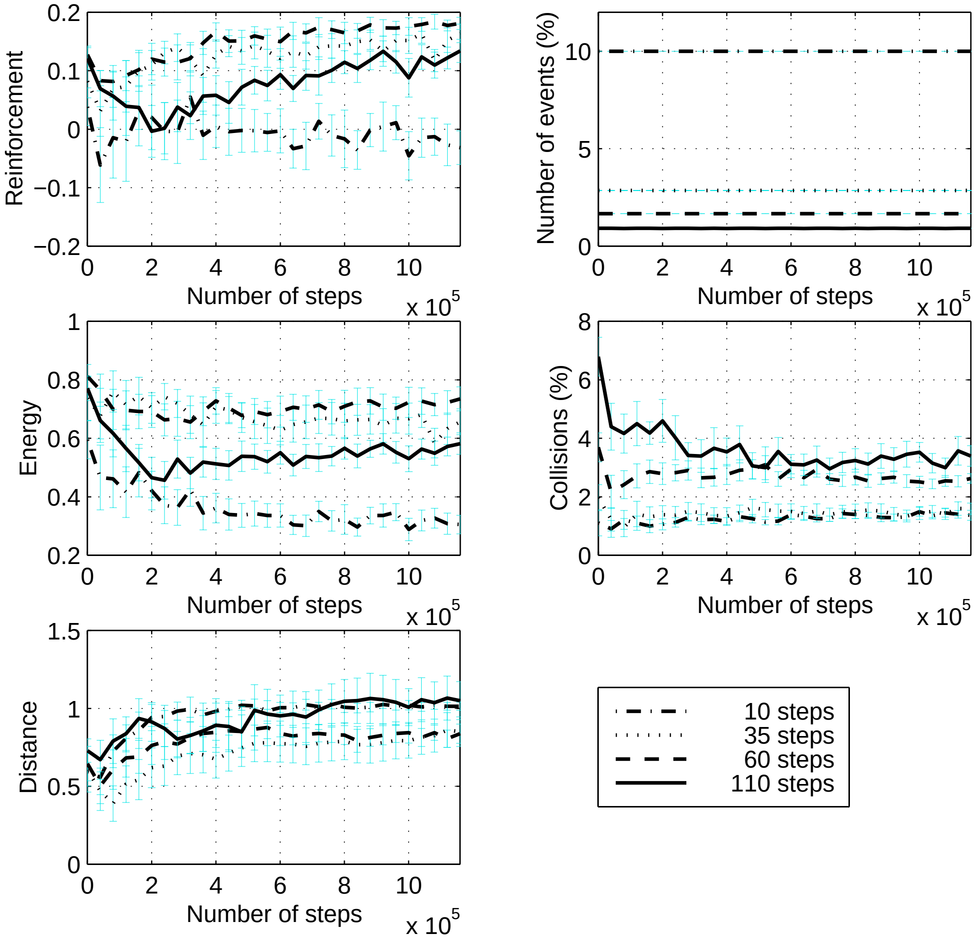


Figure 5.10: Interval-triggered controller with different durations of intervals.

its energy level, because its change of behaviours will not be fast enough to enable energy acquisition. Figures 5.10 and 5.11 portray two sets of short experiments done to find the right interval. In the first of these figures, the issues discussed above are particularly patent. In the second figure, the performance of the different intervals is not as diverse, because the space of search has been reduced.

Experiments such as these show how important it is to synchronise the duration of behaviour execution with the dynamics of the robot-environment interaction and thus allow compatible time-scales between them. The interval of 35 steps was considered the best suited, because it nicely accommodates the different issues involved, maximising the trade-off between reduced number of collisions and energy maintainance.

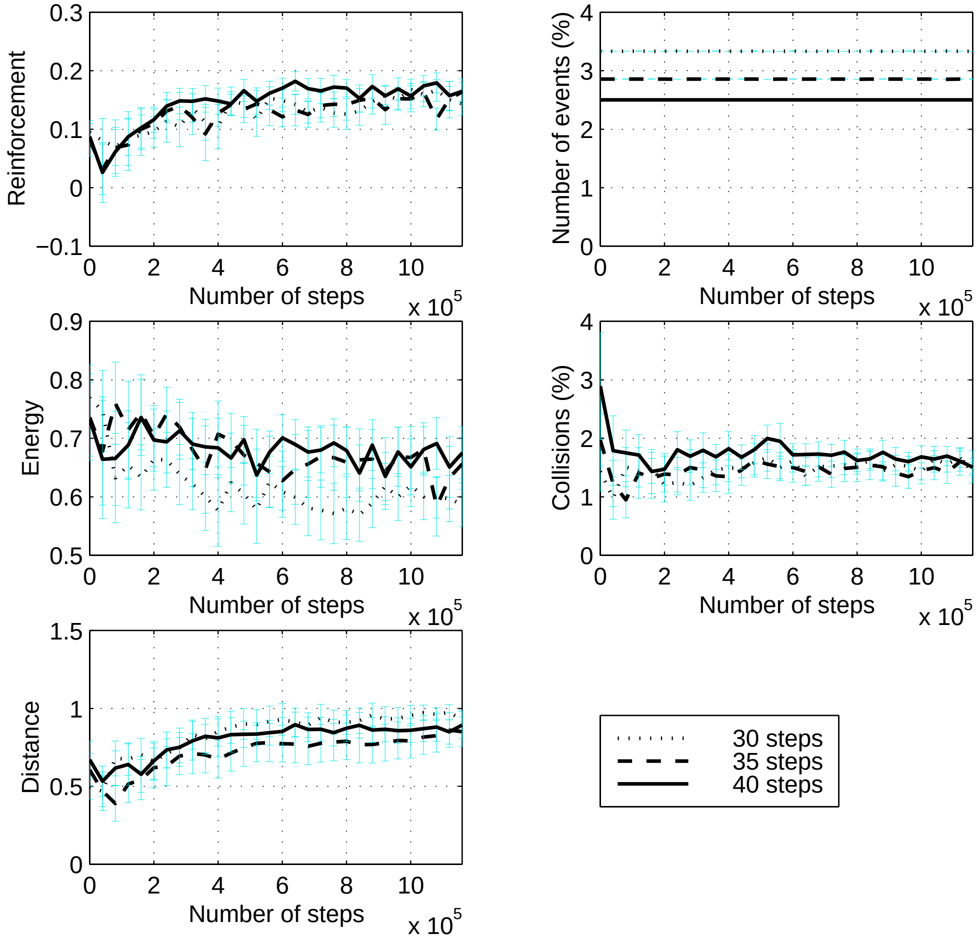


Figure 5.11: Interval-triggered controller with different durations of intervals.

5.3.4 Assessment

Establishing the standards

For a better evaluation of the controllers realised in the previous two subsections, two other controllers were produced:

Random — This controller simply selects a random behaviour at each step. It was included in the experiments to give a baseline to the result values, showing how low the performance of an unsuccessful learning controller can be. This is particularly relevant for the experiments at hand, where reinforcement tends to drop naturally with time, making it difficult to evaluate the real achievements made by the learning systems.

Hand-crafted — The purpose of designing a controller by hand was to determine how much reinforcement a reasonably successfully controller would receive. For a fair comparison with the other controllers, this controller uses the same behaviours and no extra external or memory information unavailable to the others, but has to resort to a random number generator to deal with some difficult environmental situations.

The random controller described above is the non-learning step-triggered controller examined earlier. When learning is turned off, controllers display random behaviour selection, because the initial controller's preferences are neutral, *i.e.* every behaviour starts off with the same utility value.

Designing the hand-crafted controller was not trivial. It was actually a slow and arduous cycle of test and redesign. Solving the problems of wandering in the environment and successfully eating when necessary was quite straightforward. Avoiding obstacles, on the other hand, was quite tricky and would often lead to fatal deadlock situations, the main reason being the poor sensory capabilities of the robot which allow it to lose sight of nearby obstacles very easily.

The hand-crafted controller uses the emotion-dependent event detection, with the relatively low value of 1.5 for the ξ parameter. In fact, changes in the control triggering of this controller produce significant alterations in its performance. Figure 5.12 shows examples of other settings. When the controller was tested with $\xi = 2$ or $\xi = 2.5$, its energy level dropped significantly. If, on the other hand, the hand-crafted controller is triggered at every step, the result is eventually a robot trapped in some part of the environment and incapable of maintaining its energy. An example of such a deadlock is presented in Figure 5.13. This was obtained for a robot using event triggering with ξ set to 2. These robots also suffered this kind of crash situation frequently, but would eventually recover after some thousands of steps.

It is natural that the controller works better with the settings it was designed for in the first place. Nevertheless, this pronounced dependence on the triggering mechanism shows once again how important the latter is. Setting the triggering mechanism correctly can make the difference between a successful robot or a failed robot.

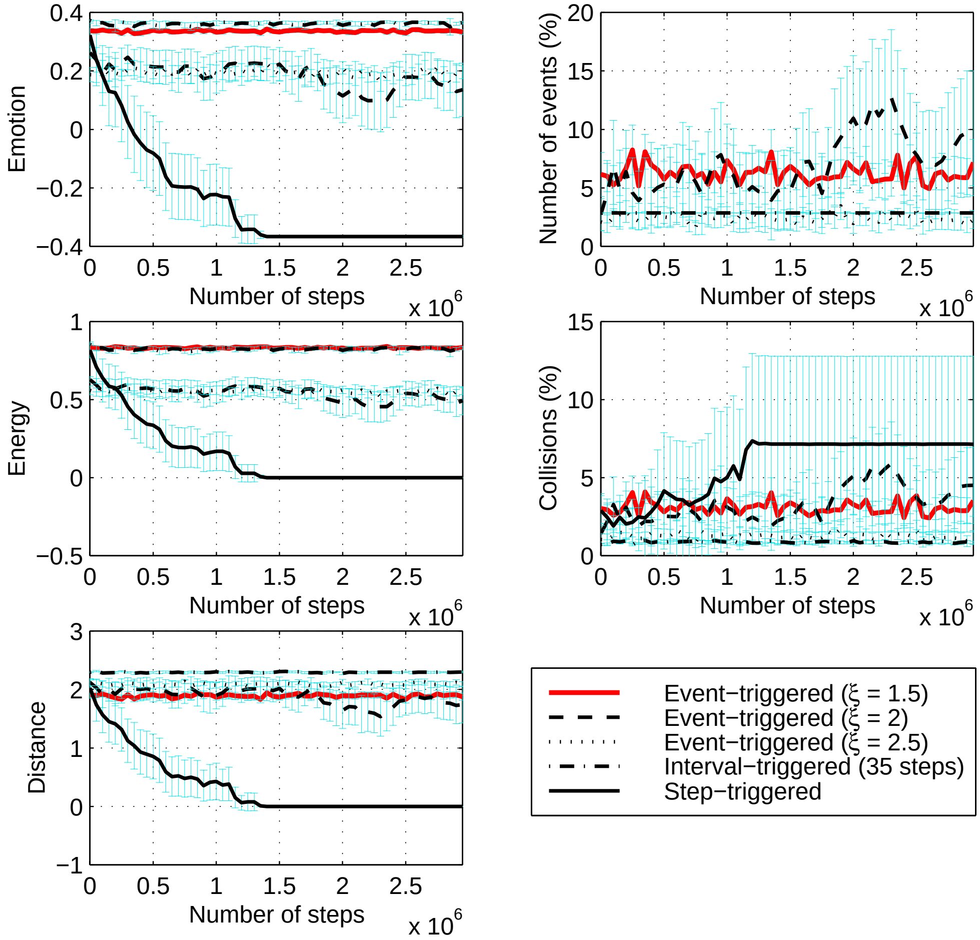


Figure 5.12: Hand-crafted controller with different triggering mechanisms.

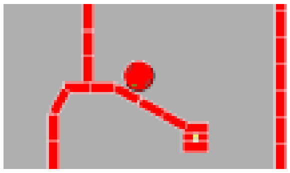


Figure 5.13: Crashed situation of a hand-crafted controller tested with $\xi = 2$.

Analysis of relative performance

Four identical experiments were done, each using one of the different controllers. In Table 5.2, a summary of the results is given. Looking at the graph curves in Figure 5.14, it can be safely assumed that, for every controller, learning has fully converged when a robot reaches the middle of its trial. The summary table presents the average of the values obtained from that point onwards.

Controller	Reinforcement	Reinforcement (Events)	Emotion	Events (%)	Energy	Collisions (%)	Distance
Event-triggered	0.13	0.07	0.19	0.4	0.71	2.8	1.5
Interval-triggered	0.18	0.17	0.22	2.9	0.65	1.5	0.9
Random	–	–	−0.38	100.0	0.02	5.6	0.6
Hand-crafted	0.24	−0.07	0.34	6.2	0.83	3.0	1.9

Table 5.2: Summary of results obtained for the controllers employing different triggering mechanisms. The values presented are the mean of all the values obtained in the last half of the trials.

Looking at the graphs, one can see that the learning controllers do manage to learn their task. Their performance is much better than that exhibited by the random controller. It is also noticeable that the successful learning controllers have significantly worse reinforcement than the hand-crafted controller. This is directly related to the higher average energy obtained by the latter. In fact, in terms of obstacle avoidance the hand-crafted controller performs worse. The lower energy of the learning controllers is actually not much of a problem, as long as they are able to keep it relatively high above zero: and this is done with success.

The hand-crafted controller having higher energy only shows that this controller acquires energy more often, which can be at least partially attributed to the higher number of events it has available. With ξ set to the relatively low value of 1.5, it has a more sensitive event detection mechanism that is triggered by smaller variations in the emotion intensity. In reality, as shown in Figure 5.12, the results obtained with ξ set to a larger value are very similar to those of the learning controllers.

There is no significant difference in performance between the two learning controllers. The difference in terms of event reinforcement does not reveal much apart from the

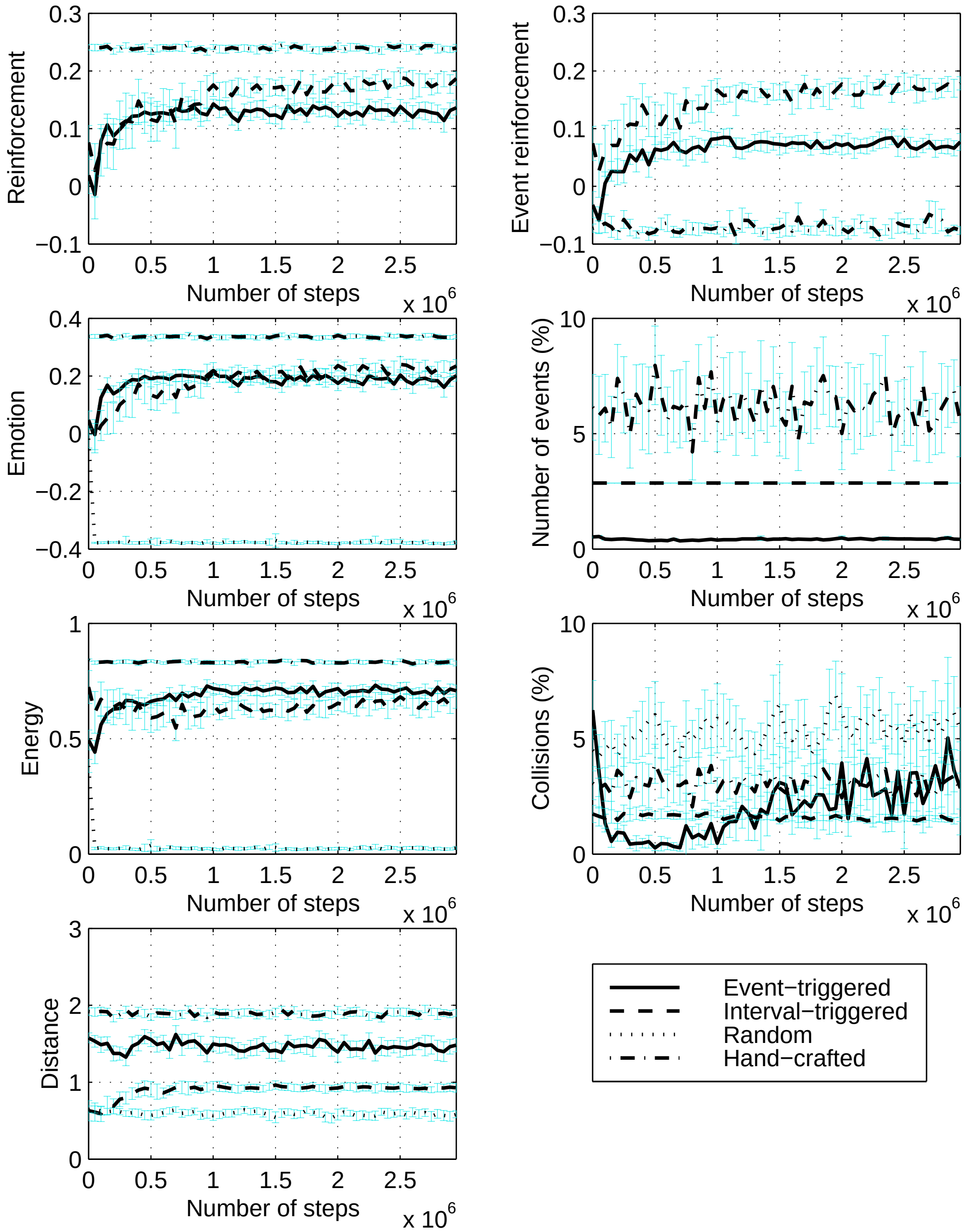


Figure 5.14: Comparison of the different triggering mechanisms.

fact that the event-driven controllers are often triggered when something goes wrong. On the one hand, the event reinforcements of the interval-triggered controller are very similar to its overall reinforcement, because the events are picked at regular intervals and independently of their value. On the other hand, the event-driven controllers are triggered in very specific situations that are often associated with negative evaluations; typically, circumstances where the current behaviour had to be changed, because it was not adequate anymore.

The event-triggered controller does not perform better than its interval-triggered counterpart, but manages to have similar learning performance with a much reduced number of events. This can also be an important issue in real time systems like robots, because it saves precious computation time.

In fact, the performance of the event-triggered controller converges in a much smaller number of learning steps than that of the interval-triggered controller. Figure 5.15 demonstrates this point by presenting the performance of the controllers in terms of the number of events, instead of the number of steps. It is the number of events that accounts for the number of learning steps because it is only during events that the robot learns, *i.e.* it updates the utility values of its behaviours. In order to obtain these results, two experiments were done: one for each controller. Each experiment consisted of thirty different robot trials of sixty intervals of five hundred events each. This actually corresponded to a significantly different number of total steps for each controller (see Table 5.3), and slightly different values for the various trials of the event-triggered controller.

Controller	Total in millions	Relative to normal
Event-triggered	6.08 ± 0.06	203%
Interval-triggered	1.05 ± 0.00	35%

Table 5.3: Mean duration of trials in steps and 95% confidence interval.

The graphs show that although the event-triggered controller has learned its task after one tenth of the trial, the interval-triggered is still improving its performance by the end of the trial. It is clear that the efficiency of the learning algorithm is increased by presenting it with only event-related situations.

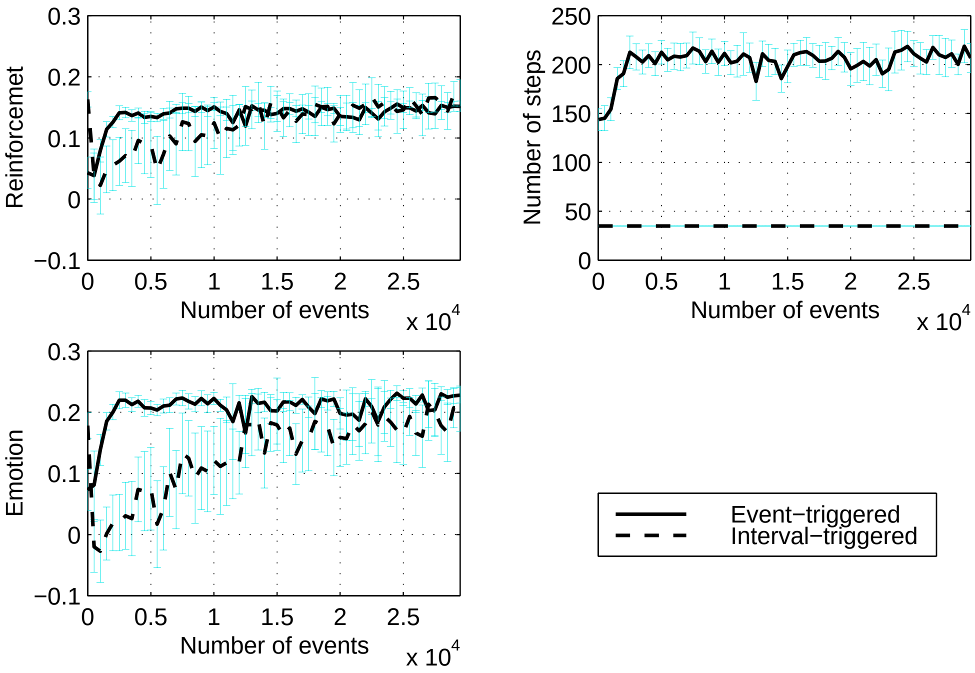


Figure 5.15: Comparing learning speeds of controllers with different triggering mechanisms in terms of events.

In the case of the event-triggered controller, it is interesting to notice how the number of events decreases as the agent learns its task. After learning how to prevent certain problems, like obstacle collisions, the robot is not as interrupted as before.

Analysis of the improvement induced by learning

The question arises of whether the triggering mechanism is not solving the task by itself. In reality, this is true to some extent. This effect can be observed in Figure 5.16 where the different triggering mechanisms are compared again, but this time with learning turned off. The task performance of the robot varies significantly with the different triggering mechanisms, even when the robot's behaviours are selected at random.

While the step-triggered continues to exhibit the worst performance, it's surprising to notice that it is actually the interval-triggered controller that has the starting advantage, because its timing is in synchrony with the task. It has an edge in terms of obstacle avoidance and energy maintainance. The event-triggered controller travels more, because it is interrupted less frequently, which is its hidden advantage.

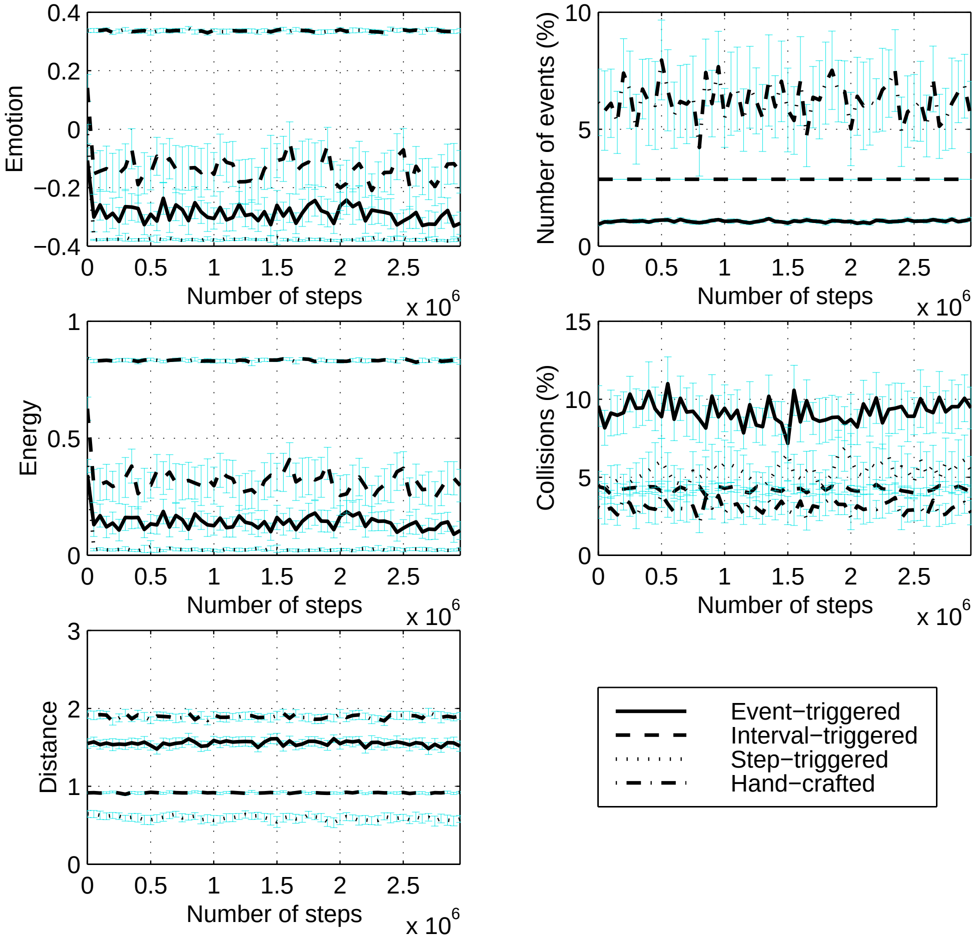


Figure 5.16: Comparing the different triggering mechanisms with no learning.

It is also interesting to consider the difference in performance between the learning controllers and the equivalent non-learning controllers. Figure 5.17 shows the relative performance of the learning controllers, taking the respective non-learning experiments as a base — *i.e.* the distribution of the non-learning performance was subtracted from the learning performance (details in Appendix B.1.2). If the learning controllers are compared in these terms then it is clear that through learning the event-triggered controller improves its performance much more than the others while the step-triggered controller is a very poor learner.

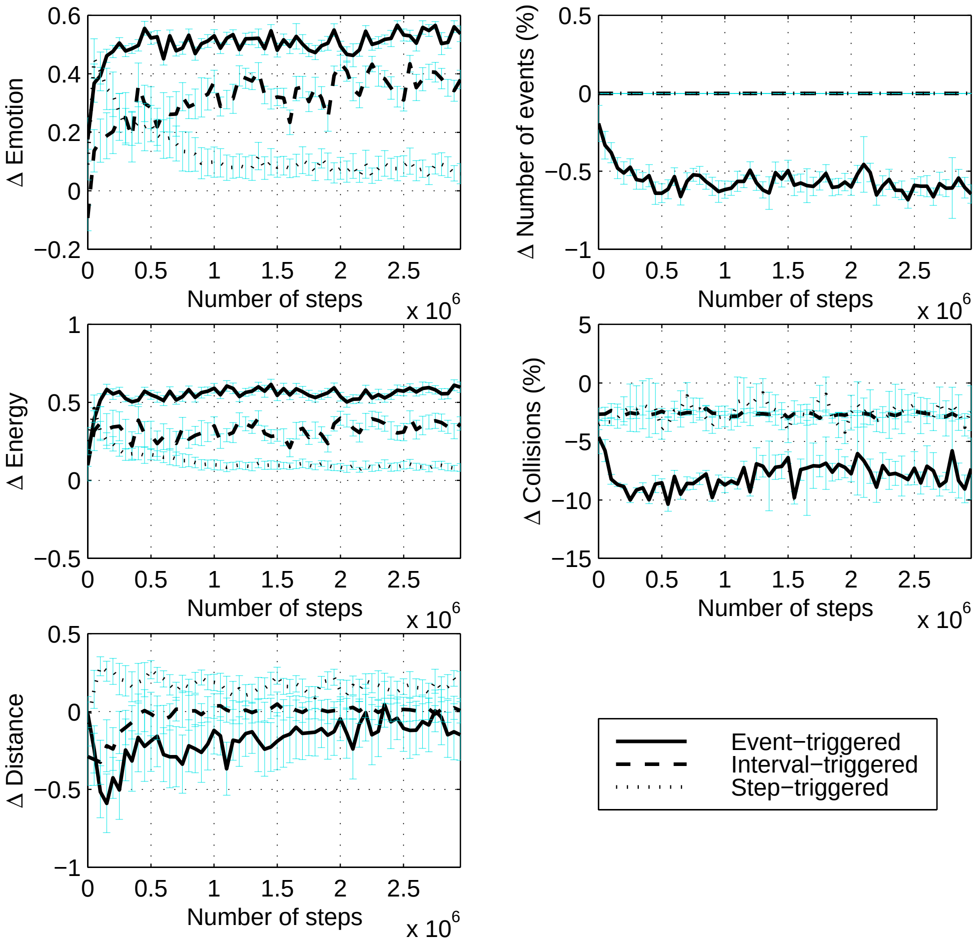


Figure 5.17: Comparing improvement of performance provided by learning with different triggering mechanisms.

Analysis of experiments' design

A closer observation of the robot's final behaviour brought forward two problems with the experiments' design:

- The restlessness feeling is intended as an indicator of the progression of the behaviour at hand. Through the emotion of ANGER it punishes the robot when the behaviour it has selected is incapable of moving the robot. Restlessness will also provide the necessary interruption in the case of emotion-dependent event detection. The problem is that it is necessary to avoid its saturation. If this happens, no more interruptions will be detected, because the dominant emotion of ANGER will not change. For this reason, the restlessness value must be reset whenever an event is detected. This is not a very far fetched solution, because it is natural for the frustration to go away when a new behaviour is selected, at least until the selected behaviour proves to be inefficient as well. However, the fact that the newly selected behaviour might be the same behaviour that was showing problems previously makes the solution a bit strange. Nevertheless, this was necessary for the controller to work effectively.

- The interval-triggered controller managed to exploit being still to save energy, and thus exhibit local behaviour around a single light. This was not the intended behaviour at all, and the only reason why the controller can get away with it follows directly from the first problem. With the frequent events provided by the control triggering of this controller, the ANGER emotion cannot reach intensities high enough to dissuade this kind of solution. Moreover, controllers that frequently select behaviours benefit from an unfair advantage in terms of reinforcement, because the ANGER emotion is not able to manifest itself.

5.3.5 Increased task difficulty

In order to prevent controllers from exploiting the low usage of the motors to save energy, two measures were taken[9]:

[9] Details of the parameter changes are in Appendix D.2.

- The normal environment was replaced by the more demanding environment pictured in Figure 5.1. This is a more corridor-like environment, where it is more difficult to travel from one light to another by chance.

- The first measure proved insufficient by itself, because the robots can apparently still manage to maintain high levels of energy if only one light is available. So the robot energetic needs were increased. Furthermore, the advantage of not moving was removed by making the value of energy decrease independent of motor usage.

Figure 5.18 shows the results obtained with the changed environment. The interval-triggered controller behaves worse than the event-triggered, but it is still quite competent. It still achieves a good level of energy by not moving a lot. This will influence its reinforcement, because it will not be rewarded by moving around the environment. However, it will not lead to punishment due to restlessness, because this feeling is reset frequently.

In order to further refine the distinction between the two, a new set of experiments was performed applying both measures discussed above: change in environment and increase in energy usage. The results shown in Figure 5.19 demonstrate the differences between the two controllers in this context. In this case, the advantages of the event-triggered controller are more evident. In particular, Figure 5.20 shows how the interval-triggered controller frequently allows its energy to reach dangerous levels while the event-triggered controller's energy is kept in a sensible range.

5.3.6 Conclusions

It was established during the experiments that triggering the controller at every step was totally inadequate. Nevertheless, the interval-triggered controller that regularly triggers the controller at longer intervals of time was found adequate. This controller has even a starting advantage over the event-triggered controller because it performs better with random behaviour selection. However, it is also less flexible. The fact that intervals are fixed *a priori* to fit the task makes it more task dependent. Furthermore, finding the right interval for the task can be time-consuming.

The event-triggered controller which triggers control at variable intervals dependent

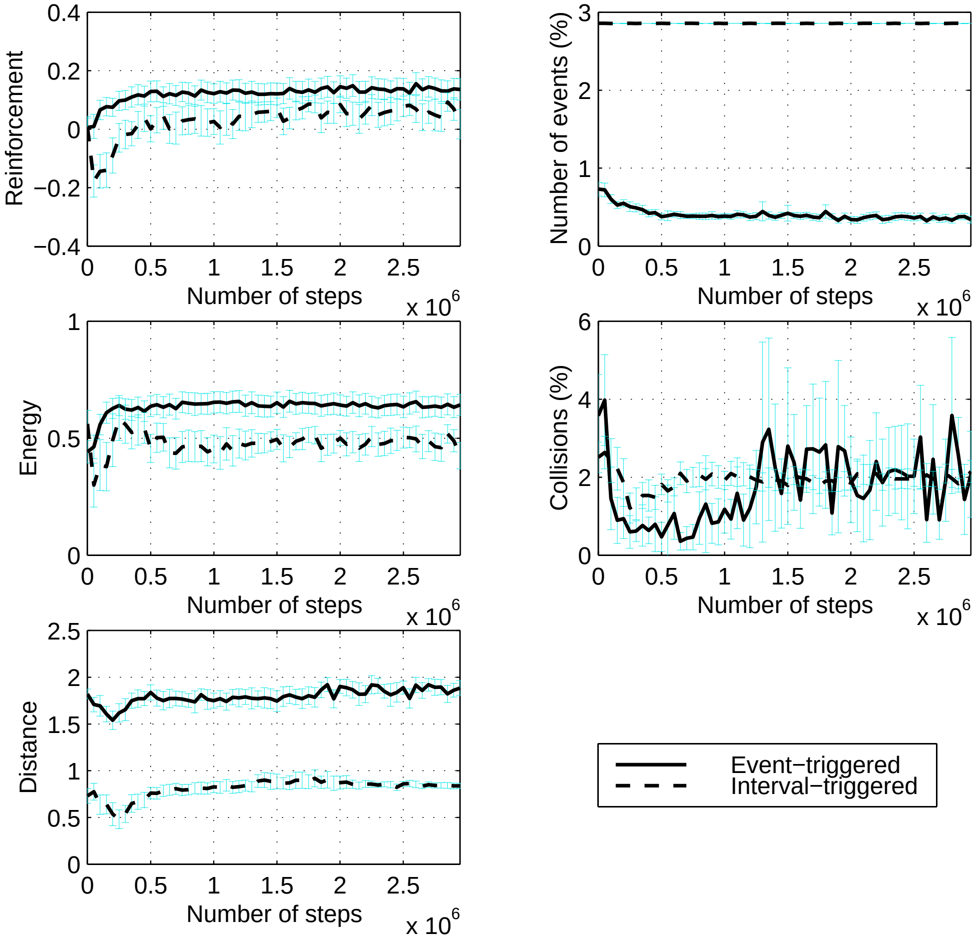


Figure 5.18: Comparing the different triggering mechanisms in the more demanding environment.

on the detection of significant changes in emotional state was the best learner. This controller has the advantage of both being a more time-efficient learner and being able to master more difficult tasks. Moreover, it manages to achieve a reinforcement similar to that of the interval-triggered controller which takes advantage of not being punished for restlessness. The reset of restlessness that permitted this unfair advantage was necessary for the event-triggered controller to work. However, other approaches to emotion-dependent control triggering could avoid this problem by looking into emotion intensity instead of variation. An example would be to have the frequency of control triggerings directly proportional to the intensity of the current emotional state.

An alternative to the use of emotion-dependent detection of events would be to look at all the controller's feelings inputs for statistical novelty instead of looking at the

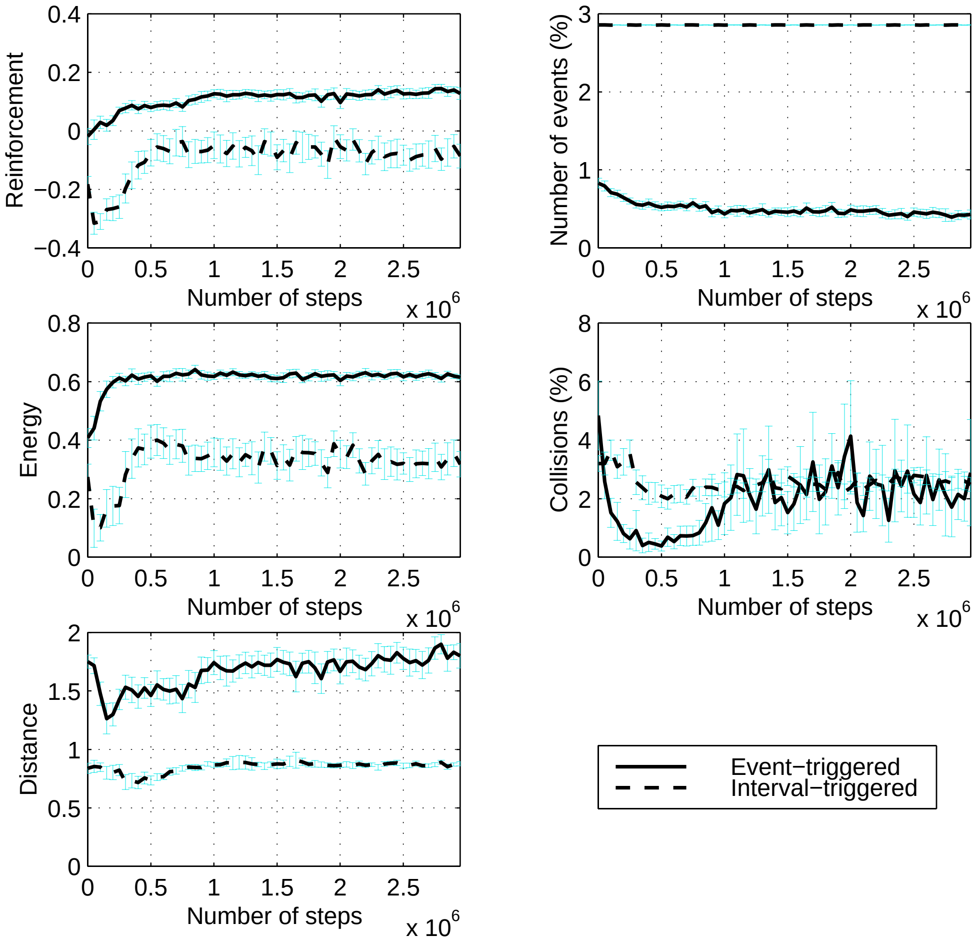


Figure 5.19: Comparing the different triggering mechanisms in the more demanding environment and with harder energy requirements.

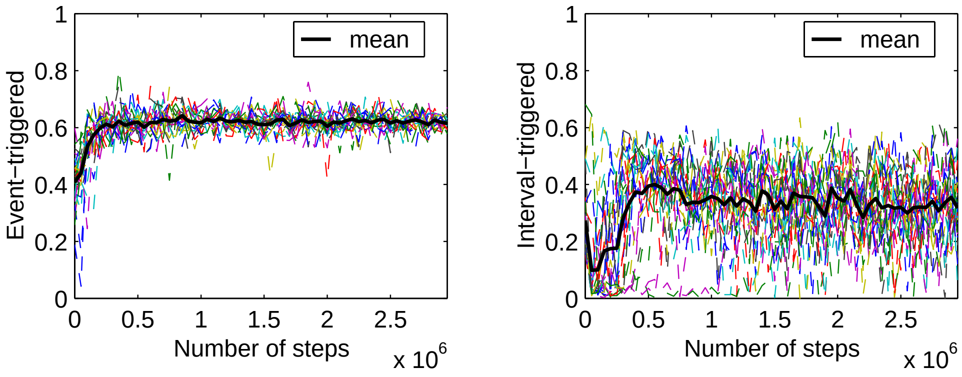


Figure 5.20: Energy values for the individual trials of different triggering mechanisms.

emotion value alone. The problem is that this solution is much less clean. Instead of only one set of statistics, this solution requires several, each one of them with a very particular behaviour. This will make a uniform test of all them difficult or even impossible, eventually requiring a separate analysis for each one of the inputs. Another advantage of using the emotional state is that emotions already take into consideration what is and what is not important in each situation, and the relative importance of each individual feature. The fact that they hide away details can even be beneficial.

5.4 Experiments: Reinforcement

5.4.1 Emotion-dependent *vs.* sensation-dependent reinforcement

After the control triggering mechanism had been established it was decided to re-test the role of emotions as a source of reinforcement (see Figure 5.21). Exactly the same emotion-dependent reinforcement function was used (see Equation 4.6). At any moment in time, the reinforcement absolute value is the intensity of the current dominant emotion or zero if there is no dominant emotion. The signal of the reinforcement value is positive when the dominant emotion is positive and negative when the dominant emotion is negative.

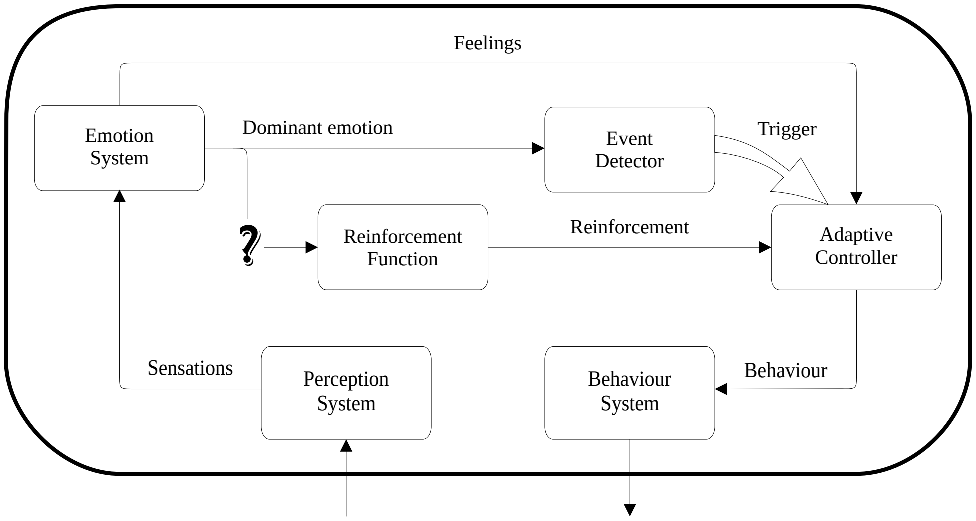


Figure 5.21: Emotions determining reinforcement value.

Again an experiment was done to test *whether emotion-dependent reinforcement is*

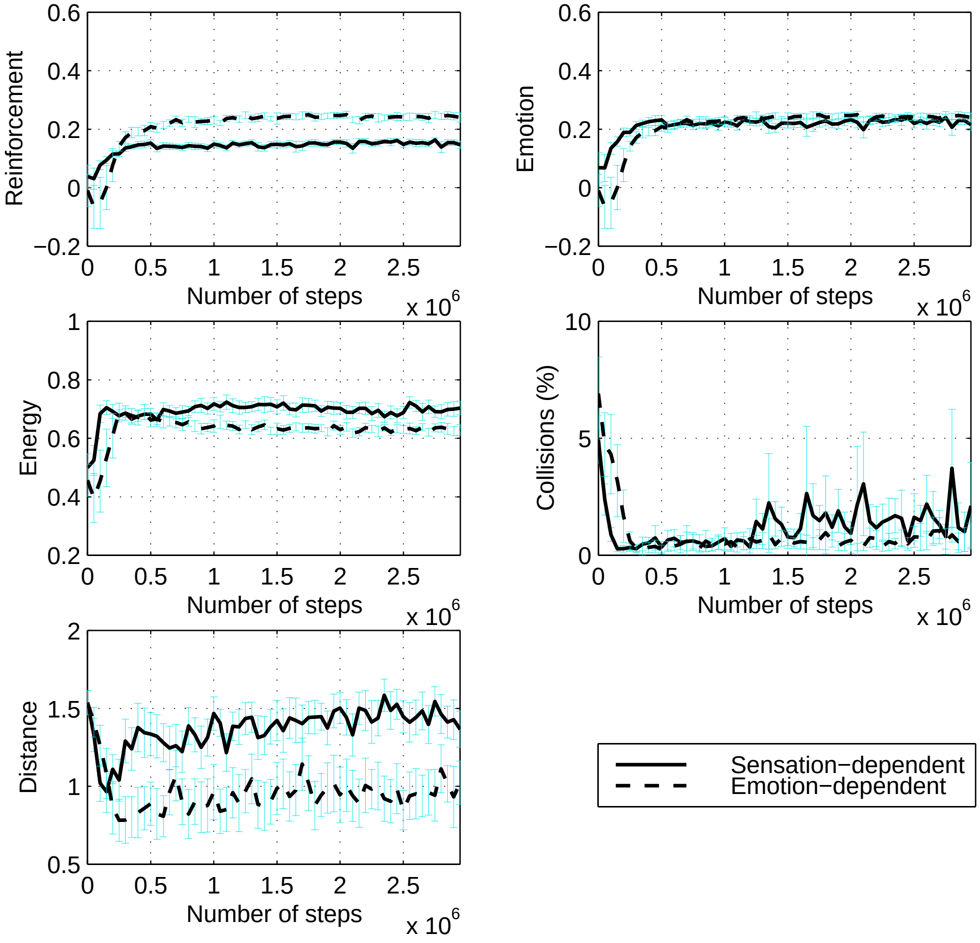


Figure 5.22: Comparing emotion-dependent reinforcement with sensation-dependent reinforcement.

competitive when compared with a more traditional reinforcement function. With the new setup for the behaviour-based controller the discrepancies between sensation-dependent and emotion-dependent reinforcement found in Chapter 4 have faded away. As shown by Figure 5.22, the emotion-dependent reinforcement is now successful and its performance is similar to that of sensation-dependent reinforcement. This can be observed in the emotion graph which is a good indicator of overall performance. The difference registered in terms of reinforcement value should not be considered in the comparative evaluation because different reinforcement functions were used for each experiment. The purpose of this particular graph is only to show the learning curve of each controller.

Taking into consideration the good results obtained, the emotion-dependent reinforcement was taken as the default for the rest of the experiments.

5.4.2 Comparison with an undifferentiated reinforcement function

The emotion-dependent reinforcement has the characteristic of only depending on one emotion at a time, if any. The reinforcement information that might be provided by emotions other than the dominant emotion is ignored. For example, if the robot is SAD and bumps into an obstacle then FEAR will overcome SADNESS and only FEAR will be taken into consideration for reinforcement. This means that reinforcement information will mostly ignore the hunger feeling and will be dominated by the pain feeling. To test whether this is an advantage for the learning controller or not, the different motivations of the agent were joined together in an undifferentiated reinforcement function ($R_n = R_u$).

The undifferentiated function (R_u, defined in Equation 5.3) was obtained from the emotions' dependencies on the feelings. It consisted in subtracting from the HAPPINESS' dependence the other emotions' dependencies and dividing the result by four to obtain a weight for each feeling. These weights were then used for the weighted sum of the sensations' values of which consists the undifferentiated reinforcement function. This would be equivalent to adding together the reinforcement values provided by each emotion and dividing by four, if the hormone system were eliminated as it was for the sensation-dependent reinforcement (see Figure 4.7). In a normal robot-environment interaction, this function has less than 0.1% difference in sign from the emotion-dependent reinforcement. This means that it rarely punishes the robot in the situations where the emotion-dependent reinforcement rewards the robot and vice-versa.

$$R_{u_n} = \frac{1}{4} \sum_{e \in \mathcal{E}} \left((B_e + \sum_{f \in \mathcal{F}} (C_{ef} S_{f_n})) \mathrm{sign}(e) \right) \tag{5.3}$$

This undifferentiated reinforcement function is not tuned but its poor performance, shown in Figure 5.23, supports the view that the non-linearities in reinforcement are important for the system. Nevertheless, it was possible to hand-craft another undifferentiated reinforcement function (R'_u, defined in Equation 5.4), by adjusting the several

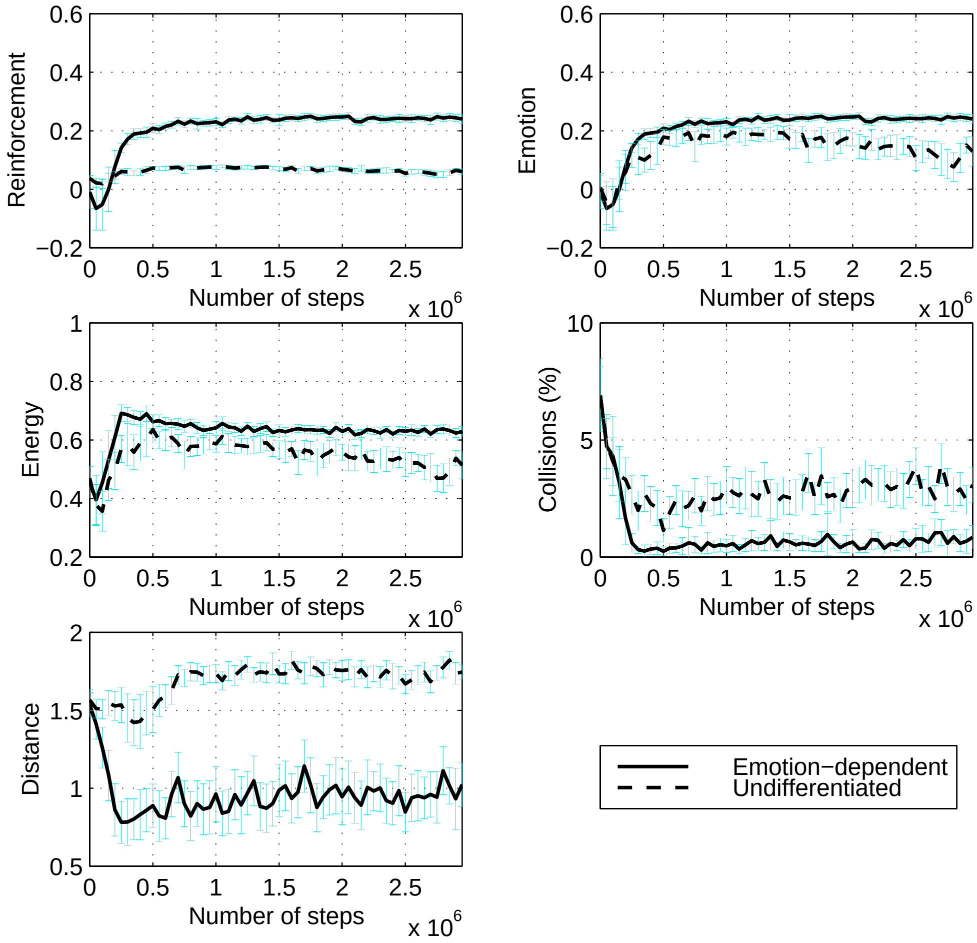


Figure 5.23: Comparing the emotion-dependent reinforcement with an undifferentiated sensation based reinforcement.

weights by trial and error, that managed to perform as well as the emotion-dependent reinforcement function.

f	Hunger	Pain	Restlessness	Temperature	Eating	Smelling	Warmth	Proximity
W_f	-0.3	-0.3	-0.3	0.2	0.5	0.2	0.1	-0.1

$$R'_{u_n} = \sum_{f \in \mathcal{F}} (W_f S_{f_n}) \tag{5.4}$$

5.4.3 Re-assessment of control triggering mechanisms.

Figure 5.24 and Table 5.4 demonstrate that the results for the different triggering mechanisms when using emotion-dependent reinforcement are consistent with the pre-

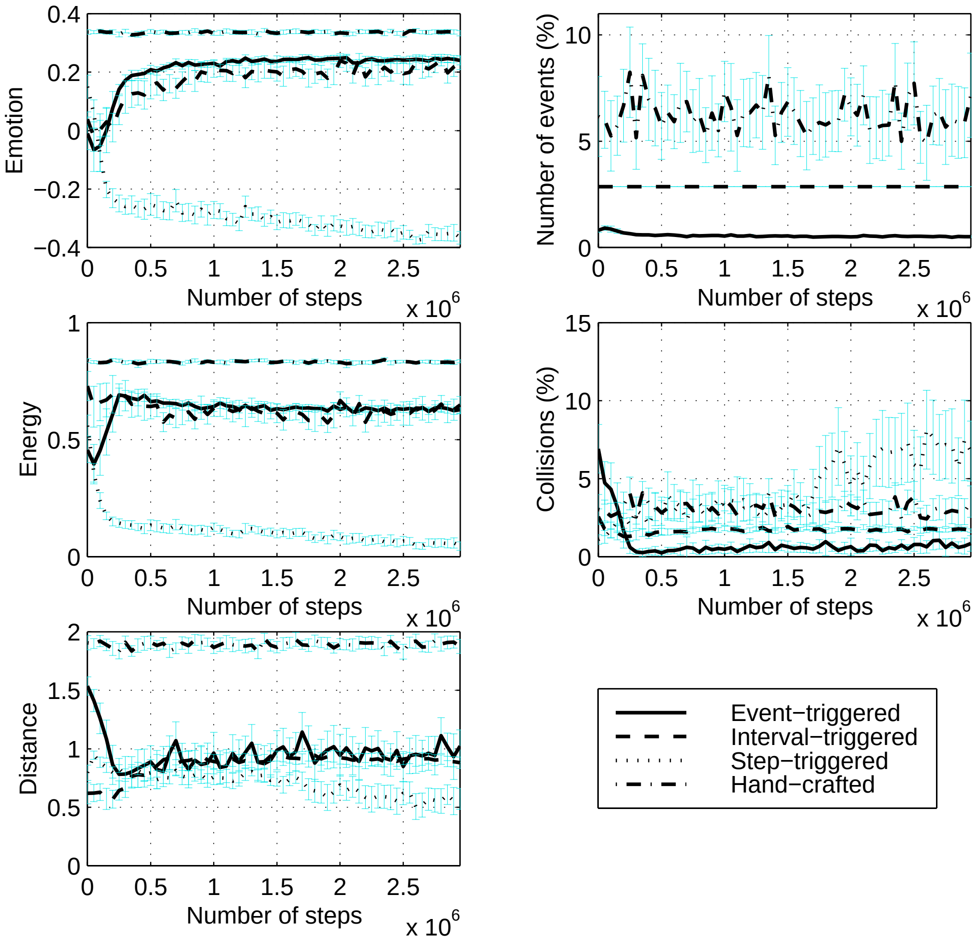


Figure 5.24: Comparing the different triggering mechanisms with emotion-dependent reinforcement.

Controller	Emotion	Events (%)	Energy	Collisions (%)	Distance
Event-triggered	0.24	0.5	0.63	0.6	1.0
Interval-triggered	0.21	2.9	0.62	1.7	0.9
Step-triggered	−0.34	100.0	0.07	5.9	0.6
Hand-crafted	0.34	6.1	0.83	3.0	1.9

Table 5.4: Summary of results obtained for different triggering mechanisms with emotion-dependent reinforcement. The values presented are the mean of all the values obtained in the last half of the trials.

viously obtained results with sensation-dependent reinforcement. The event-triggered controller still does not perform outstandingly better than the interval-triggered counterpart, but is much more efficient in terms of events. There is now also a slight difference in the number of collisions. It is natural that the event-triggered controller does better in terms of obstacle avoidance, because this controller is triggered to deal with the obstacles that the robot finds in its way instead of having to wait until the next triggering point to deal with them. However, previous results, reported in Figure 5.14, did not show a difference, possibly due to the sensation-dependent reinforcement function and the emotion-dependent triggering mechanism being out of synchronisation.

5.5 Experiments: Perception

In this section, the influence of emotions on perception was briefly examined. More specifically, a set of experiments was run to test *whether perception being influenced by the hormone system or not has an impact on the performance of the robot* (see Figure 5.25). Experiments failed to show significant differences in performance, which means that the robot can cope with a biased view of reality but does not demonstrate that emotions can be useful in this domain.

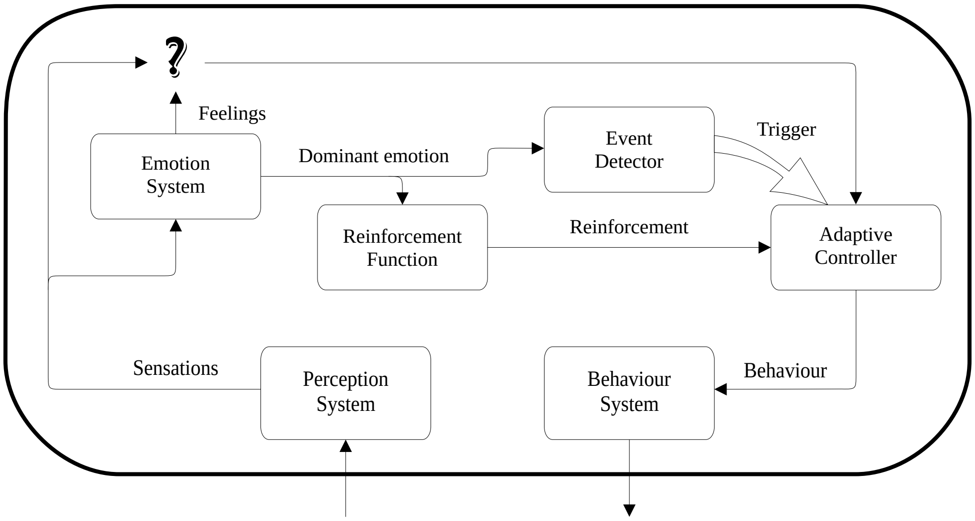


Figure 5.25: Emotions' influence on perception.

Figure 5.26 compares the normal experimental results with those obtained by replacing

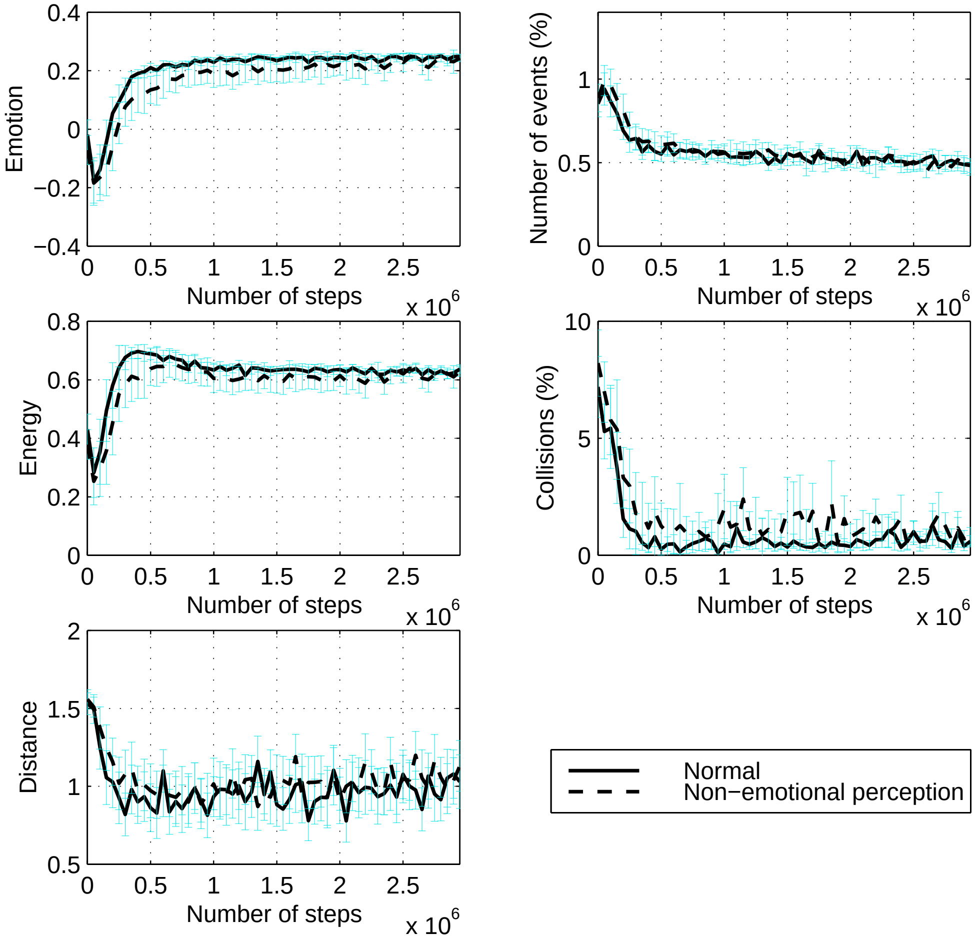


Figure 5.26: Emotional and non-emotional perception.

the neural-network inputs by sensations instead of feelings (see Figure 5.3). Although the graphs suggest that there might be an improvement provided by the emotions' influence on perception, such a deduction is not sufficiently supported by the results for a definitive conclusion to be drawn. The fact that no significant differences were found might be purely task dependent. However, the selected learning controller is surely responsible for the results to some extent. The use of neural networks to process the inputs allows for more abstraction of the input values and to compensate for any changes in magnitude caused by emotions.

5.6 Experiments: Assessment of the Emotional Controller

5.6.1 Introduction

The final emotional controller achieved is presented in Figure 5.27. It has emotional reinforcement, perception and control triggering. This controller as a whole is the topic of the present section.

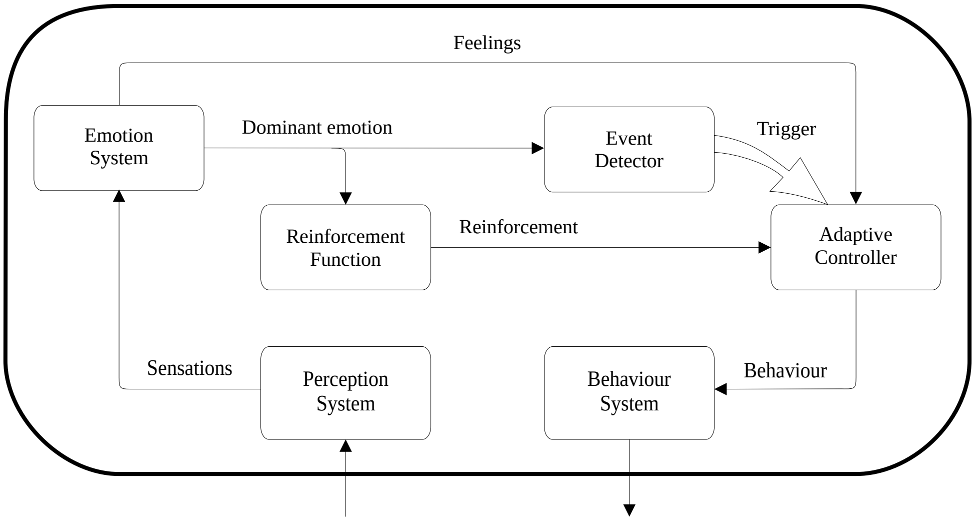


Figure 5.27: Emotional controller.

In the first subsection, this controller is compared with its non-emotional counterpart. Next, a more extensive comparison is made of the emotional controller with the hand-crafted controller. In particular, an attempt is made to mimic the hand-crafted controller through the learning controller. The final subsections examine the influence of emotions' persistence and the behaviour selection method on the performance of the emotional controller.

5.6.2 Relative to a non-emotional controller

Experiments were done to assess the competence of the final emotional system as a learning controller when compared to a non-emotional system. Results are shown in Figures 5.28 and 5.29. Figure 5.28 reports on a set of experiments done under normal conditions and Figure 5.29 on experiments done under the more demanding

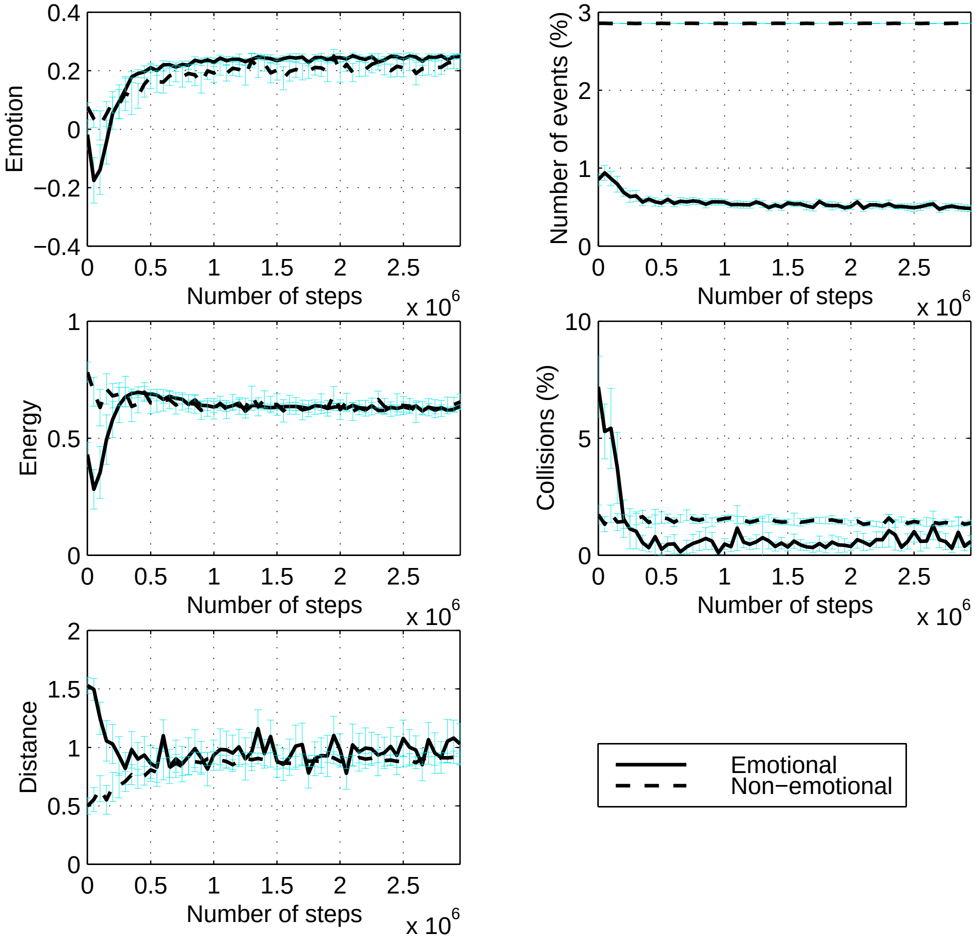


Figure 5.28: Comparing an emotional with an non-emotional controller.

environment and with harder energy requirements[10]. The differences between the two are particularly clear in more severe conditions, in which case the emotional controller's performance is much better.

Some other experiments were presented previously that had only one of the mechanisms replaced at a time by its non-emotional counterpart. Table 5.5 describes the differences between the different controllers in detail. A summary of the results obtained in normal experiments for each controller is presented in Table 5.6.

The most significant difference between the emotional and non-emotional controllers is in the number of events which is obviously due to the control triggering mechanism used. This mechanism is what is really responsible for the advantage of the emotional

[10] Details in Section 5.3.5.

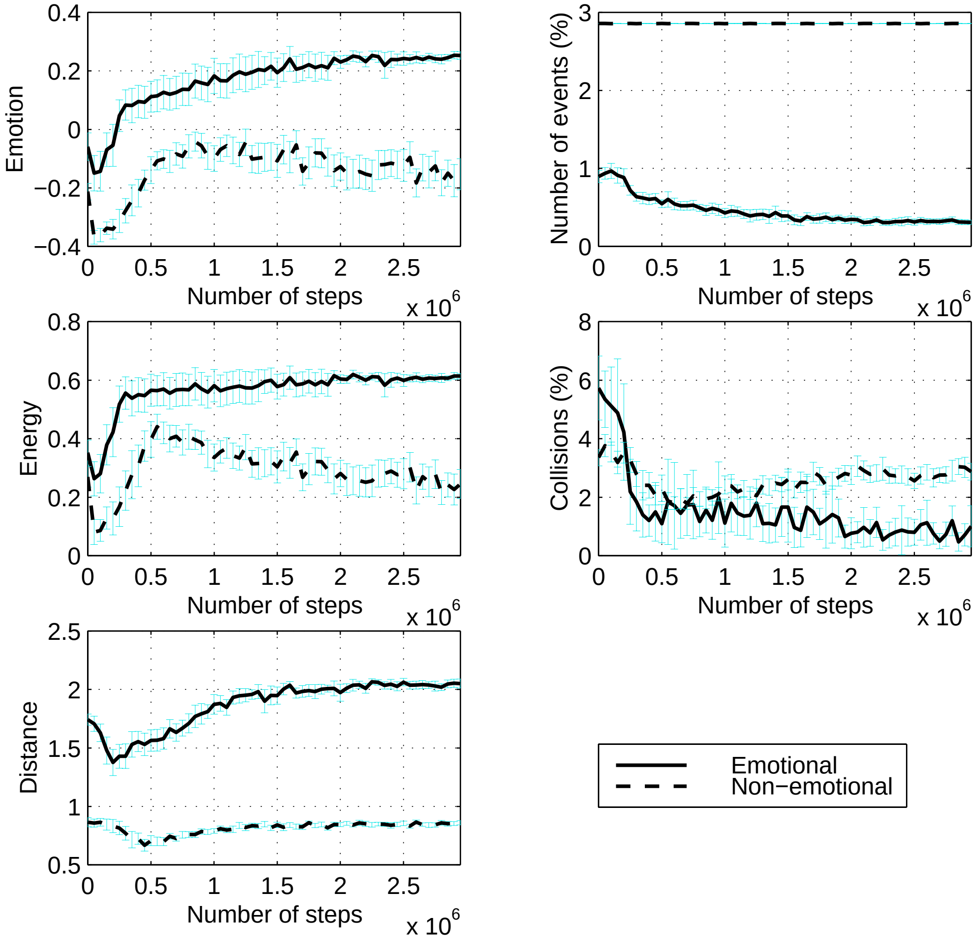


Figure 5.29: Emotional and non-emotional controllers performance in the more demanding environment and with harder energy requirements.

Controller	Reinforcement	Networks Inputs	Control Triggering	Figure
Emotional	Emotion-dependent	Feelings	Event-triggered	5.28
Non-emotional triggering	Emotion-dependent	Feelings	Interval-triggered	5.24
Non-emotional reinforcement	Sensation-dependent	Feelings	Event-triggered	5.22
Non-emotional perception	Emotion-dependent	Sensations	Event-triggered	5.26
Non-emotional	Sensation-dependent	Sensations	Interval-triggered	5.28

Table 5.5: Different mechanisms used by emotional and non-emotional controllers.

Controller	Emotion	Events (%)	Energy	Collisions (%)	Distance
Emotional	0.24 ± 0.01	0.5 ± 0.0	0.63 ± 0.01	0.6 ± 0.3	1.0 ± 0.2
Non-emotional triggering	0.21 ± 0.02	2.9 ± 0.0	0.62 ± 0.04	1.7 ± 0.1	0.9 ± 0.0
Non-emotional reinforcement	0.22 ± 0.02	0.5 ± 0.0	0.70 ± 0.02	1.6 ± 1.1	1.4 ± 0.1
Non-emotional perception	0.22 ± 0.03	0.5 ± 0.0	0.61 ± 0.03	1.2 ± 0.7	1.0 ± 0.2
Non-emotional	0.21 ± 0.02	2.9 ± 0.0	0.64 ± 0.04	1.4 ± 0.1	0.9 ± 0.0

Table 5.6: Summary of the comparison between emotional and non-emotional controllers. The means of the values and their 95% confidence interval obtained in the last half of the trials are presented.

controller over the non-emotional controller.

The emotional controller also has a slight advantage in terms of obstacle avoidance when compared with the other controllers, suggesting that the temporal synchrony between the different mechanisms might be a beneficial factor. Although the emotional controller suffers from intrinsic delays with respect to the robot-environment interaction due to the emotions' persistence, it is the only one where the learning controller input state or perception, the reinforcement and the triggering mechanism are in perfect synchronism.

5.6.3 Relative to the hand-crafted controller

The initial hope at the start of the experiments was that the learning controller would learn to behave similarly to the hand-crafted controller, or at least with the same level of reinforcement. However, this was not the case according to the results presented in Figure 5.24 and Table 5.4[11]:

Controller	Emotion	Events (%)	Energy	Collisions (%)	Distance
Emotional	0.24 ± 0.01	0.5 ± 0.0	0.63 ± 0.01	0.7 ± 0.4	1.0 ± 0.2
Hand-crafted	0.34 ± 0.01	6.1 ± 1.7	0.83 ± 0.01	3.0 ± 0.8	1.9 ± 0.1

Table 5.7: Comparing the performance of the emotional controller and the hand-crafted controller. The values presented are the means of the values and their 95% confidence interval of only the last ten test points.

[11] In the presentation of these results the emotional controller is referred to as event-triggered controller.

In this subsection, an analysis is made of the differences between the two in terms of both emotional states and behavioural preferences. For this purpose, a follow-up experiment was done for the emotional controller and another for the hand-crafted controller that recorded the required information. Next, attempts to replicate the hand-crafted controller's final behaviour are described.

Analysis of emotional states

A follow-up experiment was run for the emotional controller and another for the hand-crafted controller to analyse the differences in emotional states between the two. Results are shown in Figure 5.30. Although the distribution of the emotional states of the hand-crafted controller is quite stable, the same does not happen with the emotional controller and the results shown should be only taken as an indicative sample. Even so, some general conclusions can be drawn.

The two graphs represent two distinct distributions of emotion states: during all steps and during the steps where an event was detected. The differences between these two distributions account for the differences between reinforcement and event reinforcement found in both controllers[12]. Although, in general, negative emotional states are not frequent, they are much more frequent during events. This is particularly noticeable in the case of the hand-crafted controller which is also the controller that presents more substantial differences between the two types of reinforcement. In the case of this controller, the FEAR emotion is almost as frequent as the HAPPINESS emotion during events, suggesting that a considerable number of the events consist in the detection of collisions via the FEAR emotion. The fact that events are triggered by emotion states that tend to be more negative explains why reinforcement during events is lower than average. It is also clear that the excessive number of events of the hand-crafted controller is partly due to its large number of collisions. In these particular follow-up experiments the total number of events of the hand-crafted controller was 1512 against the 470 events of the emotional controller.

The emotional controller is different from the hand-crafted controller in that it is HAPPY less frequently and is often more SAD and ANGRY. The occurrence of the negative

[12] These differences in reinforcements were observed in Figures 5.8 and 5.14.

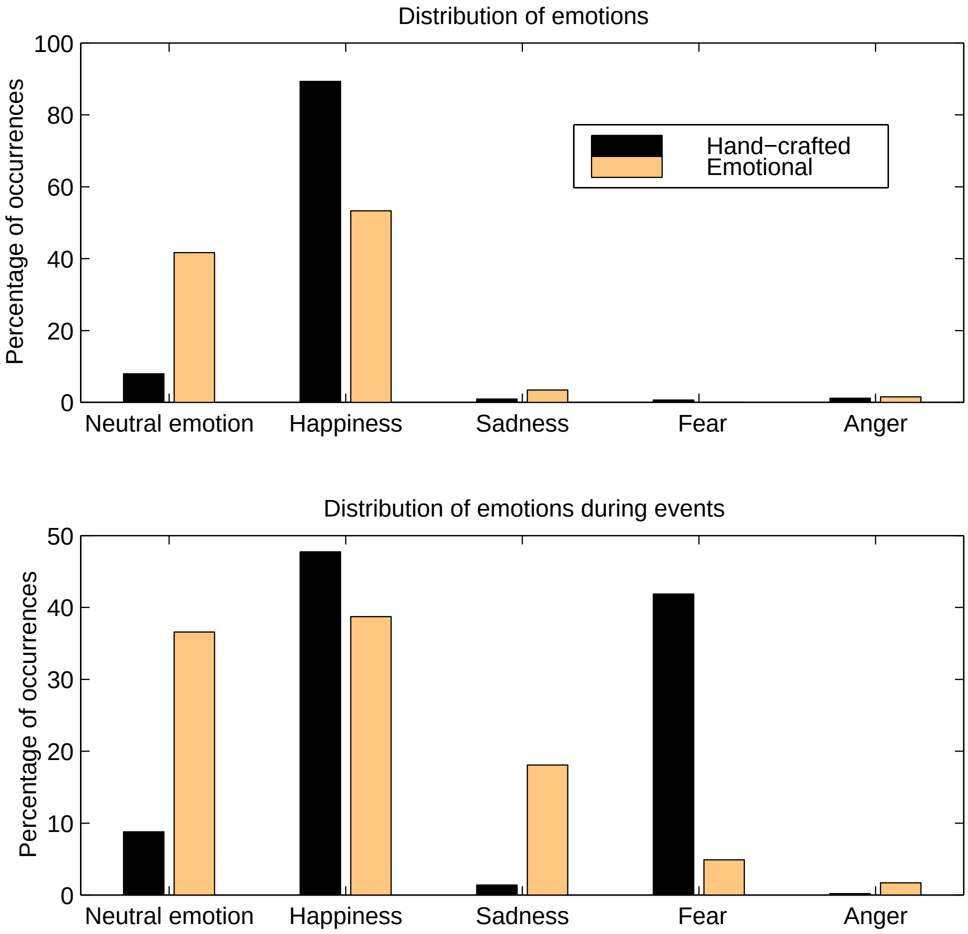


Figure 5.30: Occurrence of each emotion.

emotions can simply be due to the fact that the learning controller at times will have to test its policy by exploring behaviours other than the best one. In fact, the controller needs to be punished in order to know what it should and should not do. Another reason for the differences might be that the learning controller is concentrating its efforts in not bumping, in which it is more successful than the hand-crafted controller, and that has a detrimental effect on the rest of the problems it has to handle. The fact that the learning controller is HAPPY less often is probably due to not being so persistent in the wall-following behaviour and therefore not having the temperature high so often.

The differences between the two suggest that the emotional controller's behaviour is temporally and spatially more local. The lack of a global picture of its interaction

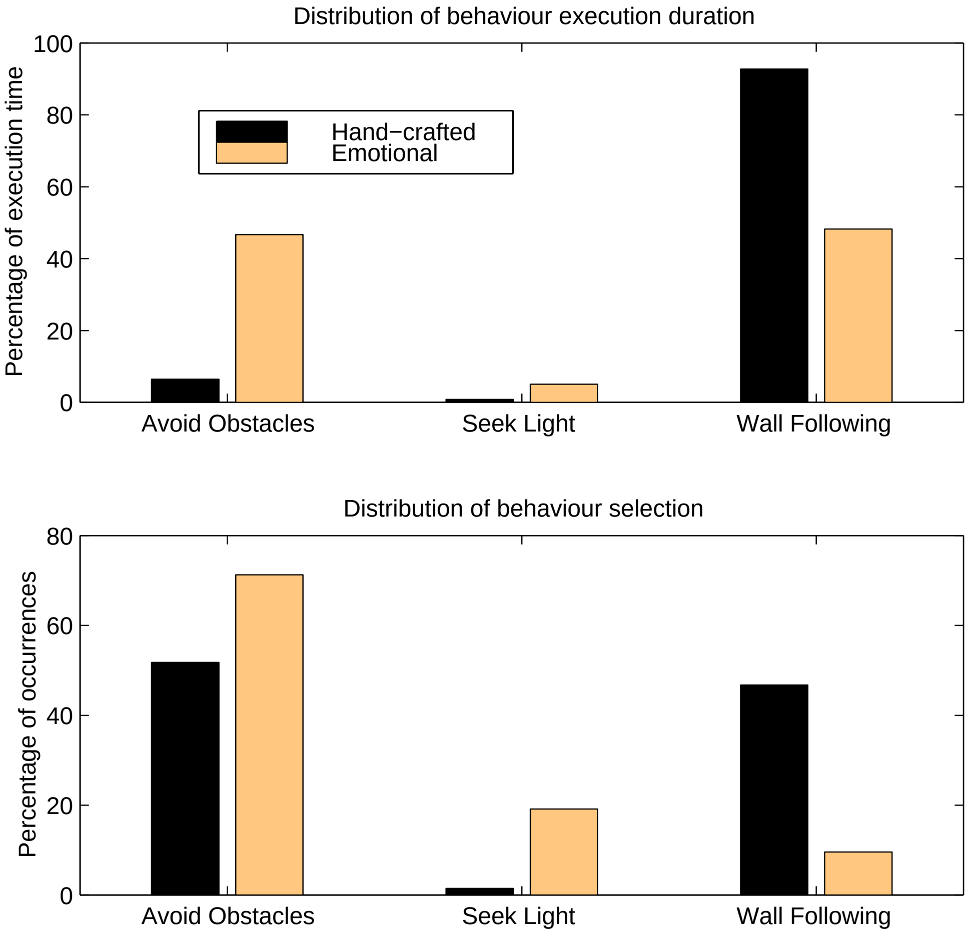


Figure 5.31: Occurrence of each behaviour.

with the environment will make it concentrate its efforts on more immediate problems, namely, by solving the obstacle avoidance problem particularly well, and eating when it is already SAD or trying another behaviour after ANGER has manifested itself.

Analysis of behaviour preferences

In the follow-up experiment of the emotional controller and the hand-crafted controller the behaviour selections were also recorded. The distribution of the selection and actual execution steps of each one of the behaviours is shown in Figure 5.31.

The differences between the distributions of each graph show that the wall following behaviour tends to be performed for long periods of time with no interruption. The other two behaviours are more likely to cause events, namely by quickly achieving their purposes.

The differences between the two controllers show, as was suspected before, that the emotional controller does not wall-follow as much as the hand-crafted controller and that the emotional controller will select the seek-light behaviour much more often. Attending to the fact that the emotional controller's energy level is lower (see Table 5.7), the emotional controller is probably having difficulties acquiring energy efficiently.

Learning the hand-crafted behaviour

In the previous sections, the differences between the emotional controller and the hand-crafted behaviour have been highlighted. In this section, several attempts at trying to transfer the knowledge of the hand-crafted behaviour into the learning controller are described. Different methods were tested to try to mimic the hand-crafted behaviour by the learning controller, but none with much success.

The first attempt consisted of having the networks learn during the normal robot simulation while the hand-crafted controller was controlling the robot. A more sophisticated attempt required saving all the details of a hundred thousand step experiment using the hand-crafted controller and then repeatedly going over the recorded experiences to learn them with the emotional controller. Randomising the order of these hundred thousand single experiences actually helped the neural networks a bit, but the results for all these methods were consistently unsatisfactory. The main problem with these methods is that the learning controller does not have a chance to learn the results of bad behaviour. The answer to this problem was to slightly punish the non-selected behaviours apart from attributing the received reinforcement to the selected behaviour. This procedure assumes that the hand-crafted controller has selected the right behaviour and the others are inappropriate for that particular situation. Even so the learning algorithm was lacking a wider range of experiences. The intrinsic random nature of the emotional controller, even when it is not learning, will always allow the controller to step into situations that are outside the normal range of the hand-crafted behaviour.

So a radically different approach was taken that consisted of directly training the neural networks with random inputs taken from a uniformly distributed input space. Each experiment consisted of 5 million steps separated by tests done at intervals of a hundred thousand steps. At each step a random neural network input vector is determined. Then the behaviour selected by the hand-designed controller for this input vector is determined. The networks are trained with a target value of 5 or -5 depending on whether their associated behaviour is the one selected or one of the others. During each test a hundred thousand different random input vectors are selected. For each input vector the hand-crafted behaviour selection is determined and compared with behaviour whose neural network has the highest value for that particular input vector. The error is the percentage of behaviour mismatches. The results for 5 different trials are presented in Figure 5.32.

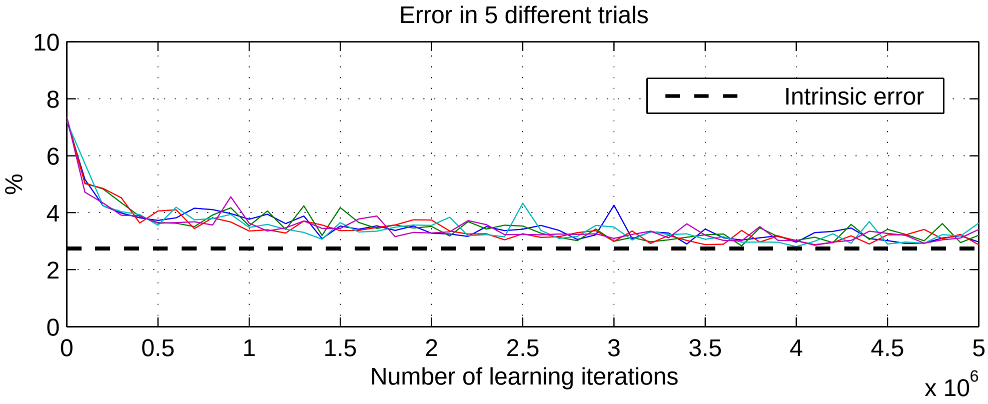


Figure 5.32: Neural networks error when knowledge is directly transfered to them from the hand-crafted controller.

It should be noticed that a deterministic procedure such as the one of selecting the network with the highest value will never reach a zero error value, because the hand-crafted controller is not totally deterministic itself. In fact, when the hand-crafted controller is evaluated against itself it returns an error of about 2.7%, also shown in the figure. The neural networks' error is not much more, with a mean of 3.1% in the last 10 tests. The error starts off quite low, only 7.3%, because the algorithm for selecting the better-ranked behaviour gives preference to the avoid-obstacles behaviour, *i.e.* this behaviour is selected when all the networks' outputs are the same value as it is at the start of the learning[13]. This reduces the starting error because the avoid-obstacles

[13] This problem was never corrected because as soon as the neural-networks start learning it disappears.

behaviour is also by far the most frequent choice of the hand-crafted controller in a uniform distribution of the input state.

It was found that a uniform distribution of the input space is very different from the distribution of the input space experienced by the robot. Namely, the distribution of the behaviours selected by the hand-designed control procedure is very different from the one obtained in normal robot simulation. For a uniform distribution the percentage of selection of the avoid-obstacles behaviour is much increased and much higher than that of any other behaviour, including the wall-following behaviour which is reduced drastically. So it is possible that the networks are being over-trained with situations that will never even be found by the robot. It is also possible that the behaviours suggested by the hand-crafted controller outside its normal range of execution are not the most appropriate.

After these networks had been set up they were put to the test, by using them in the emotional controller in a normal simulation experiment. Two experiments are reported in Figure 5.33, one with learning and one without. The results for the hand-crafted controller are also given in the figure for comparison. Both experiments use a ξ value of 1.5, which is the most adequate value for the hand-crafted controller[14].

From the non-learning controller's performances we can observe that the networks' previous experience was not very helpful. The non-learning controller's performance soon starts to diverge from the normal hand-crafted controller and it will continue to deteriorate right through to the end of the experiment. The randomness introduced by the emotional controller can easily take it away from the normal scenarios dealt with by the hand-crafted controller. Once out of its domain of expertise, it is natural that its performance deteriorates.

The learning controller overall performance, measured by emotion value, also diverges away from that of the hand-crafted controller, but will converge to a level in between the hand-crafted performance and the normal performance of an emotional controller without pre-trained neural networks. See Table 5.8 for a short summary of the performance of the three.

[14] The results for ξ valued 2 and 2.5 are shown in Appendix D.4.

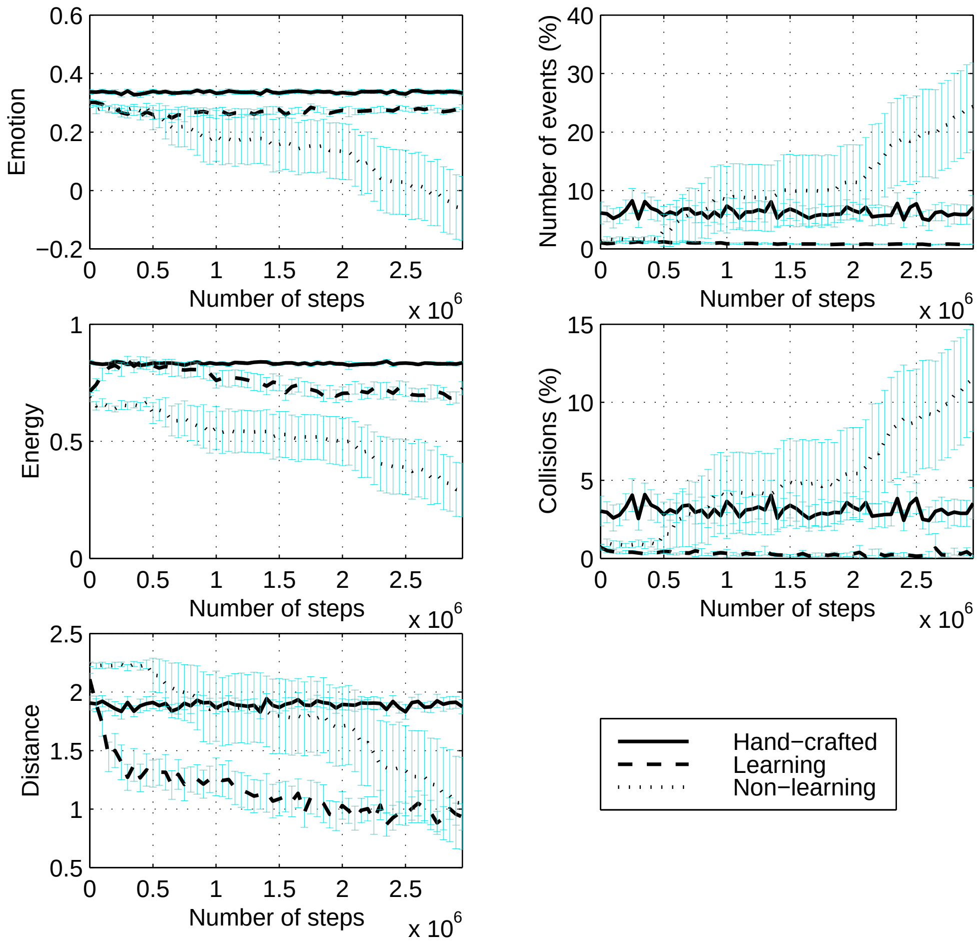


Figure 5.33: Emotional controller starting off with customised neural networks. The performance of the hand-crafted controller, *i.e.* the target, is also shown.

Controller		Emotion	Events	Energy	Collisions	Distance
		(%)		(%)		
Emotional	normal	0.24 ± 0.01	0.5 ± 0.0	0.63 ± 0.01	0.7 ± 0.4	1.0 ± 0.2
	pre-trained	0.28 ± 0.01	0.8 ± 0.1	0.70 ± 0.03	0.3 ± 0.2	1.0 ± 0.1
Hand-crafted		0.34 ± 0.01	6.1 ± 1.7	0.83 ± 0.01	3.0 ± 0.8	1.9 ± 0.1

Table 5.8: Comparing the performance of the normal emotional controller and the one with pre-trained neural networks. The values presented are the means of the values and their 95% confidence interval of only the last ten test points.

One of the main reasons for the poor results obtained is the fragility of the hand-crafted controller. The limitations of the hand-crafted controller have been discussed previously during its presentation in Section 5.3.4 where it is shown how the performance of this controller is strongly dependent on the event triggering mechanism in use. The development of this controller was an arduous process that even required engineering the environment to discard hazardous environmental locations. However, the results also point to deficiencies in the capacity of the emotional controller to learn to perform the task just as the hand-crafted controller does. It is possible that the behaviour-selection procedure used by the hand-crafted controller is not representable by the neural-network architecture used by the emotional controller.

5.6.4 Temporal persistence of emotions

For the action-based controller, the most severe drawback of the emotion system was the temporal persistence of emotions introduced by the hormone system. This suggests that the hormone system might also strongly influence the performance of the present controller.

The parameter most responsible for emotional persistence is the hormones' decay rate. In this subsection, its influence on the emotional controller is examined.

Figure 5.34 shows how the hormones' decay rate influences the emotional response. The values of decay rates examined in the figure are actually the ones used in the robot experiments shown in Figure 5.35. Although these parameters change the emotional response significantly, their influence in the emotional controller's performance is not noticeable.

All experiments of this subsection were done in the more demanding environment and with harder energy requirements[15]. This way the controllers are all tested in the most adverse circumstances available, allowing the differences between them to be more easily noticed.

In reality, the emotional controller is quite robust to changes in the temporal characteristics of the emotional system. Even if the hormone system is totally removed,

[15] Consult Section 5.3.5, for details.

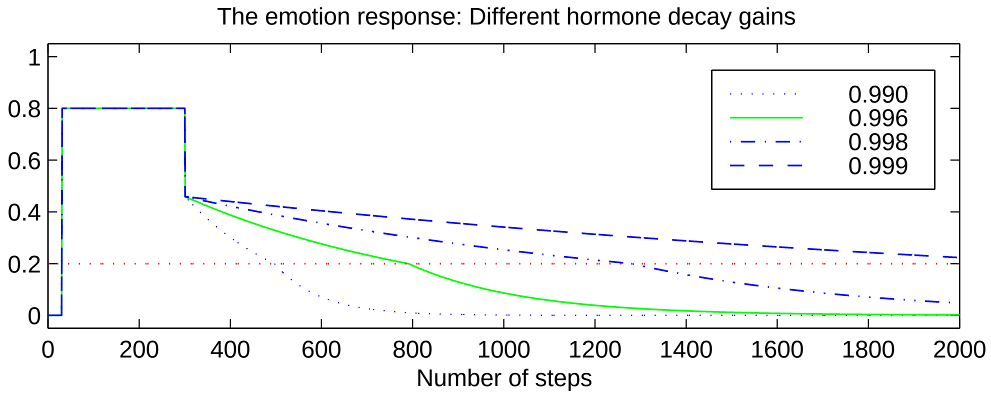


Figure 5.34: Emotional temporal response with different hormone decay rates. The default value for this parameter in previous experiments has been 0.996.

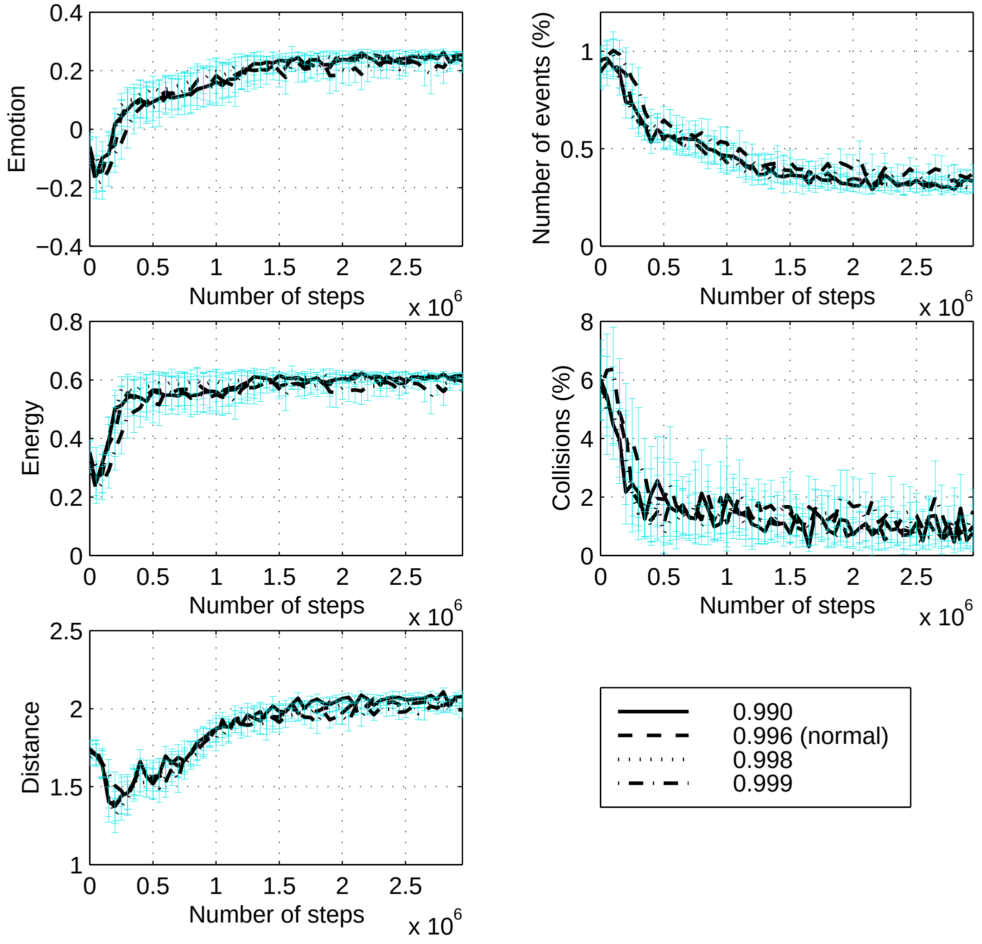


Figure 5.35: Emotional controller with different values for the hormone decay rate.

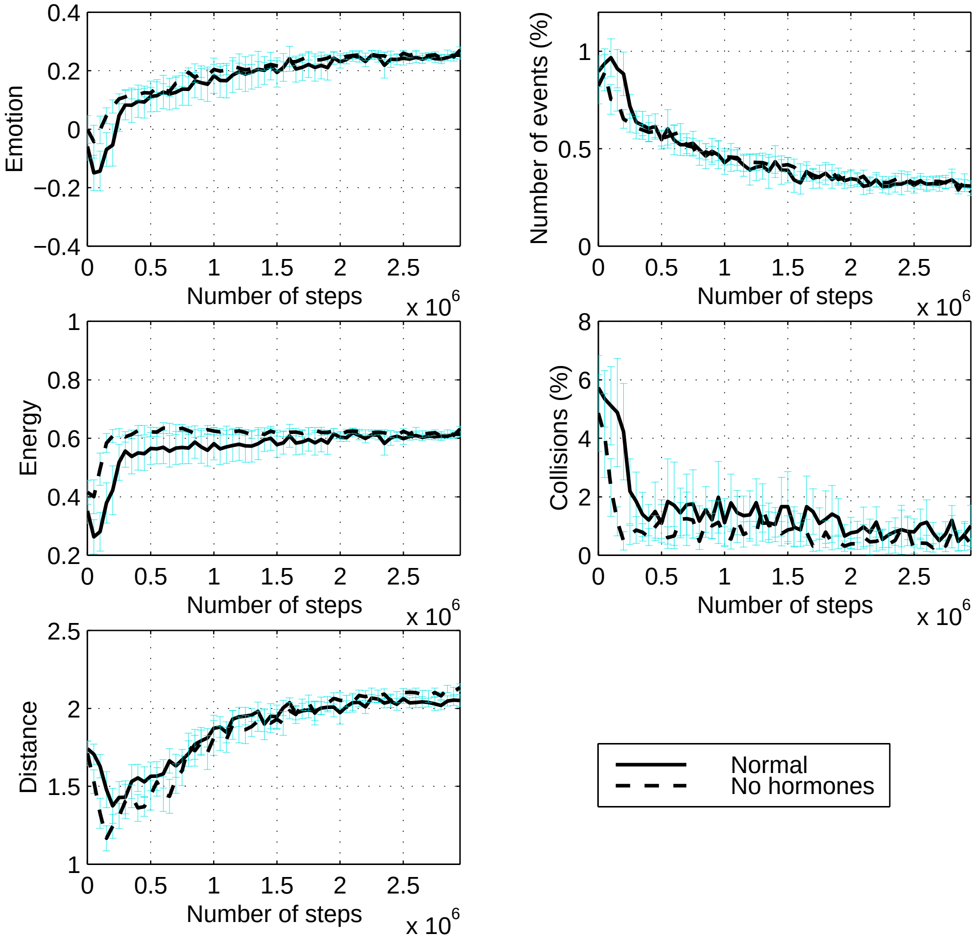


Figure 5.36: Emotional controller with and without hormones.

taking away all the emotions' persistence, the emotional controller's performance is not affected. Figure 5.36 demonstrates this point.

5.6.5 Exploration strategy in behaviour selection

In reinforcement learning there is a fundamental trade-off between exploration and exploitation of the policies learned. On the one hand, too much exploitation can lead to sub-optimal policies. On the other hand, the agent must exploit its knowledge at some point.

In the present controller, the relatively simple approach of Boltzmann exploration was taken. Another reasonable ad-hoc strategy that is widely used in reinforcement learning is p-probability exploration[16]. It consists of selecting an action at random with a probability p and otherwise taking the action with the best expected reward. This strategy has the advantage of being simpler, but unfortunately is inappropriate for the present controller as we shall see below.

The main disadvantage of this strategy is that it does not take into consideration what is known about the expected rewards of each behaviour during exploration. So it will equally select between behaviours that have proven to be promising in the past and others that are clearly hopeless.

The Boltzmann exploration allows the agent to explore more when the behaviours are similarly ranked and exploit more when one of the behaviours appears to be much more appropriate than the others. Furthermore, as the controller explores its environment and gathers knowledge about it, its preferences will grow stronger and the exploitation will increase. This effect can be observed in Figure 5.37 where an account was made of the number of times the controller did not select the best ranked behaviour in the course of a normal experiment.

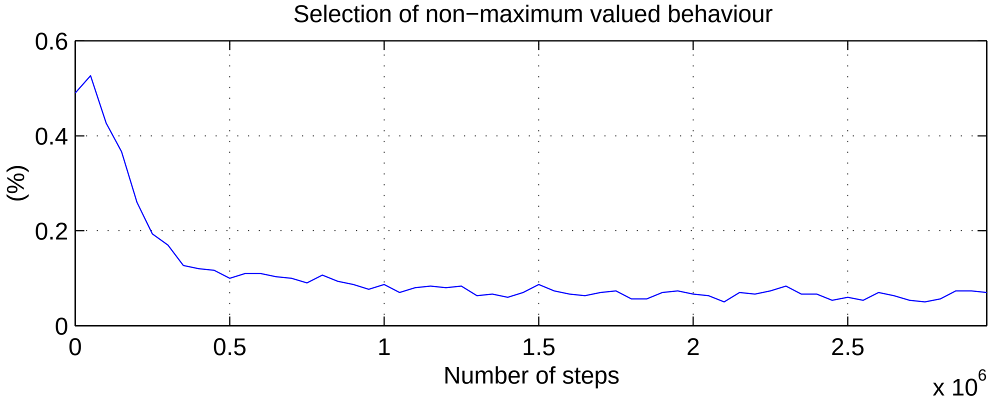


Figure 5.37: Record of how often the emotional controller does not select the best behaviour.

Another interesting advantage of the Boltzmann exploration strategy is that it does not implicitly assume the existence of a single optimal behaviour at each point. This is particularly advantageous in the case of the robot's specific task which demanded that the hand-crafted controller itself use a random generator for behaviour selection.

[16] Designated the ϵ-greedy strategy by Sutton and Barto (1998).

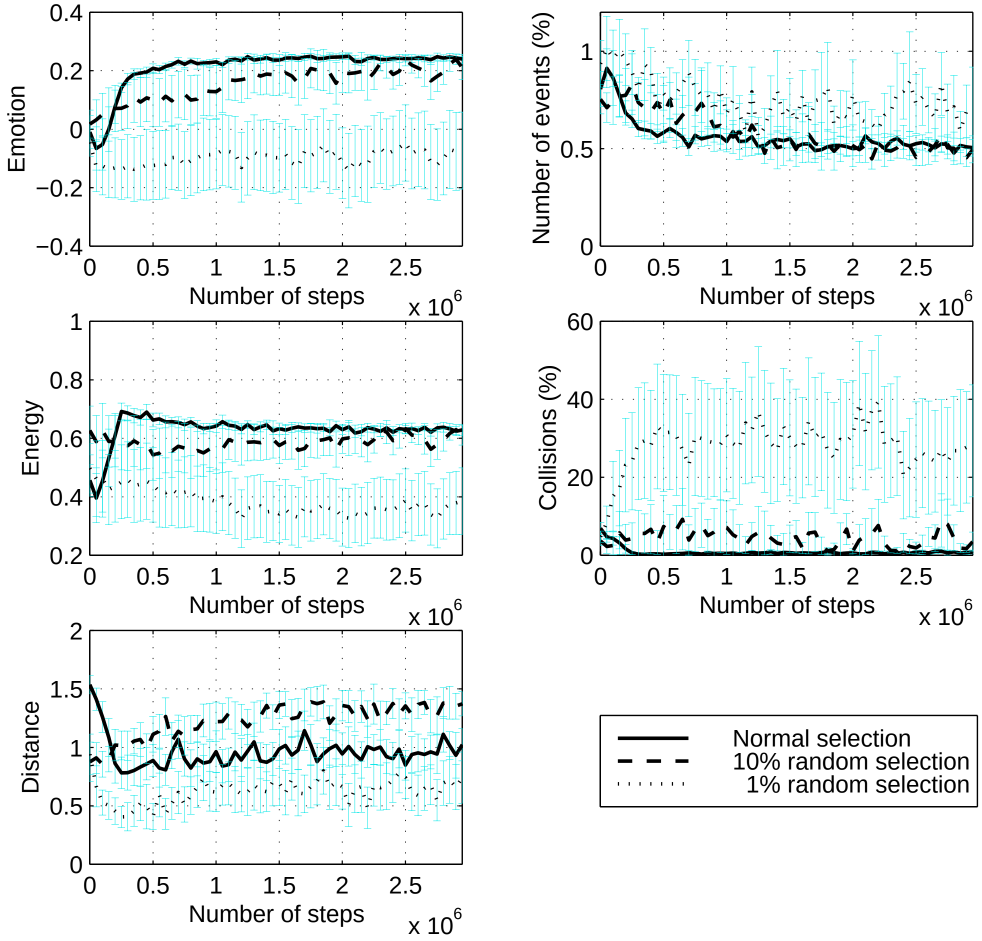


Figure 5.38: Different exploration strategies.

This was found necessary for specific environmental situations which would otherwise lead to dead-locks.

The experimental results in Figures 5.38 and 5.39 confirm the inadequacy of the p-probability exploration. In the first instance, experiments were done for two values of p: 10% which is the value most often used in the literature and 1% which is more similar to the level of exploration provided by the default Boltzmann exploration strategy. When compared with an experiment using the Boltzmann strategy, the simpler scheme performs worse. The 1% exploration is particularly ineffective, probably because it does not allow for enough policy exploration. The use of a more demanding environment and harsher energy requirements made the poor performance of the 10% exploration particularly evident (see Figure 5.39).

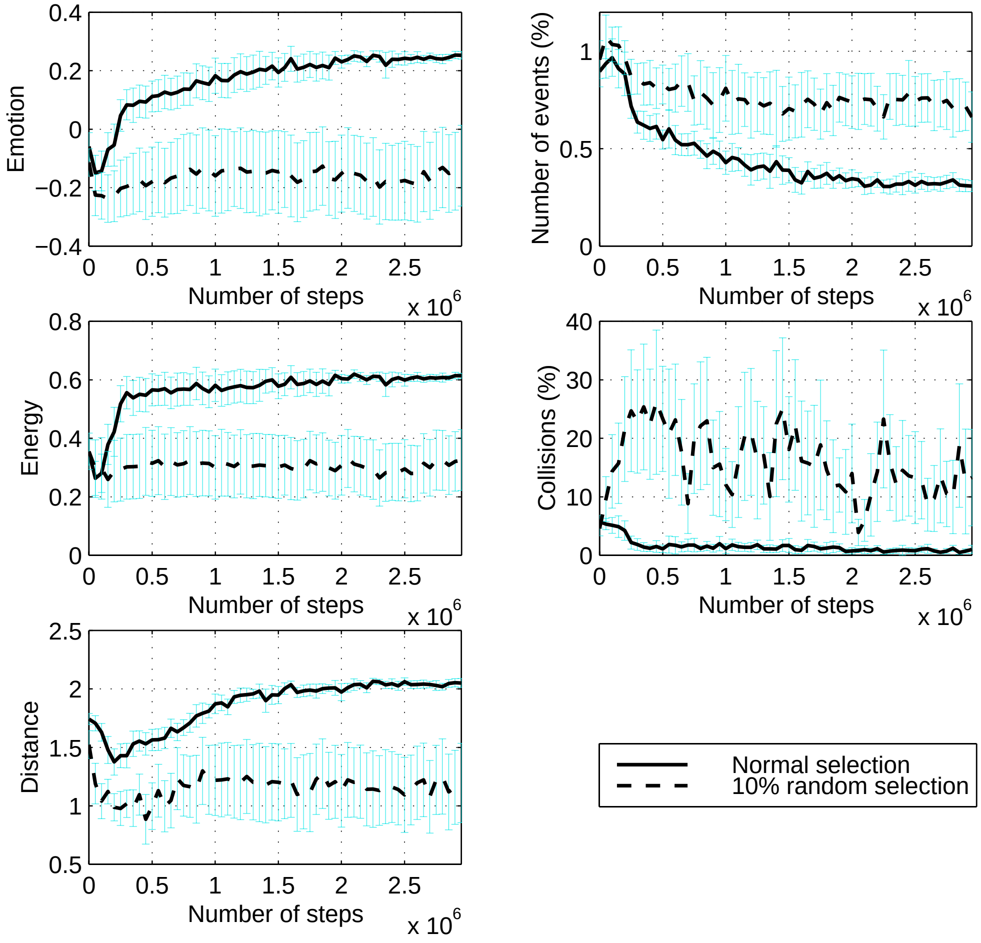


Figure 5.39: Different exploration strategies in the more demanding environment and with harder energy requirements.

The exploration *vs.* exploitation issue is particularly relevant in systems that are supposed to learn all the time, *i.e.* without an artificial division between a learning phase and an execution phase. In this case, the agent has to make the best of its knowledge to know how and when to explore, because it will not have available a separate execution phase where the randomness of its choices can be eliminated in favour of exploitation.

The exploitation should increase as the agent learns about its environment. Some researchers will decrease the temperature value in the case of the Boltzmann exploration or the p in the *p*-probability exploration to achieve this effect. The problem with this is that it assumes that the environment conditions will not change and that the agent will learn all it needs in a certain pre-defined amount of time. In the absence of a more sophisticated exploration strategy, the decrease of the exploration as a side-effect of the Boltzmann exploration strategy seems more natural and preferable for autonomous agents' learning.

5.7 Conclusions

Experiments showed that emotions can be used as an attention mechanism at different levels of a reinforcement learning task:

- making more evident the relevant aspects of the environment, *i.e.* those directly related with the current emotional state, by influencing the robot current state through the hormones;
- providing a straightforward reinforcement function which works like a powerful attention mechanism in a reinforcement learning task by attributing value to the different environmental situations;
- determining the occurrence of the significant changes in the environment that should trigger state transition, by looking at sudden changes in the emotional system state.

These were three different mechanisms that worked well experimentally. Each one of them had different levels of performance when compared with alternative methods.

No significant differences were found in using emotion-dependent perception, *i.e.* making the emotionally relevant aspects of the environment more salient, or not. This result might be task dependent but is certainly controller dependent because the learning controller used can easily ignore the differences in magnitude of the input values introduced by emotions by compensating for them with changes in the neural-network's weights. A proper assessment of this emotion role would benefit from the employment of a controller equipped with proper mechanisms of attention for input processing, *i.e.* a controller where different weights could be given to the analysis of the different inputs.

It was found that emotion-dependent reinforcement is adequate for behaviour-based control and that the non-linearities of this reinforcement function have an active role in its success.

The behaviour-based controller provides more appropriate time scales than action-based control for the use of emotion-dependent reinforcement. The difference between behaviour-based and action-based control is not only restricted to temporal duration. The behaviours themselves are by nature distinct from simple actions. They are not defined by constant motor values, but rather by a simple reactive "goal" that determines the motor values at each step as a function of the agent's current perception. This allows them enough versatility to run for longer durations of time which is their main advantage for use with emotions. The persistence of emotions over time in natural systems also suggests that they should be related to a higher level of decision-making which does not rely on simple primitive actions but on complex action patterns more suitably expressed at a behavioural level.

The emotion-dependent event detector was very successful. It allowed drastic cuts in the frequency of triggering of the learning controller while maintaining overall performance. This can be particularly advantageous in the case of very time-consuming learning controllers, where each triggering of the controller can result in a significant loss of precious real time. These results were obtained with both sensation-dependent reinforcement and emotion-dependent reinforcement.

A later analysis of the distribution of the emotional states of the emotional controller in general and in the particular situations where events were detected, showed that the robot control is triggered more often in adverse situations. This has adaptive value, because it arouses the agent's attention to the need to change behaviour when the current behaviour becomes inappropriate. Furthermore, behaviours that have some immediate goal like avoid-obstacles and seek-light tend to have shorter durations because they are terminated the moment their goal is reached.

After each of these mechanisms was evaluated on its own, the emotional controller as a whole was also evaluated.

This controller proved to be more successful than an equivalent non-emotional controller where the emotional mechanisms were replaced by their non-emotional counterparts. The triggering mechanism was strongly responsible for the difference in performance between the two, but the joint use of all three mechanisms also seems to provide an advantage.

Nevertheless, the final emotional controller exhibits a performance that falls short of the one achieved by a hand-crafted controller. Despite all the efforts to reproduce the hand-crafted behaviour with the emotional controller this was not accomplished. The conclusion reached was that the hand-crafted controller behaviour was very fragile and could not be achieved by the particular learning architecture used. The permanent element of randomness present in this architecture would move the robot to scenarios where the competence of that behaviour would quickly deteriorate.

Finally, the emotional controller was found to be robust to different degrees of emotion persistence and strongly dependent on the exploration strategy.

Chapter 6

Concluding Discussion

6.1 Introduction

The work reported in this dissertation consisted in bringing emotions to the field of autonomous robots following an animat approach. In opposition to more traditional approaches, the emotional agent deals with a continuous and non-symbolic environment and has to adapt to its environment through learning. Emotions were mostly used to help it in its learning task by providing the domain-dependent mechanisms necessary to fulfill key reinforcement-learning components.

To test the feasibility of the approach a body of experimental work was carried out in a realistic simulated robot which tested the integration of several emotion functions in a reinforcement learning framework. Experiments compared these functions with other more traditional reinforcement-learning approaches while always looking after the several problems posed by robot autonomy.

This work required the development of an emotional model adequate for the animat approach. The model was designed to cope with a non-discrete world and to be suitable for integration in a simple, but complete, robot control architecture. In fact, the emotion system itself was found useful for the temporal segmentation of the world.

The developed and empirically tested emotion model has the characteristic property of directly influencing the perception of the agent through a hormone system. This allows emotions to colour the agent's perception by focusing its attention on the features of the environment that are most congruent with the agent's dominant emotion. Furthermore, this hormone system also endows the system with persistence of emotions through the near future and avoids sudden swings of emotional state. Modelling the persistence of emotions uncovered difficulties in the use of emotions as reinforcement that were hidden away by the fact that most existing emotion models do not incorporate inertia.

The persistence of emotions modelled in the experiments introduced some restrictions in the effective use of emotions in robot control. Experiments showed such an emotion system can be used much more successfully in the context of behaviour-based control than action-based control. In particular, the role usually attributed to emotions of providing an evaluation of the state of the world was particularly unsuccessful in a action-based controller. The time-scales involved in the execution of a behaviour proved to be more appropriate for emotion-dependent reinforcement than the smaller time-scales associated with the execution of single primitive actions. In fact, the behaviour-based controller was quite robust to variable degrees of emotion persistence. Therefore, results seem to point to the intuition that emotion-level information is more relevant to higher level control, such as behaviour-based, than to lower level control, such as based on primitive actions.

This introduction is only a brief summary of the achievements of the work carried out for this thesis. The next sections provide a more detailed discussion of the different issues involved in this work both in terms of autonomy and emotions. This is followed by the presentation of some suggestions for future work. The use of emotions offered a new perspective over autonomous learning which grounds the final conclusions drawn in this dissertation.

6.2 Issues in Autonomous Learning Robots Research

6.2.1 Design guidelines

In Section 2.2.8, a few guidelines were established for the design of autonomous robots. These guidelines were followed to some extent in the design of the autonomous learning robot presented in this dissertation:

Perception — Unfortunately the selected robot simulator did not provide the robot with very rich sensory input, yet attempts were made to endow the agent with some attention mechanisms;

Movement — The movement required in the behaviour-based control experiments involved both moving around in the environment and interacting with food sources, but although the global robot behaviour had some degree of complexity the number of available actions, or sources of different movements, at any one time was relatively small;

Homeostatic Goals — The agent was given several homeostatic goals in the definition of its task: maintain energy level, move around, avoid obstacles;

Reactions and Learning — the selected learning architecture permits fast reactions while providing the means for the robot to adapt to its environment;

Navigation — the robot has limited navigation capabilities but still manages to travel from light to light in order to obtain food.

Taking into consideration the state of the art in robot technology and autonomy, the developed robot exhibits a fair amount of autonomy. However, the simplicity of its controller and its limited capacity to learn complex behaviour constrain its autonomy. In particular, the network-based architecture has a tendency to forget important but rare experiences easily and shows difficulties in correctly differentiating different experiences. Furthermore, if the agent is changed to a different environment it will be able to adapt to it but, in the process of learning, it will forget what it had learned specific to the previous environment.

6.2.2 Design problems

The learning architecture selected has a few disadvantages in terms of autonomy achievement that have been pointed out previously in Section 2.5.2, yet those were not subject to investigation. An exception was made for the examination of the exploration strategy of the learning architecture, which proved to have an important role in the success of the robot. The limitations of the simpler p-probability exploration or ϵ-greedy strategy were demonstrated in Section 5.6.5.

Most of the problems identified in the development of the autonomous learning robot were related with the learning controller's open specifications detailed in Section 2.5.3, which were central to the topic of this thesis: emotions interacting with control. These problems, which were often responsible for robot learning failures are discussed next in terms of those specifications.

Reinforcement function

As expected, the learning algorithm was very sensitive to the reinforcement function. Employing emotion-dependent reinforcement was not straightforward. The lack of synchrony between reinforcement and local goal achievement proved fatal in the domain of the action-based controller. The reinforcement delay was not properly handled by the learning algorithm probably because this was unable to effectively propagate rewards across long sequences of actions.

This hypothesis was confirmed by the experiments with behaviour-based control. The decomposition into behaviours permitted an increase of the number of physical steps between learning iterations which allowed a reduction in the reinforcement delay in terms of number of learning iterations and enabled an effective propagation of rewards and punishments.

The emotion-dependent reinforcement function designed has the characteristic of only taking at most one emotion into consideration at each time step and ignoring the reinforcement information provided by all the non-dominant emotions. This introduced non-linearities in the reinforcement function procedure that not only did not impair the performance of the learning algorithm, but were actually used advantageously.

Action set

One interesting point that was uncovered by the experiments was that providing the agent with a complete action set, in the sense that it can reach every state of its environment, might not be enough. In Section 4.3.4 it was shown that the use of a slightly different complete action set can result in a significant drop in learning performance. In this particular case the existence of a single action, moving backwards with a twist, that permitted the robot to ward off obstacles, proved to be important. This was probably due to a question of hidden space: once the robot has simply backed off an obstacle and its sensors do not register it anymore, the robot can easily select a forward movement that will lead it to the same obstacle. The problem is that the robot has no available memory to inform it that the obstacle is there and that other actions should be taken.

The behaviour complexity achieved by the learning algorithm through the use of an action set composed of primitive actions is limited. It is clear that it is necessary to have some kind of hierarchical solution if complex behaviour is intended. The simplified solution found for the experiments was to add competence to the action set by replacing the primitive actions by behaviours. This is a very rigid solution that by itself does not provide much in terms of added complexity. Solutions that enable the robot to learn its own hierarchy of behaviours allowing it to organise by itself its task in sub-problems to solve, like the reinforcement-learning system proposed by Digney (1998), are much more flexible and are probably the correct route to robot autonomy. Unfortunately the solution proposed is still at a rudimentary stage and is not yet applicable to robot domains.

Input state

The particular algorithm used was quite robust in terms of sensory input. The influence of emotions on the state input was largely ignored by the system.

It was thought that the short term memory provided by this emotional influence might be helpful to solve the hidden state problem for obstacle avoidance but apparently this was not the case. However, the concordance in delay between sensory input and reinforcement seemed to help slightly.

State transition

Although not much importance is usually given to this issue and people usually resort to domain-specific solutions that artificially constrain the learning algorithm, the system proved to be particularly sensitive to the definition of state transition used.

The simplest solution of defining a state transition at each step proved particularly disastrous for the behaviour-based controller. Furthermore, if intervals were used instead, the performance of controller was strongly dependent on the interval size. Nevertheless, the failure of a step-triggered controller can be partly task dependent, because this agent has to persist in its action to travel from one light to another.

The proposed solution based on the detection of significant changes in the input state was quite successful. Unlike other work in the field, the detection of changes in the input state was dependent on the robot's dominant emotion and therefore intrinsically related with its reinforcement.

Meta-control variables' values

There were no problems in finding suitable values for the two major learning parameters of the learning algorithm: back-propagation learning rate and action selection temperature. Nevertheless, it seems inadequate having to tune these parameters *a priori*, *i.e.* there should not be a need to run preliminary experiments to explore different values for these parameters. Furthermore, the robot would certainly benefit from being able to change them on-line. The autonomous learning of complex behaviour probably requires that the agent is able to determine when it needs new skills and when it should simply use the skills it has and avoid forgetting them by over-learning.

6.2.3 Evaluation methods

Test procedure

To avoid the problems concerning evaluation discussed in the reinforcement-learning review (Section 2.4.2), the controllers reported in this document were evaluated in different trials and using different evaluation mechanisms.

Other issues concerning the evaluation of autonomous learning agents were raised further along the dissertation.

The main one was whether the agents should or should not be allowed to learn during the testing phase. As autonomous learning agents, they should not have a distinctive performance phase with learning turned off, but be able to learn throughout their lifetime. Learning can cease if the agent itself so determines, but that should not be an external decision. This suggests that learning should not be externally turned off for tests. However, a more through evaluation of a certain learning stage may demand that the controller be evaluated in different situations while still in that stage, *i.e.* with no learning in between. Moreover, if the robot is continually learning it is difficult to evaluate whether the learning controller is actually acquiring long term knowledge or just temporally learning to solve the immediate problems it is faced with. On the other hand, this last ability can be considered an advantage of the continuous learning agent that should be taken into account in the evaluation; and instead be taken as one more reason for not turning learning off.

Experiments revealed that either testing method resulted in similar evaluations for the present controller. This demonstrated that the agent is actually taking advantage of long term knowledge and that it is able to maintain its performance while learning.

This last point is important for a learning agent that does not have the advantage of having its exploratory learning mechanisms turned off for the execution of its task. This characteristic of autonomous learning agents also raises the important point of learning stability. These agents should be subject to long tests so that possible stability problems of the learning algorithm can be detected.

In the particular task chosen for the experiments, the performance criteria are such that the performance of the agent suffers a deterioration in time if the agent does not successfully learn its task. This is typical of a survival task where the non-observance of a certain number of subsistence behaviours can lead to a decrease in the welfare of the agent. In these cases, the performance may not be required to reach a maximum value but to be maintained at an adequate level during the agent's lifetime. This means that a proper evaluation of the learning algorithm requires a comparison of its performance with other non-learning algorithms and eventually taking the non-learning counterpart as a base in the presentation of the results. Only this way can the learning abilities of the agent be properly tested.

Unsuccessful experiments

Usually people tend to write only about successful experiments, but unsuccessful experiments can also provide valuable knowledge. It is important to report on innovative designs that are bound to be useful, but is also important to point out their limitations and capacity to adapt to different circumstances. In particular, it is important to mention which were the most crucial design issues so that others can avoid the trouble of rediscovering them through failures. This was the philosophy followed in this dissertation and the reason why a considerable part of the presented results were negative.

Simulation *vs.* real world

The experimental work reported in these dissertation was done in a simulated robot instead of in a real robot. There were a few reasons for this choice:

- longer experiments are possible;
- the evaluation of different control strategies is less time-consuming;
- much more data can be extracted from the experiments for analysis of the results;
- experiments can be reproduced, making it possible to answer particular questions that were not contemplated in the first instance of the experiment;
- environment and task can be easily modified;
- there is more freedom in the agent design, for instance sensors that are readily available for the physical robot can be easily provided to the simulated robot.

These are all significant advantages particularly for research that is still in its early stages. Even so many robotic researchers tend to consider robot simulation unworthy. The reason for this is the simplifications that are necessarily introduced in robot simulation, which can introduce unrealistic simplifications of the robot-environment interaction that can both hinder the development of simpler solutions and offer solutions grounded on specious abstractions.

Sometimes robot controllers are faced with control problems in simulation that disappear once the controller is changed to the real world. For instance, real world noise can help to take the robot out of what in simulation are unsolvable dead-lock situations (*e.g.*, Mahadevan and Connell, 1992). Although the Khepera simulator used in the experimental work modelled noise to some extent, it would also often produce this kind of problem. Modelling noise in simulation is important to avoid this problem and can even be beneficial to the robot training (Meeden et al., 1993).

Furthermore, modelling realistic physics with some accuracy is extremely difficult (Webb, 1994). This means that if the control strategy is strongly dependent on the physics of the robot-environment interaction then the use of a real robot is probably a more practical solution.

A typical example of the dangers of robot simulation in producing incorrect control solutions is the temptation of using reinforcement, or even behaviour, dependent on information that is normally not accessible to the real robot — in particular, the simulation of sensory data that cannot be obtained with robotic technology. In the work reported in this dissertation special care was taken to avoid falling into such traps. All the data available to the robot controller is based on sensory data available to real Kheperas apart from detection of battery level and rough movement detection not available in the robot simulator. These and other sensory information not ready available to the robot were found necessary in order to add some complexity to the agent's emotional system. Nevertheless, all this sensory information can be easily acquired by a real robot.

Nowadays, a few robot simulators, like for example the Khepera simulator used, are being made available avoiding the need for each researcher to develop their own. Apart from saving effort, these also allow the use of the same platform by different researchers making their results more easily comparable. As real robots' software programming tools become increasingly sophisticated, it would be worthwhile for the manufacturers to consider the inclusion of a simulator of the robot as well.

6.3 Emotions in Autonomous Robots

Emotions play an important motivational role in natural systems, directing their attention to what is relevant for their survival. They do so in multiple ways by influencing different basic cognition mechanisms such as attention, memory, learning and reasoning.

In autonomous robots, emotions can play a similar role, filling in the lack in motivational mechanisms of traditional architectures. In the current research, this approach was taken within a reinforcement-learning architecture. An emotional system was used as a unified construct to solve separate problems that implicitly demanded some sort of attention mechanism. The emotional system is particularly appropriate for this purpose because it attributes relevance to the different experiences of the agent in the context of its internal motivations.

The existence of an explicit global appraisal system proved helpful in providing an integrated solution for different mechanisms such as reinforcement, behaviour interruption, modulation of the learning rate and the tradeoff between exploration and exploitation through the variation of the selection temperature.

The different mechanisms were tested under two different control strategies, one based on actions and one based on behaviours.

Emotions were found more useful in systems with behaviour level complexity, in particular the use of emotion-dependent reinforcement in an action-based controller was found inadequate although it was adequate for behaviour-based control.

In the experiments with the behaviour-based controller, behaviour interruption was used as an alternative to the modulation of learning parameters used in the action-based controller. In fact, these are two alternative ways to influence the learning process that serve common goals. On the one hand, the interruption of behaviour at particular points is actually determining when the agent should learn, which is equivalent to setting the learning rate to a non-zero value only at these points. On the other hand, the frequency of behaviour interruption directly influences the number of different behaviours the agent tries out, which also happens if the selection temperature is raised.

Furthermore, the developed emotion model also influences the robot's perception. This was devised as one more mechanism of attention that would make salient the features of the environment that were related to the current emotional state. Unfortunately, this mechanism was not demonstrably useful for the controllers here, which performed as well with or without this mechanism.

The robot emotions themselves were very simplified and not very realistic when compared with human-like emotions. It was considered more important to have emotions fit for purpose.

6.4 Future Work

The work reported in this dissertation focuses mostly on the different roles emotions can have in autonomous robots. For this reason, the learning architecture used leaves much room for improvement in terms of autonomy. In fact, this architecture has not even been particularly tuned for performance. The use of evolutionary techniques would probably be helpful for the refining of the architecture by selecting the design options and parameters most fit for the robot adaptation.

However, following the same line of reasoning as before, the suggestions that are given below for future research are also directed towards the strengthening of the interaction between emotions and control.

6.4.1 Emotions and their influence on control

First of all, the emotion model itself could be improved. Some deficiencies have already been pointed out during its presentation that could be corrected. For example, the emotion dependency on feelings is rather simplified. The model can certainly be extended to take into consideration temporal relations and more complex functional dependencies.

Another simplification of the current work was to associate each emotion with a single problem when in reality emotions are much more multi-coloured and complex, suggesting that they should be associated with groups of related problems instead.

Furthermore, from all the different roles associated with emotions that were discussed in Chapter 3 only a few were explored and even those with only limited success.

One of the solutions that was particularly poor in this work was the influence of emotion on perception. In fact, the learning architecture used was quite limiting. The only form of attention possible in perception was changing the perceptual values themselves, and those changes were actually compensated for by the architecture's neural networks. To properly solve this problem the learning architecture itself would have to be changed to another one which was equipped or liable to be equipped with mechanisms of attention at the level of perception. An example of such an architecture is a case-based architecture (Kolodner, 1993) where variable weights could be given to the different perceptions in the selection of the most similar case.

One of the most interesting roles of emotions that was not explored was their influence on memory. This is an extension that could be easily made to the system by associating each emotion with different memory mechanisms, more specifically with different neural networks to compute the utility values of each action. This way the robot could be made to only remember the experiences that were associated with emotional states similar to the current one. This would probably be an advantage, because it would produce a categorisation of the memorised events according to the type of problem.

In fact the recall of only the directly relevant facts is one of the benefits provided by emotions to reasoning. For example, in the solution provided by Ventura et al. (1998) emotions are actually considered an alternative method of classification that provide extra efficiency by selecting only the relevant cases at each decision point. Another alternative for memory dependent reasoning proposed by El-Nasr et al. (1998) used mood dependent recall with non-deterministic Q-learning. In this solution, when the agent is choosing an action, it gives more weight to positive outcomes if it is in a happy state, and conversely more weight to negative outcomes if it is in a sad state. Unfortunately, this algorithm required a table to save the reinforcement-learning utility values and an implementation of the algorithm with neural networks is not straightforward.

Emotions are also usually associated with action tendencies which can be important for making fast decisions. Different emotions endow the agent with different domains of actions, specifying which actions or behaviour are more appropriate for the different emotional states. For example, the emotion of fear is usually associated with fleeing or freezing and anger with fighting. This was one of the aspects of emotions that was totally overlooked by the current work where the emphasis was given to the freedom of action choice. Nevertheless, there are clear advantages to equipping the robot with a set of fast responses for emergency situations.

Another issue associated with fast responses is the physiological arousal of the body by emotions. This is a clear advantage to biological systems, but its transposition to artificial systems is not very straightforward. However, a simple solution can consist of having the robot's motor response dependent on the emotional state.

Finally all the emotions' influence on social interaction, which was omitted in the current work, can be a great source of future development.

6.4.2 Emotions development

A very important aspect of emotions that has been left out for the moment is the development of more sophisticated emotions. Humans have innate emotions that are experienced early in life (Primary emotions) and emotions that are built on the previous (Secondary emotions) by pairing experiences with emotional responses (Damásio, 1994).

The emotions model that was implemented did not support the development of the emotions through learning during the agent lifetime. The agent depends only on its primary emotions for survival. However, the system could be extended to contemplate the emergence of secondary emotions. These could for instance result from the associations between stimuli, or feelings in the case of the current model, and existing emotions. The development of new and more complex emotions on the top of the primary ones is a more difficult issue and a subject of research. This can be based on the exploration of temporal relations between the existing emotion (*e.g.*, relief) and consist of the categorisation of recurrent patterns of emotion activation. In the case of social agents, the new emotions can also consist of the categorisation of recurrent emotional experiences associated with certain individuals (*e.g.*, love and hate) or characteristic experiences of social interaction (*e.g.*, jealousy).

In the current model, the initial hand-designed associations between feelings and basic emotions constitute the robot's initial frame of reference. These associations should be maintained by the robot and new ones should be developed on top based on the robot's experiences. This added feature can provide an element of change in the value system that will, hopefully, increase its autonomy.

The fact that emotions are used as reinforcement means that if the emotion system has the ability to develop during the agent lifetime then the agent will have a dynamical and incremental value system that is also learned by the agent. In the adaptation of an agent to its environment, value systems, although needed and useful, will always work as a limitation on what the robot can learn. If the value system is very broad, the learning task will be very slow, very difficult or even impossible. On the other hand, if the value system is very specific, the learning will be very limited. A solution to this problem that might help in scaling-up learning architectures is to have a dynamical and incremental value system that is also learned by the system. Several researchers (Verschure et al., 1995; Cariani, 1992a) have suggested in the past the on-line development of the value system for higher adaptation to the environment.

For the introduction of new emotions, the triggering mechanism as it is might be inadequate. The introduction of an emotion of surprise might be necessary to allow the agent to take notice and learn about features of its environment that have not been caught by its emotional system, because they have never been experienced before.

It is fundamental for the correct functioning of the control system that this is triggered whenever something that might be relevant happens. For example, it was crucial for the correct execution of its task that the presence of lights, its food source, influenced the robot's emotional state, which they do through the feeling of warmth. For the same reasons, it is important that the controller can also be triggered by a sense of novelty.

Surprise has been proposed previously to drive learning (*e.g.*, Moffat and Frijda, 1995).

Similarly, the detection of a failure in predicting the environment has been used to drive learning but under the denomination of curiosity (Schmidhuber, 1991). Together with interest, excitement and boredom these are probably the most used emotions in robot applications where emotions influence learning.

In the long run, the triggering mechanism will end up benefiting from the introduction of new emotions in a straightforward way. As the new emotion associations are created they will influence the robot's emotional state which in turn will result in the detection by the triggering mechanisms of events related to the new associations.

This will not only happen with the control triggering mechanism but with all the other mechanisms that are based on emotions — one of the advantages of having a unified solution.

6.5 Conclusions

The experimental test of the developed emotional mechanisms against more traditional approaches to the realization of different reinforcement-learning problematic components demonstrated that emotions were a competent alternative. In the specific case of the detection of state transition, emotions were actually a more successful alternative in terms of the agent's learning performance.

The use of emotions as an abstraction has the advantage of allowing different components of reinforcement learning to be brought together under the same construct. This was found helpful for two reasons:

- the synchrony and coherence between the different components achieved by this unified solution represented a slight enhancement of the agent's performance;
- the design of the different components was simplified to the design of a single construct, the robot's emotions;

Furthermore, the use of emotions provided a new perspective over these different task-dependent components of a reinforcement-learning framework. This resulted in the introduction of innovative mechanisms that were tested in the robot experiments. The most important innovations being in terms of the reinforcement function and the specification of state transition:

- a multi-dimensional reinforcement function that takes into consideration the different problems faced by the robot with variable degrees of attention dependent on the robot's current priorities;
- a simplified definition of state transition based on detection of significant events captured by variation in the reinforcement function value.

The emotional system selects between different reinforcement functions according to the context of the world, *i.e.* it might choose to ignore other problems that exist when faced with a more important one. For example, the reinforcement function might not punish the robot for its collision with an obstacle and instead reinforce it for successfully extracting energy from a light. The attribution by the reinforcement function of variable degrees of attention to each of the different problems might be taken to be a source of confusion for the learning process. However, experiments showed that instead of confusing the learning process, this was actually advantageous and that the learning algorithm was exploiting the non-linearities of the reinforcement function.

The presented event detection mechanism also profits from the novel structure of the reinforcement function. Apart from providing an absolute reinforcement value that varies with the robot's situation, the developed reinforcement function based on emotion also differentiates and prioritises the different problems faced by the robot. This added information allows the detection of events when there is a difference in type of dominant problem and not just in problem degree.

Emotions have a vital motivational role in natural cognition. They have the power to drive and influence a great variety of basic cognition mechanisms. As such they served as inspiration for the current research in the development of innovative mechanisms in an artificial robot, but many of the important features of emotions were left unexplored and can be considered for possible extensions of this research. The field of emotional agents has promising research directions and should be regarded as an important source of inspiration to meet some of the serious deficiencies present in today's artificial systems.

Bibliography

Ackley, D. H. and Littman, M. L. (1990). Generalization and scaling in reinforcement learning. In Touretzky, D. S., editor, *Advances in Neural Information Processing Systems*, pages 550–557. CA. Morgan Kaufmann.

Agre, P. E. and Chapman, D. (1991). What are plans for? In Maes, P., editor, *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, pages 17–34. The MIT Press. Special Issue of Robotics and Autonomous Systems. First published in Robotics and Autonomous Systems 6 (1990).

Albus, J. S. (1990). The role of world modeling and value judgment in perception. In Meystel, A., Herath, J., and Gray, S., editors, *Proceedings of the* 5^{th} *IEEE International Symposium on Intelligent Control*, Los Alamitos, California. IEEE Computer Society Press.

Almássy, N. and Verschure, P. (1992). Optimizing self-organizing control architectures with genetic algorithms: The interaction between natural selection and ontogenesis. In *Proceedings of the Second Conference on Parallel Problem Solving from Nature*, Brussels. Elsevier. Also available as Technical Report 92.10, Institute for informatics, AI Laboratory University Zürich-Inchel.

Araujo, A. F. R. (1994). *Memory, emotions & neural networks.* PhD thesis, Sussex University.

Arcos, J.-L., Cañamero, D., and de Mántaras, R. L. (1998). Affect-driven generation of expressive musical performances. In *AAAI Fall Symposium on Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 1–6. AAAI Press.

Asada, M. (1996). An agent and an environment: A view on body scheme. In Tani, J. and Asada, M., editors, *Proceedings of the 1996 IROS Workshop on Towards real autonomy.*

Aubé, M. (1998a). A commitment theory of emotions. In *AAAI Fall Symposium on Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 13–18. AAAI Press.

Aubé, M. (1998b). Designing adaptive cooperating animats will require designing emotions: Expanding upon Toda's urge theory. In *SAB'98 Workshop on Grounding Emotions in Adaptive Systems.*

Baldwin, J. M. (1896). A new factor. *The American Naturalist*, 30:441–451.

Balkenius, C. (1995). *Natural intelligence in artificial creatures.* Lund University Cognitive Studies 37.

Ballard, D. H. (1991). Animate vision. *Artificial Intelligence*, 48:57–86.

Bates, J. (1994). The role of emotions in believable agents. Technical Report CMU-CS-94-136, Carnegie Mellon University, School of Computer Science.

Bates, J., Loyall, A. B., and Reilly, W. S. (1992a). An architecture for action, emotion, and social behavior. In *Artificial social systems: Fourth European workshop on modeling autonomous agents in a multi-agent world*. Elsevier. Also available as internal report CMU-CS-92-144, Carnegie Mellon University.

Bates, J., Loyall, A. B., and Reilly, W. S. (1992b). Integrating reactivity, goals, and emotion in a broad agent. CMU-CS 92-142, Carnegie Mellon University, School of Computer Science.

Beaudoin, L. and Sloman, A. (1993). A study of motive processing and attention. In A.Sloman, D.Hogg, G.Humphreys, Ramsay, A., and Partridge, D., editors, *Proceedings of AISB93*, pages 229–238. IOS Press, Oxford.

Bechara, A., Damasio, H., Tranel, D., and Damásio, A. R. (1997). Deciding advantageously before knowing the advantageous strategy. *Science*, 275:1293–1295.

Beck, R. C. (1983). *Motivation: Theories and principles.* Prentice-Hall, Inc., second edition.

Beckers, R., Holland, O. E., and Deneubourg, J. L. (1994). From local actions to global tasks: Stimergy and collective robotics. In *Proceedings of the International Conference Artificial Life IV*, pages 181–189. The MIT Press.

Beer, R. D. (1990). *Intelligence as adaptive behaviour — An experiment in computational Neuroethology.* Academic Press, Inc.

Blaney, P. (1986). Affect and memory. *Psychological bulletin*, 99(2):229–246.

Blidberg, D. R. (1989). Autonomous underwater vehicles: Current activities and research opportunities. In Kanade, T., Groen, F. C. A., and Hertzberger, L. O., editors, *Intelligent Autonomous Systems 2*.

Blumberg, B. (1994). Action-selection in hamsterdam: Lessons from ethology. In *From animals to animats 3 — Proceedings of the Third International Conference on Simulation of Adaptive Behavior*. The MIT Press.

Blumberg, B. (1995). Multi-level direction of autonomous creatures for real-time virtual environments. In *Computer Graphics Proceedings, Siggraph'95*, Los Angeles, California.

Boden, M. A. (1993). Autonomy and artificiality. CSRP 307, University of Sussex.

Boer, B. G. (1994). An autonomous robot learning basic behaviours. Technical report, Leiden University, The Netherlands.

Bolles, R. C. (1980). Some functionalistic thoughts about regulation. In Toates, F. M. and Halliday, T. R., editors, *Analysis of motivational processes.* Academic Press.

Botelho, L. M. and Coelho, H. (1997). Emotion-based attention shift in autonomous agents. In *Intelligent agents III — Agent theories, architectures and languages. ECAI'96 Workshop proceedings.* Springer-Verlag.

Bourgine, P. and Varela, F. J. (1992). Introduction. In Varela, F. J. and Bourgine, P., editors, *Toward a practice of autonomous systems — Proceedings of the First European Conference on Artificial Life*, Cambridge, Mass. London. The MIT Press.

Bower, G. H. (1981). Mood and memory. *American Psychologist*, 36(2):129–148.

Bower, G. H. and Cohen, P. R. (1982). Emotional influences in memory and thinking: Data and theory. In *Affect and cognition — The seventeenth annual Carnegie symposium on cognition.* Lawrence Erlbaum Associates.

Bower, G. H. and Mayer, J. D. (1985). Failure to replicate mood-dependent retrieval. *The Bulletin of the Psychonomic Society*, 23(1):39–42.

Bozinovski, S. (1982). A self-learning system using secondary reinforcement. In Trappl, R., editor, *Cybernetics and Systems Research*, pages 397–402. North-Holland Publishing Company.

Braitenberg, V. (1984). *Vehicles. Experiments In Synthetic Psychology.* The MIT Press.

Breazeal, C. (1998). Early experiments using motivations to regulate human-robot interaction. In *AAAI Fall Symposium on Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 31–36. AAAI Press.

Brooks, R. A. (1986a). Achieving artificial intelligence through building robots. A.I. Memo 899, MIT.

Brooks, R. A. (1986b). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23. Also MIT AI Memo 864, September 1985.

Brooks, R. A. (1989). A robot that walks: emergent behaviours from a carefully evolved network. A.I. Memo 1091, MIT.

Brooks, R. A. (1991). Intelligence without reason. In *Proceedings of 12th Int. Joint Conf. on Artificial Intelligence.* Also MIT A.I. Memo 1293, April 1991.

Brustoloni, J. C. (1991). Autonomous agents: characterization and requirements. CMU-CS 91-204, Carnegie-Mellon University.

Cañamero, D. (1997). Modeling motivations and emotions as a basis for intelligent behavior. In *Proceedings of the First International Symposium on Autonomous Agents, AA'97.* The ACM Press.

Cañamero, D. (1998). Issues in the design of emotional agents. In *AAAI Fall Symposium on Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 49–54. AAAI Press.

Cariani, P. (1992a). Emergence and artificial life. In Langton, C. G., Taylor, C., Farmer, J. D., and Rasmussen, S., editors, *Artificial life II*. Addison-Wedley Publishing Company.

Cariani, P. (1992b). Some epistemological implications of devices which construct their own sensors and effectors. In Varela, F. J. and Bourgine, P., editors, *Toward a practice of autonomous systems — Proceedings of the First European Conference on Artificial Life*. The MIT Press.

Chapman, D. and Kaelbling, L. P. (1990). Learning from delayed reinforcement in a complex domain. Technical Report TR-90-11, Teleos Research.

Chapman, D. and Kaelbling, L. P. (1991). Input generalization in delayed reinforcement learning: An algorithm and performance comparisons. In *IJCAI'91 — Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*.

Charland, L. C. (1995). Emotion as a natural kind: towards a computational foundation for emotion theory. *Philosophical psychology*, 8(1):59–84.

Cliff, D., Harvey, I., and Husbands, P. (1992). Incremental evolution of neural network architectures for adaptive behaviour. CSRP 256, University of Sussex.

Correia, L. (1995). *Veículos autónomos baseados em comportamentos — um modelo de controlo de decisão*. PhD thesis, Universidade Nova de Lisboa, Faculdade de Ciências e Tecnologia.

Correia, L. and Steiger-Garção, A. (1995). A useful autonomous vehicle with a hierarchical behavior control. In Morán, F., Moreno, A., Merelo, J., and Chacón, P., editors, *Advances in artificial life — Proceedings of the Third European Conference on Artificial Life*. Springer-Verlag.

Covrigaru, A. A. and Lindsay, R. K. (1991). Deterministic autonomous systems. *AI Magazine*, 12(3):110–117.

Cytowic, R. E. (1993). *The man who tasted shapes*. Abacus.

Damásio, A. R. (1994). *Descartes' error — Emotion, reason and human brain*. Picador.

Damásio, A. R. (1995). The selfless consciousness. *Behavioral and Brain Sciences*, 18(1):130–131.

Darwin, C. (1965). *The expression of the emotions in man and animals*. The University of Chicago Press.

Dawkins, R. (1976). *The Selfish Gene*. Oxford University Press, Oxford.

De Sousa, R. (1987). *The rationality of emotion*. The MIT Press.

Demiris, J., Rougeaux, S., Hayes, G. M., Berthouze, L., and Kuniyoshi, Y. (1997). 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.

Digney, B. L. (1998). Learning hierarchical control structures for multiple tasks and changing environments. In *From animals to animats 5 — Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*, pages 321–330. The MIT Press.

Donnett, J. G. (1992). *Analysis and synthesis in the design of locomotor and spatial competences for a multisensory mobile robot.* PhD thesis, University of Edinburgh.

Dorigo, M. (1995). ALECSYS and the AutonoMouse: Learning to control a real robot by distributed classifier systems. *Machine Learning*, 19:209–240.

Dorigo, M. and Colombetti, M. (1993). Robot shaping: Developing situated agents through learning. Technical Report 40, International Computer Science Institute, Berkeley.

Dreyfus, H. L. (1992). *What computers still can't do: A critique of artificial reason.* The MIT Press.

Dyer, M. G. (1987). Emotions and their computations: Three computer models. *Cognition and emotion*, 1(3):323–347.

Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6(3/4):169–200.

El-Nasr, M. S., Ioerger, T. R., and Yen, J. (1998). Learning and emotional intelligence in agents. In *AAAI Fall Symposium on Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 150–155. AAAI Press.

Ferguson, I. A. (1992). *Touring Machines: An architecture for dynamic, rational, mobile agents.* PhD thesis, University of Cambridge.

Floreano, D. and Mondada, F. (1996). Evolution of plastic neurocontrollers for situated agents. In Maes, P., Mataric, M. J., Meyer, J.-A., Pollack, J., and Wilson, S. W., editors, *From animals to animats 4 — Proceedings of the fourth International Conference on Simulation of Adaptive Behavior*, pages 402–410. The MIT Press.

Floreano, D. and Urzelai, J. (1998). Evolution and learning in autonomous robotic agents. In Mange, D. and Tomassini, M., editors, *Bio-Inspired Computing Systems.* PPUR, Lausanne.

Foliot, G. and Michel, O. (1998). Learning object significance with an emotion based process. In *SAB'98 Workshop on Grounding Emotions in Adaptive Systems.*

Frijda, N. H. and Swagerman, J. (1987). Can computers feel? theory and design of an emotional system. *Cognition and Emotion*, 1(3):235–257.

Gallistel, C. R., Brown, A. L., Carey, S., Gelman, R., and Keil, F. C. (1991). Lessons from animal learning for the study of cognitive development. In Carey, S. and Gelman, R., editors, *The Epigenesis of Mind: Essays on biology and Cognition.* L. Erlbaum Associates.

Gazzaniga, M. S. and LeDoux, J. E. (1978). *The integrated mind.* Plenum Press.

Giralt, G., Alami, R., and Chatila, R. (1989). Autonomy versus teleoperation for intervention robots? a case for task level teleprogramming. In Kanade, T., Groen, F. C. A., and Hertzberger, L. O., editors, *Intelligent Autonomous Systems 2*.

Goleman, D. (1995). *Emotional Intelligence*. Bloomsbury Publishing Plc.

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

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

Grossberg, S. (1971). On the dynamics of operant conditioning. *J. Theor. Biol.*, 33:225–255.

Grossberg, S. and Gutowski, W. (1987). Neural dynamics of decision making under risk: Affective balance and cognitive-emotional interactions. *Psychological Review*, 94(3):300–318.

Hallam, B. and Hayes, G. (1992). Comparing robot and animal behaviour. DAI Research Paper 598, University of Edinburgh.

Hallam, J. C. T. and Malcolm, C. A. (1993). Behaviour: perception, action and intelligence - the view from situated robotics. *Phil. Trans. R. Soc. Lond.*, 11.

Harnad, S. (1989). Minds, machines, and searle. *Journal of Experimental & Theoretical Artificial Intelligence*, 1:5–25.

Haugeland, J. (1985). *Artificial Intelligence: The very idea*. The MIT Press.

Hertz, J., Krogh, A., and Palmer, R. G. (1991). *Introduction to the theory of neural computation*. Addison-Wesley Publishing Company.

Hexmoor, H. H., Lammens, J. M., and Shapiro, S. C. (1993). An autonomous agent architecture for integrating "unconscious" and "conscious", reasoned behaviors. CS 93-37, State U of New York Buffalo.

Horswill, I. (1993). Polly: A vision-based artificial agent. In *AAAI*, Washington DC.

Izard, C. E. (1993). Four systems for emotion activation: Cognitive and noncognitive processes. *Psychological Review*, 100(1):68–90.

James, W. (1890). *Principles of psychology*. Holt, New York.

Jamon, M. (1991). The contribution of quantitative models to the long distance orientation problems. In *From animals to animats — Proceedings of the First Conference on Simulation of Adaptive Behavior*, pages 160–168. The MIT Press.

Kaelbling, L. P. (1990). Learning in embedded systems. Ph.D. Thesis STAN-CS-90-1326, Department of Computer Science, Stanford University.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Kaiser, S. and Wehrle, T. (1996). Situated emotional problem solving in interactive computergames. In *Proceedings of the* 8^{th} *Conference of the International Society for Research on Emotions, ISRE'96*, pages 276–280. ISRE Publications.

Kirchhoff, U. (1989). Space robotics and the increase of system autonomy. In Kanade, T., Groen, F. C. A., and Hertzberger, L. O., editors, *Intelligent Autonomous Systems 2*.

Kitano, H. (1995). A model for hormonal modulation of learning. In *IJCAI-95 — Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, volume 1, pages 532–538.

Klein, J. T. (1996). Computer response to frustration. Msc in Media Arts and Sciences, The MIT School of Architecture and Planning.

Kolodner, J. (1993). *Case-based reasoning*. Morgan Kaufmann Publishers.

Kravitz, E. A. (1988). Hormonal control of behavior: Amines and the biasing of behavioral output in lobsters. *Science*, 241:1775–1781.

Kröse, B. and Eecen, M. (1994). A self-organizing representation of sensor space for mobile robot navigation. In *IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, pages 9–14.

Kuniyoshi, Y., Inaba, M., and Inoue, H. (1994). Learning by watching: Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10(6):799–822.

Lammens, J. M., Hexmoor, H. H., and Shapiro, S. C. (1993). Of elephants and men. In *Biology and Technology of Intelligent Autonomous Agents*, Trento, Italy. NATO-ASI.

Langton, C. G. (1992). Preface. In Langton, C. G., Taylor, C., Farmer, J. D., and Rasmussen, S., editors, *Artificial life II*. Addison-Wedley Publishing Company.

Lazarus, R. S. (1982). Thoughts on the relations between emotion and cognition. *American Psychologist*, 37:1019–1024.

LeDoux, J. E. (1997). Emotion, memory and the brain. *Scientific American*. Special Issue:*Mysteries of the Mind*.

LeDoux, J. E. (1998). *The Emotional Brain*. Phoenix.

Liaw, M. A. A. J.-S. (1995). Sensorimotor transformations in the worlds of frogs and robots. *Artificial Intelligence*, 72:53–79.

Lin, L.-J. (1991). Programming robots using reinforcement learning and teaching. In *AAAI-91: proceedings of the ninth National Conference on Artificial Intelligence*, pages 781–786. AAAI Press/The MIT Press.

Lin, L.-J. (1992). Self-improving reactive agents based on reinforcement learning planning and teaching. *Machine Learning*, 8:293–321.

Lin, L.-J. (1993). *Reinforcement learning for robots using neural networks*. PhD thesis, Carnegie Mellon University. Technical report CMU-CS-93-103.

Loewenstein, G. (1996). Out of control - visceral influences on behavior. *Organizational Behavior and Human Decision Processes*, 65(3):272–292.

Luck, M. and d'Inverno, M. (1995). Agency and autonomy: a formal framework. CS-RR 276, University of Warwick.

Maes, P. (1989). The dynamics of action selection. In *IJCAI-89 — Proceedings of the Eleventh International Joint Conference on Artificial Intelligence.*

Maes, P. (1991a). A bottom-up mechanism for behavior selection in an artificial creature. In *From animals to animats — Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pages 238–246. The MIT Press.

Maes, P. (1991b). Guest editorial. In Maes, P., editor, *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, pages 1–2. The MIT Press. Special Issue of Robotics and Autonomous Systems.

Maes, P. (1992). Learning behavior networks from experience. In Varela, F. J. and Bourgine, P., editors, *Toward a practice of autonomous systems — Proceedings of the First European Conference on Artificial Life.* The MIT Press.

Maes, P. and Brooks, R. A. (1990). Learning to coordinate behaviors. In *AAAI*, pages 796–802.

Mahadevan, S. and Connell, J. (1992). Automatic programming of behavior-based robots using reinforcement learning. *Artificial intelligence*, 55:311–365.

Malcolm, C. and Smithers, T. (1989). Symbol grounding via a hybrid architecture in an autonomous assembly system. DAI research paper 420, University of Edinburgh.

Mataric, M. J. (1994). Reward functions for accelerated learning. In *Proc. of Conference on Machine Learning - 1994*, pages 181–189.

Mataric, M. J. (1995). Designing and understanding adaptive group behavior. *Adaptive Behavior*, 4(1):51–80.

Mataric, M. J. and Cliff, D. (1996). Challenges in evolving controllers for physical robots. *"Evolutional Robotics", special issue of Robotics and Autonomous Systems*, 19(1):67–83. Also Brandeis University Computer Science Technical Report CS-95-184, November 1995.

Maturana, H. R. (1969). The biology of cognition. In Cohen, R. S. and Wartofsky, M. W., editors, *Autopoiesis and cognition: The realization of the living.* D. Reidel Publishing Company, Dordrecht : Holland / Boston : U.S.A./ London : England.

Maturana, H. R. and Varela, F. J. (1973). Autopoiesis: The organization of the living. In Cohen, R. S. and Wartofsky, M. W., editors, *Autopoiesis and cognition: The realization of the living.* D. Reidel Publishing Company, Dordrecht : Holland / Boston : U.S.A./ London : England.

Maturana, H. R. and Varela, F. J. (1980). *Autopoiesis and cognition: The realization of the living.* D. Reidel Publishing Company, Dordrecht : Holland / Boston : U.S.A./ London : England.

Maturana, H. R. and Varela, F. J. (1987). *The tree of knowledge — The biological roots of human understanding.* Boston: Shambhala.

McCallum, A. K. (1996). Learning to use selective attention and short-term memory in sequential tasks. In Maes, P., Mataric, M. J., Meyer, J.-A., Pollack, J., and Wilson, S. W., editors, *From animals to animats 4 — Proceedings of the fourth International Conference on Simulation of Adaptive Behavior*. The MIT Press.

McCauley, L. and Franklin, S. (1998). An architecture for emotion. In *AAAI Fall Symposium on Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 122–127. AAAI Press.

McFarland, D. (1992). Autonomy and self-sufficiency in robots. AI-MEMO 92-03, VUB.

McFarland, D. (1993). *Animal Behaviour*. Longman Scientific & Technical, second edition.

McFarland, D. (1994). Animal robotics — from self-sufficiency to autonomy. In Gaussier, P. and Nicoud, J.-D., editors, *Proceedings of "From Perception to Action" conference*, pages 47–54.

McFarland, D. and Spier, E. (1997). Basic cycles, utility and opportunism in self-sufficient robots. *Robotics and Autonomous Systems*, 20:179–190.

Meeden, L., McGraw, G., and Blank, D. (1993). Emergent control and planning in an autonomous vehicle. In *Proceedings of the* 15*th* *Annual Cognitive Science Society Conference*.

Melhuish, C., Holland, O., and Hoddell, S. (1998). Collective sorting and segregation in robots with minimal sensing. In *From animals to animats 5 — Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*, pages 465–470. The MIT Press.

Melzack, R. (1997). Phantom limbs. *Scientific American*. Special Issue:*Mysteries of the Mind*.

Michaud, F., Lachiver, G., and Dinh, C. T. L. (1996). A new control architecture combining reactivity, planning, deliberation and motivation for a situated autonomous agent. In Maes, P., Mataric, M. J., Meyer, J.-A., Pollack, J., and Wilson, S. W., editors, *From animals to animats 4 - Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 245–254. The MIT Press.

Michel, O. (1996). *Khepera Simulator* package version 2.0: Freeware mobile robot simulator written at the University of Nice Sophia–Antipolis. Downloadable from the World Wide Web at `http://wwwi3s.unice.fr/˜om/khep-sim.html`.

Minsky, M. (1986). *The Society of Mind*. Simon and Schuster, New York.

Moffat, D., Frijda, N., and Phaf., R. (1993). Analysis of a model of emotions. In et al., A. S., editor, *Prospects for Artificial Intelligence*, pages 219–228. IOS Press.

Moffat, D. and Frijda, N. H. (1995). Where there is a *will* there's an agent. In *Intelligent agents: ECAI-94 Workshop on Agent Theories Architectures, and languages*. Springer-Verlag.

Mondada, F., Franzi, E., and Ienne, P. (1994). Mobile robot miniaturization: A tool for investigation in control algorithms. In Yoshikawa, T. and Miyazaki, F., editors, *Experimental Robotics III*, Lecture notes in Control and Information Sciences. Springer-Verlag, London.

Morignot, P. and Hayes-Roth, B. (1995). Why does an agent act? In *AAAI Spring Symposium on Representing Mental States and Mechanisms.* Also available as report KSL 94-76 of Stanford University.

Mowrer, O. H. (1960). *Learning theory and behavior.* John Wiley & Sons, Inc., New York.

Nehmzow, U. (1994). Autonomous acquisition of sensor-motor couplings in robots. Technical report UMCS-94-11-1, University of Manchester.

Nolfi, S. and Parisi, D. (1996). Learning to adapt to changing environments in evolving neural networks. *Adaptive Behavior*, 5:75–98.

Oatley, K. (1987). Editorial: Cognitive science and the understanding of emotions. *Cognition and Emotion*, 1(3):209–216.

Ortony, A., Clore, G. L., and Collins, A. (1988). *The cognitive structure of emotions.* Cambridge University Press.

Panksepp, J. (1982). Toward a general psychobiological theory of emotions. *The behavioural and brain sciences*, 5(3):407–422.

Panksepp, J. (1995). The emotional brain and biological psychiatry. *Advances in Biological Psychiatry*, 1:263–288.

Papez, J. W. (1937). A proposed mechanism of emotion. *Archives of Neurology and Psychiatry*, 38:725–743.

Patel, M. J. and Schnepf, U. (1992). Concept formation as emergent phenomena. In Varela, F. J. and Bourgine, P., editors, *Toward a practice of autonomous systems — Proceedings of the First European Conference on Artificial Life*, pages 11–20. The MIT Press.

Pavlov, I. P. (1927). *Conditioned reflexes.* Oxford University Press, London.

Pebody, M. (1995). Learning and adaptivity: Enhancing reactive behavior architectures in real-world interactions systems. In Morán, F., Moreno, A., Merelo, J., and Chacón, P., editors, *Advances in artificial life — Proceedings of the Third European Conference on Artificial Life.* Springer-Verlag.

Pfeifer, R. (1982). Cognition and emotion: An information process approach. CIP working paper 436, Department of Psychology, Carnegie-Mellon University.

Pfeifer, R. (1994). The "Fungus Eater approach" to emotion: A view from artificial intelligence. *Cognitive Studies: Bulletin of the Japanese Cognitive Science Society*, 1(2):42–57. English version: Technical report IFI-AI-95, Artificial Intelligence Laboratory, University of Zurich.

Pfeifer, R. and Nicholas, D. W. (1985). Toward computational models of emotion. In Steels, L. and Campbell, J. A., editors, *Progress in Artificial Intelligence*, pages 184–192. Ellis Horwood, Chichester, U.K.

Picard, R. (1997). *Affective Computing.* The MIT Press.

Picard, R. W. (1995). Affective computing. Technical Report 321, MIT Media Laboratory.

Plutchick, R. (1984). Emotions: A general psychoevolutionary theory. In Scherer, K. R. and Ekman, P., editors, *Approaches to Emotion.* Lawrence Erlbaum, London.

Power, M. and Dalgleish, T. (1997). *Cognition and Emotion.* Psychology Press.

Prem, E. (1995). Grounding and the entailment structure in robots and artificial life. In Morán, F., Moreno, A., Merelo, J., and Chacón, P., editors, *Advances in artificial life — Proceedings of the Third European Conference on Artificial Life.* Springer-Verlag.

Ram, A., Arkin, R., Boone, G., and Pearce, M. (1994). Using genetic algorithms to learn reactive control parameters for autonomous robotic navigation. *Adaptive Behavior*, 2(3):277–304.

Reeke, Jr., G. N. (1996). Responding to the unexpected: Neural darwinism as a basis for autonomy in biological and robotic systems. In Tani, J. and Asada, M., editors, *Proceedings of the 1996 IROS Workshop on Towards Real Autonomy.*

Reeke, Jr., G. N. and Sporns, O. (1993). Behaviorally based modeling and computational approaches to neuroscience. *Annual Review Neuroscience*, 16:597–623.

Riecken, D. (1998). Wolfgang: “emotions” and architecture which enable learning to compose music. In *SAB’98 Workshop on Grounding Emotions in Adaptive Systems.*

Rodriguez, M. and Muller, J.-P. (1995). Towards autonomous cognitive animats. In Morán, F., Moreno, A., Merelo, J., and Chacón, P., editors, *Advances in artificial life — Proceedings of the Third European Conference on Artificial Life.* Springer-Verlag.

Russell, S. J. and Norvig, P. (1995). *Artificial Intelligence — A Modern Approach.* Prentice Hall, Inc.

Rutkowska, J. C. (1995). Reassessing piaget’s theory of sensorimotor intelligence: a view from cognitive science. Cognitive Science Research Paper 369, School of Cognitive and Computing Sciences at the University of Sussex. In J. G. Bremner’s “Infant Development: Recent Advances”.

Schachter, S. (1964). The interaction of cognitive and physiological determinants of emotional state. *Advances in Experimental Social Psychology*, 1:49–80.

Scheier, C. and Pfeifer, R. (1995). Classification as sensory-motor coordination — a case study on autonomous agents. In Morán, F., Moreno, A., Merelo, J., and Chacón, P., editors, *Advances in artificial life — Proceedings of the Third European Conference on Artificial Life*, pages 657–667. Springer-Verlag.

Scherer, K. R. (1984). On the nature and function of emotion: A component process approach. In Scherer, K. R. and Ekman, P., editors, *Approaches to Emotion*. Lawrence Erlbaum, London.

Schmidhuber, J. (1991). A possibility for implementing curiosity and boredom in model-building neural controllers. In *From animals to animats — Proceedings of the First Conference on Simulation of Adaptive Behavior*, pages 222–227. The MIT Press.

Schwartz, B. and Reisberg, D. (1991). *Learning and memory*. W.W. Norton and Company, Inc.

Shaver, P., Schwartz, J., Kirson, D., and O'Connor, C. (1987). Emotion knowledge: Further exploration of a prototype approach. *Journal of personality and social psychology*, 52(6):1061–1086.

Shibata, T., Ohkawa, K., and Tanie, K. (1996). Spontaneous behavior of robots for cooperation - emotionally intelligent robot system. In *Proceedings 1996 IEEE International Conference on Robotics and Automation*.

Simon, H. A. (1967). Motivational and emotional controls of cognition. *Psychological Review*, 74:29–39.

Sloman, A. (1987). Motives mechanisms and emotions. *Emotion and Cognition*, 1(3):217–234. Reprinted in M. A. Boden, editor, *The Philosophy of Artificial Intelligence*, "Oxford Readings in Philosophy" Series, Oxford University Press, pages 231-247, 1990.

Sloman, A. (1998). Review of: Affective computing. Review written for *AI Magazine*.

Sloman, A., Beaudoin, L., and Wright, I. (1994). Computational modeling of motive-management processes. In Frijda, N., editor, *Proceedings of the Conference of the International Society for Research in Emotions*, pages 344–348, Cambridge. ISRE Publications. Poster.

Sloman, A. and Croucher, M. (1981). Why robots will have emotions. In *IJCAI'81 — Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 2369–71. Cognitive Science Research Paper 176, Sussex University.

Smithers, T. (1992). Taking eliminative materialism seriously: A methodology for autonomous systems research. In Varela, F. J. and Bourgine, P., editors, *Toward a practice of autonomous systems — Proceedings of the First European Conference on Artificial Life*. The MIT Press.

Staller, A. and Petta, P. (1998). Towards a tractable appraisal-based architecture for situated cognizers. In *SAB'98 Workshop on Grounding Emotions in Adaptive Systems*.

Steels, L. (1994a). The artificial life roots of artificial intelligence. *Artificial Life Journal*, 1(1).

Steels, L. (1994b). Building agents with autonomous behavior systems. In Steels, L. and Brooks, R. A., editors, *The artificial life route to artificial intelligence. Building situated embodied agents.* Lawrence Erlbaum Associates, New Haven.

Steels, L. (1994c). A case study in the behaviour-oriented design of autonomous agents. In *Simulation of adaptive behaviour.*

Steels, L. (1994d). Emergent functionality in robotic agents through on-line evolution. In *Artificial Life Conference MIT*, Cambridge. The MIT Press.

Stewart, J. (1995). The implications for understanding high-level cognition of a grounding in elementary adaptive systems. *Robotics and Autonomous Systems*, 16(2-4):107–116.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning.* The MIT Press.

The Oxford English dictionary (1989). *The Oxford English dictionary.* Oxford University Press, second edition. Prepared by J. A. Simpson and E .S. C. Weiner.

Thompson, E., Palacios, A., and Varela, F. J. (1992). Ways of coloring: comparative color vision as a case study for cognitive science. *Behavioral and Brain Sciences*, 15:1–74.

Toda, M. (1982). *Man, robot and society.* Martinus Nijhoff Publishing.

Toda, M. (1993). The urge theory of emotion and cognition: Chapter 1 Emotions and urges. SCCS technical report 93-1-01, Chuyko University. English version of the book "Kanjo (Emotion)".

Toda, M. (1994). Emotion, society and the versatile architecture. SCCS technical report 94-1-02, Chuyko University.

Tomkins, S. S. (1984). Affect theory. In Scherer, K. R. and Ekman, P., editors, *Approaches to Emotion.* Lawrence Erlbaum, London.

Varela, F. J. (1979). *Principles of biological autonomy.* Elsevier North Holland, Inc, North Holland / New York / Oxford.

Varela, F. J. (1992). Whence perceptual meaning? a cartography of current ideas. In Varela, F. J. and Dupuy, J.-P., editors, *Understanding origins — Contemporary views on the origin of life, mind and society.* Kluwer Academic Publishers, Dordrecht: Holland / Boston: U.S.A. / London: England.

Varela, F. J., Thompson, E., and Rosch, E. (1991). *The embodied mind — cognitive science and human experience.* The MIT Press, Cambrige, Massachusetts; London, England.

Velásquez, J. D. (1998). A computational framework for emotion-based control. In *SAB'98 Workshop on Grounding Emotions in Adaptive Systems.*

Ventura, R., Custódio, L., and Pinto-Ferreira, C. (1998). Emotions — the missing link? In *AAAI Fall Symposium on Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 170–175. AAAI Press.

Verschure, P., Kröse, B. J. A., and Pfeifer, R. (1992). Distributed adaptive control: The self-organization of structured behavior. *Robotics and Autonomous Systems*, 9:181–196.

Verschure, P. F. M. J., Wray, J., Sporns, O., and Tononi, G. (1995). Multilevel analysis of classical conditioning in a behaving real world artifact. *Robotics and Autonomous Systems*, 16:247–265.

Walker, A., Peremans, H., and Hallam, J. (1998). One tone, two ears, three dimensions: A robotic investigation of pinnae movements used by rhinolophid and hipposiderid bats. *J. Acoust. Soc. Am.*

Watkins, C. (1989). *Learning from delayed rewards*. PhD thesis, King's College.

Webb, B. (1994). Robotic experiments in cricket phonotaxis. In Dave Cliff, Philip Husbands, J.-A. M. and Wilson, S. W., editors, *From animals to animats 3 — Proceedings of the Third International Conference on Simulation of Adaptive Behavior*. The MIT Press.

Weher, R. (1987). Matched filters - neural models of the external world. *Journal of comparative physiology A*, 161:511–531.

Wehrle, T. (1998). Motivations behind modeling emotions agents: Whose emotions does your robot have? In *SAB'98 Workshop on Grounding Emotions in Adaptive Systems*.

Whaite, P. and Ferrie, F. P. (1993). Autonomous exploration: Driven by uncertainty. TR-CIM 93-17, McGill Centre for Intelligent Machines.

Wilson, S. W. (1991). The animat path to AI. In *From animals to animats — Proceedings of the First Conference on Simulation of Adaptive Behavior*. The MIT Press.

Winograd, T. and Flores, F. (1986). *Understanding computers and cognition*. Addison-Wesley Publishing Company.

Wright, I. (1994). An emotional agent – the detection and control of emergent states in an autonomous resource-bounded agent (phd thesis proposal). Technical report, University of Birmingham.

Wright, I. (1996). Reinforcement learning and animat emotions. In Maes, P., Mataric, M. J., Meyer, J.-A., Pollack, J., and Wilson, S. W., editors, *From animals to animats 4 — Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 273–281. The MIT Press.

Wyatt, J., Hoar, J., and Hayes, G. (1998). Design, analysis and comparison of robot learners. *Robotics and Autonomous Systems: Special Issue on quantitative methods in mobile robotics*, 24(1-2):17–32.

Yavnai, A. (1989). Criteria for systems autonomability. In Kanade, T., Groen, F. C. A., and Hertzberger, L. O., editors, *Intelligent Autonomous Systems 2*.

Zajonc, R. B. (1984). On primacy of affect. In Scherer, K. R. and Ekman, P., editors, *Approaches to Emotion*. Lawrence Erlbaum, London.

Zajonc, R. B., Pietromonaco, P., and Bargh, J. (1982). Independence and interaction of affect and cognition. In *Affect and cognition — The Seventeenth Annual Carnegie Symposium on Cognition*. Lawrence Erlbaum Associates.

Appendix A

Emotions model

A.1 Parameters

The system parameters used in the emotions model's functions are the following:

$$
\begin{aligned}
C_{ef} &= \text{Value of the emotion-feeling dependency between emotion } e \text{ and feeling } f \\
B_e &= \text{Value of the bias of emotion } e \\
I_{th_a} &= \texttt{EmotionActiveTH} \\
I_{th_s} &= \texttt{EmotionSelectTH} \\
C_h &= \texttt{HormCoef} \\
\alpha_{up} &= \texttt{HormAlphaUp} \\
\alpha_{dn} &= \texttt{HormAlphaDn}
\end{aligned}
$$

These parameters can take values within $[0,1)$ apart from C_{ef} and B_e which can take values within $(-1,1)$.

$$-1 < \quad C_{ef}, B_e \quad < 1 \tag{A.1}$$

$$0 \leq \quad C_h, I_{th_a}, I_{th_s}, \alpha_{up}, \alpha_{down} \quad < 1 \tag{A.2}$$

It is also assumed that the sensations (S_{f_n}) have been normalised to the range $[0,1]$.

Furthermore, to prevent the system from saturating by the feedback of values through the hormone system, it was found necessary to make extra restrictions to the system parameters. In practice, it must be guaranteed that if no stimulus is available then the hormone values will decrease.

$$\text{If} \quad S_{f_n} = 0 \quad \text{then} \quad |H_{f_{n+1}}| < |H_{f_n}| \tag{A.3}$$

The following restrictions allow to guarantee this. The sum of the positive coupling coefficients associated with an emotion (C_e^+) or with a feeling (C_f^+) should be limited in the following way:

$$\forall e \in \mathcal{E}, \quad C_e^+ = B_e^+ + C_h \sum_{f \in \mathcal{F}} C_{ef}^+ \quad \leq 1 \tag{A.4}$$

$$\forall f \in \mathcal{F}, \quad C_f^+ = \sum_{e \in \mathcal{E}} C_{ef}^+ \quad < 1 \tag{A.5}$$

$$C_{ef}^+ = \begin{cases} C_{ef} & \text{if } C_{ef} > 0 \\ 0 & \text{otherwise} \end{cases} \tag{A.6}$$

$$B_e^+ = \begin{cases} B_e & \text{if } B_e > 0 \\ 0 & \text{otherwise} \end{cases} \tag{A.7}$$

Care should also be taken to have C_{ef} high enough to guarantee the emotions to be active when necessary and to enable them to take high values.

Consult the appendices concerning experimental details to know which parameter values were used. The emotions' bias (B_e) and dependencies on feelings (C_{ef}) used in the different control strategies are specified in the main text while the implementation details of how the sensations are calculated from the sensors are specified in the appendices.

Appendix B

Presentation of Experimental Results

B.1 Error Bar Calculation

B.1.1 Single experiment

The error bars displayed on the graphs assume a normal distribution of the raw values and are based on the following calculations. The mean and the standard deviation of the means obtained for each one of the runs was calculated.

So supposing that $x_1, x_2 \cdots x_n$ are the means obtained for each of the n runs at a particular testing point, the following standard formulae show how the mean (μ) and the standard deviation (σ) are calculated.

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{B.1}$$

$$\sigma = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \mu)^2} \tag{B.2}$$

The error bars are based on the confidence interval of $1-q$, where $q = 5\%$. Their value is $\mu \pm \Delta$. Δ is calculated as shown in the following equation.

$$\Delta = \chi_{1-\frac{q}{2}} \frac{\sigma}{\sqrt{n}} \tag{B.3}$$

$$\chi_{1-\frac{q}{2}} = 1.96 \tag{B.4}$$

B.1.2 Difference between two experiments

In some cases, it is interesting to show the difference between two experiments, instead of each one of them by themselves. This is only done for experiments of the same size.

In these cases, the mean and standard deviation of the final displayed results are calculated from the individual mean and standard deviation of each of the two experiments. So consider that, through the calculations specified above, the mean and standard deviation obtained are μ_r and σ_r for one of the experiments, and μ_b and σ_b for the other. Supposing that the latter is taken to be the base experiment and that the distributions of the data are independent from experiment to experiment, the mean and standard deviation of the difference between the two are:

$$\mu = \mu_r - \mu_b \tag{B.5}$$

$$\sigma = \sqrt{\sigma_r^2 + \sigma_b^2} \tag{B.6}$$

Once again, the error bars shown in the graphs are $\mu \pm \Delta$, where Δ is the same function of σ as before.

Appendix C

Action-Based Control Experimental Details

C.1 Sensations Specification

Energy level and sensation intensities are values bounded between zero and one.

The robot is initialised with maximum energy level. To lose all its energy, the robot takes `EnergyAutoStopSteps` iteration steps if it is stopped, and `EnergyAutoRunSteps` if it is moving at full speed. The decrease in energy level is proportional to the total motor activity (*i.e.* the sum of the absolute values of both left and right motor values). `EnergyRechargSteps` is the number of steps necessary to recover all its energy if the robot receives maximum light on its sensors. The energy will increase only if the sum of the values of the robot's front light sensors is high enough (*i.e.* 60% of its maximum value). If that condition is met, then the increase in energy is directly proportional to the light received by these sensors. The previous descriptions of the processes of increase and decrease of energy level assume independent processes, *i.e.* each description considers that the energy value is only modified by the process being described. In reality, the effects of both processes in the energy level are calculated separately and subsequently added.

Hunger is one minus the energy level.

The **Eating** sensation is non-zero and directly proportional to the light perceived if the light received by the robot's front sensors is considered high enough to increase its energy level. If the energy level is very high (> 0.95) the Eating sensation is in addition multiplied by one minus the energy level, *i.e.* the Hunger sensation.

If the robot is bumping then the **Pain** is proportional to the number of distance sensors with high values (over 1020), otherwise it is zero. The pain value starts at 1/3 and increases by 1/6 for each high-valued distance sensor until it reaches its maximum value. This means that the Pain sensation does not differentiate between 3 or more high-valued sensors.

If the robot travels a good distance (Manhattan distance higher than 1), its **Restless-**

ness will decrease; otherwise it increases. The parameter `BoredomRaiseSteps` is the number of steps the robot has to be totally stopped to reach its maximum restlessness value. While the increase in restlessness is inversely proportional to `BoredomRaiseSteps`, the decrease is inversely proportional to the parameter `BoredomLowerSteps`. The change in restlessness intensity is directly proportional to one minus the Manhattan distance (*i.e.* the sum of the absolute values of the distances covered in x and in y), which lies between -4 and 1.

If the total motor activity is above the `TempRaiseTh` threshold then the **Temperature** will rise. If the value is low (*i.e.*, motor activity $<$ `TempLowerTh`), then the temperature will decrease. Basically, the robot will lose temperature with all actions in its action set except for the fast-forward action. When motors are at full power (motor activity $= 20$), the robot takes `TempRaiseSteps` steps to go from no temperature to maximum temperature. It will take `TempLowerSteps` to lower its temperature back to zero with the motors off. The temperature increase is directly proportional to the total motor activity and the decrease is directly proportional to one minus the rescaled total motor activity (rescaled to lie between 0 and 1).

The values used for the constants mentioned above are given in Table C.1.

C.2 System Parameters

Emotions Parameters	
EmotionActiveTH	0.2
EmotionSelectTH	0.2
HormCoef	0.9
HormAlphaUp	0.98
HormAlphaDn	0.996

Controller Parameters	
Learning Rate	0.1
Selection Temperature	0.1

Sensation Constants	
EnergyAutoStopSteps	1000000
EnergyAutoRunSteps	50000
EnergyRechargSteps	2000
BoredomRaiseSteps	600
BoredomLowerSteps	600
TempRaiseSteps	200
TempLowerSteps	1000
TempRaiseTh	14
TempLowerTh	10

Table C.1: Parameter values used in the experiments.

C.3 Actions

The set of discrete actions used in the experiments is defined by the motor values shown in Table C.2. The second set of action used in the experiments reported in Section 4.3.4 is defined by the same motor values, except for the backwards movement.

Actions	Motor power	
	Right motor	Left motor
Slow forward	3	3
Turn left	5	2
Turn right	2	5
Fast forward	8	8
Stop	0	0
Backwards	-1 (-3)	-5 (-3)

Table C.2: Values of the robot's motors for each action. The values in brackets are the modified values for the second set of actions.

C.4 Program Environment and Performance

The program developed, *sim*, is an extension of the X-windows simulator by Olivier Michel (Michel, 1996). The original code was in C, but the extensions were done in C++. Some extensions were made to the original code in order to allow it to run in the background with no graphical output.

The *sim* program takes about five hours to run an experiment with one hundred and twenty thousand learning steps and sixty-one two thousand step evaluation tests (*e.g.* the experiment reported in Figure 4.8). The execution time reported was obtained for a Sun SparcStation 4 at 125 MHz with *sim* running in the background with low priority while other programs were running on the same machine.

C.5 Earlier Experiments

The experiments reported in this dissertation are the end result of many other experiments that provided insights for the gradual improvement of the system.

Much of the improvement consisted in having a more adequate adaptive controller whether by changing the networks used or the learning algorithms. Others introduced changes in the emotional system with the purpose of achieving a more stable system able to provide a reinforcement function more adequate for learning interesting behaviour.

This involved re-design of the basic emotions, by changing the existing relations between feelings and emotions, changing many of the system functions and even redefining the feelings. Although the magnitudes of the reinforcements received and the final behaviours learned by this early system where quite different, the conclusions that were derived are consistent with the ones reported in the main text.

Some of the differences of one of these earlier systems when compared with the final version follow.

The functions used by the earlier emotion system which are different are:

$$\forall f \in \mathcal{F}, \forall n \in \mathbb{N}, \quad H_{f_{n+1}} = \alpha H_{f_n} + (1-\alpha)\, \tanh(d\, A_{f_n}) \tag{C.1}$$

$$\alpha = \texttt{HormAlpha} \tag{C.2}$$

$$d = \texttt{HormDeclive} \tag{C.3}$$

The Temperature and Pain sensations and the energy level were calculated in a sightly different fashion. The main differences being:

- the temperature would only decrease if the action taken was the stop action;
- even if the robot was not bumping, pain could be non-zero if distance sensors with high values existed;
- the light sensor used for increasing the energy level was the middle right light sensor. This selection was made in the assumption that a slightly offset sensor should provide a more difficult task. However, this was not case.

In addition, the turning actions of the action set were associated with the following motor values:

Actions	Motor power	
	Right motor	Left motor
Turn left	7	3
Turn right	3	7

Furthermore, the system parameters had the values shown in Tables C.3 and C.4.

	Hunger	Pain	Restlessness	Temperature	Eating	Bias
Happiness	-0.4	-0.3	-0.2	0.4	0.7	0.15
Sadness	0.7	0.0	0.5	0.0	-0.4	0.0
Fear	-0.4	0.6	-0.2	0.4	0.0	0.0
Anger	0.2	0.4	0.3	-0.2	0.0	0.0

Table C.3: The emotions' dependencies on feelings for earlier experiments.

Finally, the robot's environment was simpler (see Figure C.1). In the more recent version, lights had to be surrounded by bricks to avoid having the robot getting stuck on the lights. This would happen quite frequently, especially with the new setup. As a result, experiments would often have to be invalidated, because the robot was caught in one of these situations where it would become helpless with no action available capable

Emotions Parameters	
EmotionActiveTH	0.2
EmotionSelectTH	0.2
HormCoef	0.5
HormAlpha	0.996
HormDeclive	3.0

Controller Parameters	
Learning Rate	0.1
Selection Temperature	0.04

Sensation Constants	
EnergyAutoStopSteps	1000000
EnergyAutoRunSteps	50000
EnergyRechargSteps	2000
BoredomRaiseSteps	600
BoredomLowerSteps	600
TempRaiseSteps	800
TempLowerSteps	1000
TempRaiseTh	14
TempLowerTh	6

Table C.4: Parameter values used in the earlier experiments.

of getting it out of its stuck position. It was found that the robot would "jump" into such positions because the simulator only verifies intersection with obstacles for the final position of the step movement, and not for all the intermediate positions. This way, the robot can arrive at invalid positions that should normally not be reachable. When this happens, in order to get back to a valid position, the robot might have to execute the exact inverse action that led to the invalid position in the first place and might be unable to free itself only because of a question of speed of the motors.

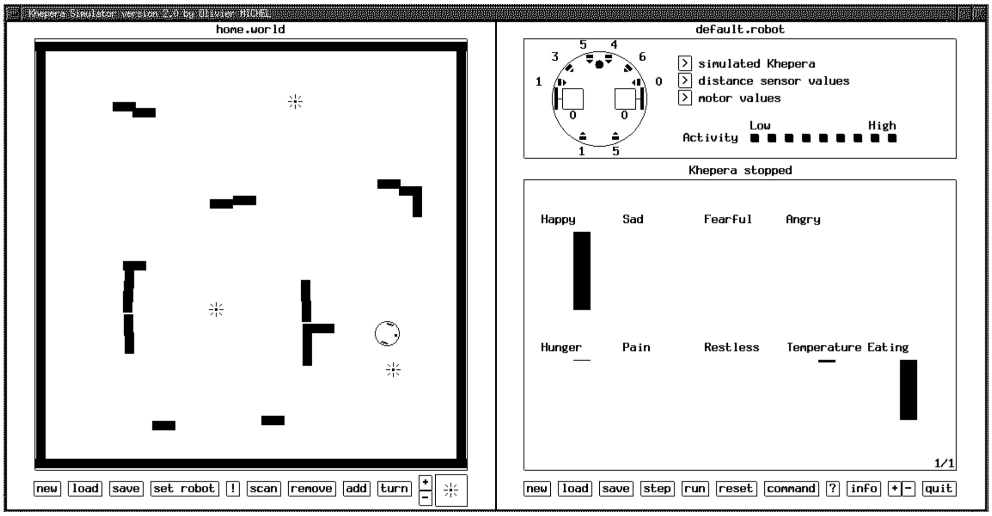


Figure C.1: The robot and its environment. Earlier version.

Appendix D

Behaviour-Based Control Experimental Details

D.1 Sensations and Food Sources Specification

As stated in the main text, the acquisition of new energy can only be made at specific intervals of time, and the sensors used for this purpose are now the two rear light sensors instead of the two front ones. To specify the implications of the robot's harder task for its sensations, this task will be described in detail next.

Each light in the robot's environment is a food source. A food source contains several food items, varying between 0 and `MaxFoodItems`, that are decreased by one every time it releases energy. Food items are continuously being produced, unless the limit of food items per food source, *i.e.* `MaxFoodItems`, is reached. A new food item is created after a random number of steps varying between 1 and `MaxNewFoodSteps`, unless production is stopped. In the latter case, a food item has to be used up first. If the robot collides with the associated light, then a food item is released in the form of available energy to the robot and therefore used up. This energy will only be available for `MaxFoodAvailableSteps` steps. Only during this interval of time can the robot acquire energy by receiving light in its rear sensors. The food items are actually only released when the robot state changes from not bumping to bumping. So after having received a food item, to have a new one, the robot has to back out and hit the light again. This avoids all the food items being released in one go when the robot keeps in collision with the light for a few steps.

For simplification of the implementation, it is not necessary that the robot actually collide with the light or the bricks around it. The robot has only to collide with some wall within a pre-specified square area around the light. Around each light, there is such an area. These areas are represented in Figure D.1 by the smaller and lighter areas around the lights. This figure also shows the larger areas, represented by a less bright colour, where the robot can smell and eventually eat the food provided by the light.

The other modifications introduced to previously existing sensations are:

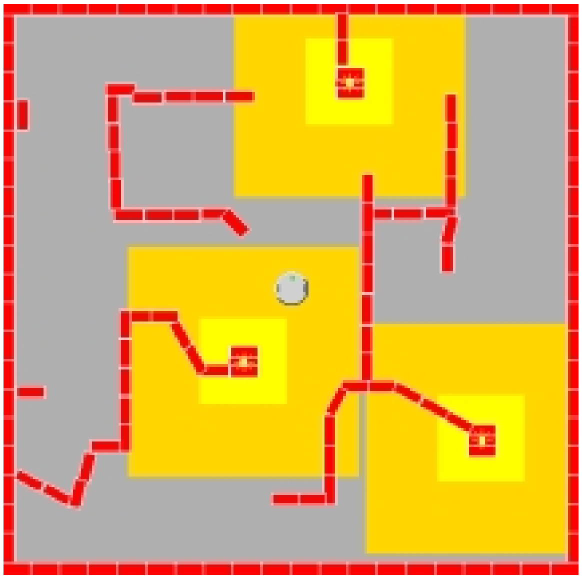


Figure D.1: Different regions for energy acquisition.

- **Temperature** — The temperature's variation thresholds were lowered because the actions employed by the behaviours tend to be smaller than the primitive actions previously used.
- **Restlessness** — Is calculated just the same way as before, but is reset to value zero, together with the associated hormone, whenever the controller selects a behaviour.

Three new sensations were introduced. These, like the original sensations, also have values bounded between zero and one.

- **Smell** — is only active if there is food available and its intensity is directly proportional to the number of time steps it will still be available. It has the maximum value of one when the food is made available.
- **Warmth** — is the normalised value of the light sensor that is receiving most light at the moment; the highest the the intensity of the light received in that sensor, the higher the value of warmth.
- **Proximity** — is the normalised value of the distance sensor with the highest value. Reflects the proximity of the nearest obstacle.

The parameter values used by the behaviour-based controller in the calculation of the sensations and food availability are given in Section D.2.

D.2 System Parameters

Figure	Experiment	Reinforcement	Environment	Parameters
5.5, 5.6, 5.7, 5.8	Short	Sensation	–	–
5.9	–	Sensation	–	–
5.10, 5.11	Short	Sensation	–	–
5.12	–	N. A.	–	–
5.14	–	Sensation	–	–
5.15	Event-driven	Sensation	–	–
5.16, 5.17	–	Sensation	–	–
5.18	–	Sensation	Demanding	–
5.19, 5.20	–	Sensation	Demanding	Energy
5.22, 5.23	–	N. A.	–	–
5.24	–	Emotion	–	–
5.26	–	Emotion	–	–
5.28	–	N. A.	–	–
5.29	–	N. A.	Demanding	Energy
5.30, 5.31	Follow-up	Emotion	–	–
5.33, D.2, D.3	–	Emotion	–	–
5.35	–	Emotion	Demanding	Energy + Hormone decay
5.36	–	Emotion	Demanding	Energy + No hormones
5.38	–	Emotion	–	–
5.39	–	Emotion	Demanding	Energy

Table D.1: Experimental procedure for individual experiments.

Interpretation of Table D.1:

Figure — Indicates the figures where the experimental results are shown. The list of figures provides a brief description of the experiments themselves.

Experiment — The difference between short experiments and normal experiments lies in the number of steps per trial, see Section 5.2.4 for details. Some of the experiments are actually follow-ups on other experiments, showing particular details of the behaviour of a controller after learning has converged.

Reinforcement — Specifies whether sensation-dependent or emotion-dependent reinforcement was used during the experiments.

Environment — The different available environments, the default and the demanding, are pictured in Figures 5.2 and 5.1, respectively.

Parameters — The default parameters are the ones on Table D.2, the modifications made are specified in Tables D.3 (energy), D.4 (no hormones) and D.5 (hormone decay).

Emotions Parameters	
EmotionActiveTH	0.2
EmotionSelectTH	0.2
HormCoef	0.9
HormAlphaUp	0.98
HormAlphaDn	0.996

Controller Parameters	
Learning Rate	0.3
Selection Temperature	0.1

Triggering Parameters	
Tolerance Threshold	0.02
Maximum Step Limit	10000

Sensation Constants	
EnergyAutoStopSteps	100000
EnergyAutoRunSteps	20000
EnergyRechargSteps	100
BoredomRaiseSteps	1000
BoredomLowerSteps	200
TempRaiseSteps	200
TempLowerSteps	1000
TempRaiseTh	10
TempLowerTh	3

Food Constants	
MaxFoodItems	5
MaxNewFoodSteps	20000
MaxFoodAvailableSteps	200

Table D.2: Parameter values used in the experiments.

Sensation Constants	
EnergyAutoStopSteps	15000
EnergyAutoRunSteps	15000

Table D.3: Modified parameters for harder energy requirements.

Emotions Parameters	
HormCoef	0
HormAlphaUp	0
HormAlphaDn	0

Table D.4: Modified parameters for no influence from hormone system.

Emotions Parameters	
HormAlphaDn	0.99, 0.998, 0.999

Table D.5: Modified parameter values for hormone decay rate.

D.3 Behaviours

For an accurate description of the behaviours, their pseudo-code is given next starting by the presentation of the constants[1] and auxiliary functions used.

```
% Constants
SmallProximityValue ← 10;
MediumProximityValue ← 400;

AllSensors ← { 0, 1, 2, 3, 4, 5, 6, 7 };
LeftSensors ← { 0, 1, 2 };
RightSensors ← { 3, 4, 5 };
RearSensors ← { 6, 7 };

RightOrientTo ← [ 8, 8, 10, 8, 2, -4, -10, -10 ];
LeftOrientTo ← [ -4, 2, 8, 10, 8, 8, 10, 10 ];

RightWall.Th ← [ 5, 5, 5, 5, 100, 800, 5, 5 ];
RightWall.left ← [ -1, -1, -1, 5, 8, 8, 0, 0 ];
RightWall.leftSum ← 100 × $\sum$ | RightWall.left[$\forall$ i $\in$ AllSensors] |
RightWall.right ← [ 1, 1, 1, -5, -2, -2, 0, 0 ];
RightWall.rightSum ← 100 × $\sum$ | RightWall.right[$\forall$ i $\in$ AllSensors] |

LeftWall.th ← [ 800, 100, 5, 5, 5, 5, 5, 5 ];
LeftWall.left ← [ -2, -2, -5, 1, 1, 1, 0, 0 ];
LeftWall.leftSum ← 100 × $\sum$ | LeftWall.left[$\forall$ i $\in$ AllSensors] |
LeftWall.right ← [ 8, 8, 5, -1, -1, -1, 0, 0 ];
LeftWall.rightSum ← 100 × $\sum$ | LeftWall.right[$\forall$ i $\in$ AllSensors] |
```

```
function BoundMotorValue(x)
    return max( -10, min( 10, x));
end
```

[1] Some of the wall-following constants were actually determined by genetic algorithms, thanks to Hanson Schmidt-Cornelius.

```
function ApplyReflex(reflex, proximity)
    leftMotor $\leftarrow$ 0;
    rightMotor $\leftarrow$ 0;
    For i $\in$ AllSensors do
        leftMotor $\leftarrow$ leftMotor + reflex.left[i] $\times$ (reflex.th[i] - proximity[i]);
        rightMotor $\leftarrow$ rightMotor + reflex.right[i] $\times$ (reflex.th[i] - proximity[i]);
    leftMotor $\leftarrow$ BoundMotorValue((leftMotor / reflex.leftSum) + 1 );
    rightMotor $\leftarrow$ BoundMotorValue((rightMotor / reflex.rightSum) + 1);
end
```

```
Behaviour Avoid Obstacles
    For i $\in$ AllSensors do
        proximity[i] $\leftarrow$ Value registered by distance sensor i;
    end
    $i_{max}$ $\leftarrow$ i : proximity[i] $\geq$ proximity[$\forall$ j $\in$ AllSensors];
    if ($i_{max} \notin$ RearSensors) or
      (| proximity[6] - proximity[7] | > MediumProximityValue) do
      % Turn back to the obstacle
      leftMotor $\leftarrow$ 4;
      rightMotor $\leftarrow$ -4;
      return;
    end
    leftMotor $\leftarrow$ (proximity[7] - max(proximity[$\forall$ i $\in$ RightSensors])) / 51;
    rightMotor $\leftarrow$ (proximity[6] - max(proximity[$\forall$ i $\in$ LeftSensors])) / 51;
end
```

```
Behaviour Seek Light
    For i $\in$ AllSensors do
        value[i] $\leftarrow$ Value registered by light sensor i;
        light[i] $\leftarrow$ 450 - value[i];
    end
    if max(light[$\forall$ i $\in$ AllSensors]) < 0 do
        % No lights nearby
        leftMotor $\leftarrow$ 0;
        rightMotor $\leftarrow$ 0;
        return;
    end
    adjust $\leftarrow$ 2 $\times$ max(light[$\forall$ i $\in$ AllSensors]);
    leftMotor $\leftarrow$ BoundMotorValue((light $\times$ LeftOrientTo)/adjust);
    rightMotor $\leftarrow$ BoundMotorValue((light $\times$ RightOrientTo)/adjust);
end
```

```
Behaviour **Wall Following**
    For i $\in$ AllSensors do
        proximity[i] $\leftarrow$ *Value registered by distance sensor i*;
    end
    $i_{max}$ $\leftarrow$ i : proximity[i] $\geq$ proximity[$\forall$ j $\in$ AllSensors];
    if (max(proximity[$i_{max}$]) $<$ SmallProximityValue) or $i_{max}$ $\in$ RearSensors do
        *% No walls nearby or on the back of the robot*
        wall $\leftarrow$ none;
        leftMotor $\leftarrow$ 10;
        rightMotor $\leftarrow$ 10;
        return;
    end
    if wall = none do
        if $i_{max}$ $\in$ LeftSensors do
            wall $\leftarrow$ left;
        else
            wall $\leftarrow$ right;
        end
    end
    if wall = left do
        ApplyReflex(LeftWall, proximity);
    else
        ApplyReflex(RightWall, proximity);
    end
    leftMotor $\leftarrow$ BoundMotorValue(6 $\times$ leftMotor);
    rightMotor $\leftarrow$ BoundMotorValue(6 $\times$ rightMotor);
end
```

D.4 Additional Experimental Results

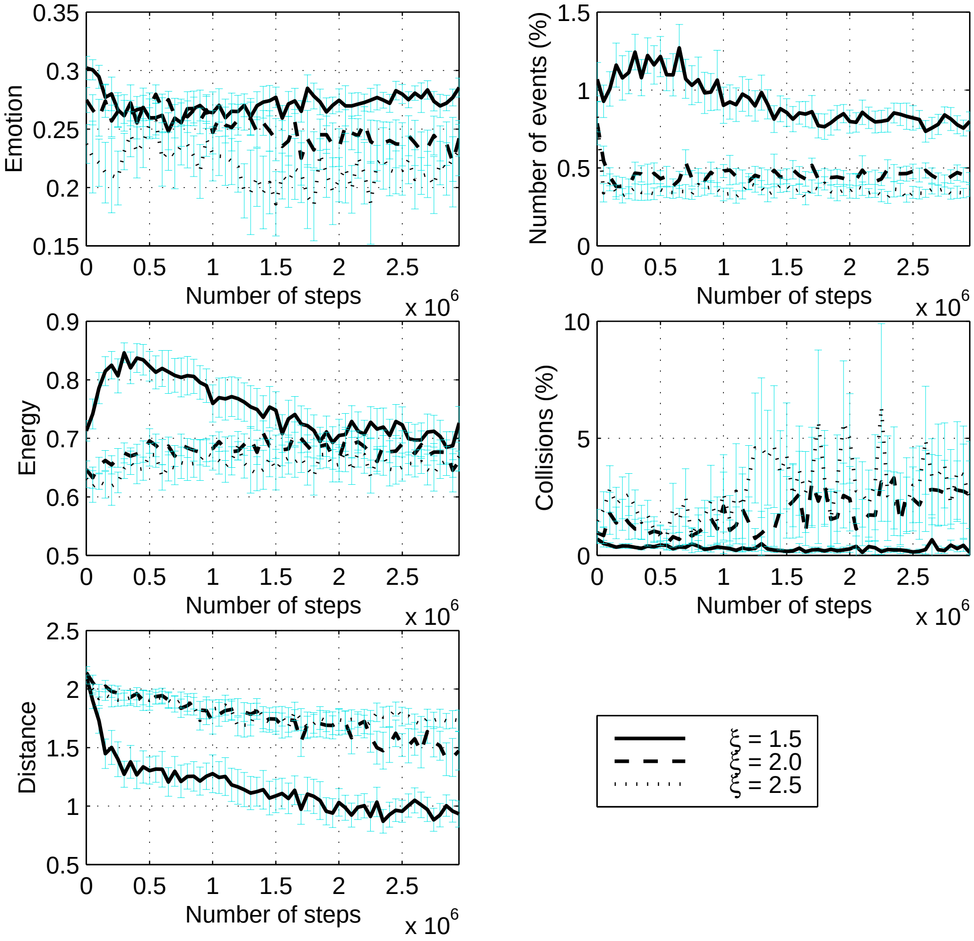


Figure D.2: Emotional controller starting off with customised neural networks. Repetition of learning experiment in Figure 5.33 testing different ξ values.

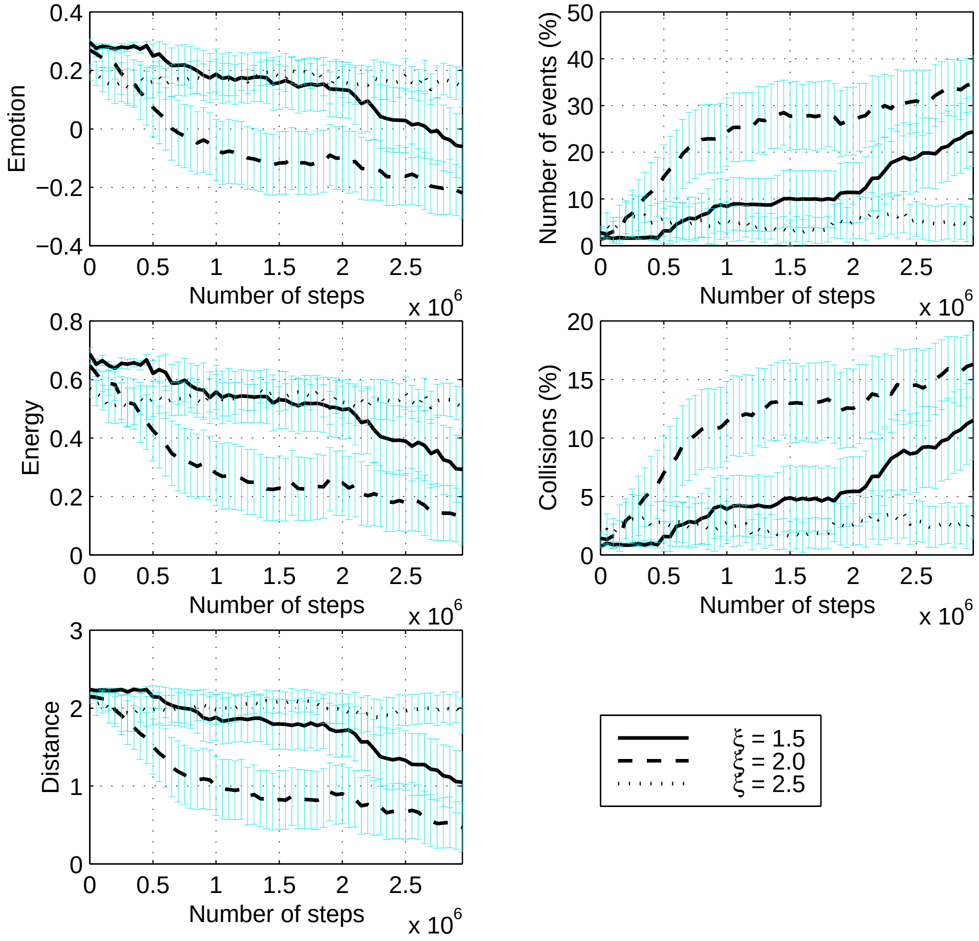


Figure D.3: Emotional controller with no learning and starting off with customised neural networks. Repetition of non-learning experiment in Figure 5.33 testing different ξ values.

D.5 Program Environment and Performance

The new controller demanded further modifications to the Khepera simulator, apart from those already done for the action-based controller. As discussed previously, sometimes the robot would end up in crash situations where all its available actions were useless to move it away. With the new controller the problems with robot crash situations increased dramatically, making it necessary to change the Test Collision routine of the simulator. Minor changes to the routine were sufficient to make it much more robust and avoid crashing problems altogether. As a side effect, the overall performance of the robots, not taking into account the crashed robots, also improved slightly.

The new controller is more complex and takes more steps to run. This resulted in a significant increase in the computing time of each experiment. Now a normal experiment, as described in Section 5.2.4 and usually corresponding to one of the curves of the results graphs, takes about ten hours processing time on a Sun Ultra 5 workstation.

Appendix E

Publications

This appendix contains the following papers:

Gadanho, S. C. and Hallam, J. (1998). **Emotion-driven Learning for Animat Control**. In *From animals to animats 5.* Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior, pages 354-359. The MIT Press.

Gadanho, S. C. and Hallam, J. (1998). **Emotion-triggered Learning for Autonomous Robots**. In *SAB'98 Workshop on Grounding Emotions in Adaptive Systems.*

Gadanho, S. C. and Hallam, J. (1998). **Exploring the Role of Emotions in Autonomous Robot Learning**. In *AAAI Fall Symposium — Emotional and Intelligent: The tangled knot of cognition*, Technical Report FS-98-03, pages 84-89. AAAI Press.