Emotion-triggered Learning in Autonomous Robot Control

Sandra Clara Gadanho and John Hallam
Department of Artificial Intelligence
University of Edinburgh
5, Forrest Hill
Edinburgh EH1 2QL

{sandrag,john}@dai.ed.ac.uk

Telephone: +44 131 6503084
Fax: +44 131 6506899

Abbreviated Title: Emotion-triggered Robot Learning

Emotion-triggered Learning in Autonomous Robot Control

Sandra Clara Gadanho and John Hallam
Department of Artificial Intelligence, University of Edinburgh
5, Forrest Hill, Edinburgh EH1 2QL
{sandrag,john}@dai.ed.ac.uk

Abstract

The fact that emotions are considered to be essential to human reasoning suggests that they might play an important role in autonomous robots as well. In particular, the decision of when to interrupt on-going behaviour is often associated with emotions in natural systems. The question under examination here is whether this role of emotions can be useful for a robot which adapts to its environment.

For this purpose, an emotion model was developed and integrated in a reinforcement-learning framework. Robot experiments were done to test an emotion-dependent mechanism for the automatic detection of the relevant events of a learning task, against more traditional approaches. Experimental results are presented that confirm that emotions can be useful in this role, specifically by improving the efficiency of the learning algorithm.

Introduction

In recent years, the importance of emotions and their assistance to cognition has been increasingly acknowledged. For example, Toda (1994) argues that emotions are the ultimate source of intelligence and might provide robots with the autonomy they need. Doubts have even been posed on whether machines can exhibit intelligent behaviour without emotions (Minsky, 1986; Charland, 1995).

In robotics, emotions are often used to modulate activity (Cañamero, 1997; Bates et al., 1992a). The social role of emotions has been particularly explored. The external demonstration of emotions has been used as a sort of communication mechanism that allows the robot to report to others its internal state (e.g. its level of task achievement Shibata et al., 1996) or makes the robot capable of generating empathy emotions in people by creating an illusion of life in a believable character (Bates, 1994).

In contrast, in the research reported here emotions are used in the control of a solitary autonomous robot that adapts to its environment using reinforcement learning. The work was done under an animat philosophy (Wilson, 1991a), by building a biologically inspired

[1]The first author is supported by a scholarship from the Portuguese programme PRAXIS XXI. Facilities for this work were provided by the University of Edinburgh.

complete agent where emotions form an integral part of the whole. Although the experiments reported here focus on the evaluation of the emotion-dependent event-detection mechanism, emotions were also used to influence perception and provide a reinforcement function within the reinforcement-learning framework.

Reinforcement-learning (*e.g.*, Sutton and Barto, 1998; Kaelbling et al., 1996) 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. 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. The reinforcement function value can also be zero meaning that no reward or punishment was attributed and that the evaluation is neutral. In opposition to simpler techniques, 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.

One of the most important design problem faced when employing reinforcement learning techniques in robotics applications is to determine when a discrete state transition occurs, *i.e.* when the controller needs to re-evaluate its previous decision and make a new one, since reinforcement-learning techniques assume that the system is an Markov decision process. An incorrect state transition design can be fatal to the success of the learning agent Gadanho (1999).

There are several approaches to the definition of state transition. Some researchers (*e.g.*, Mahadevan and Connell, 1992; Lin, 1993; Mataric, 1994) extend the duration of the current action according to some domain specific conditions of goal achievement or applicability of the action while others 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.

Emotions are 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), precisely because 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).

Taking as inspiration the emotions' role of interrupting behaviour in natural systems, the current work explores the usefulness of emotions in determining state transitions in a reinforcement-learning system.

In the next section, a description of the emotion model developed is presented. This is a non-symbolic model that takes the form of a recurrent artificial neural network where emotions influence the perception of the state of the world, from which they ultimately depend. This model is afterwards integrated in a reinforcement learning architecture.

The experiments done with this model are reported in the following section. 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). 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. The emotion-dependent mechanism under study, *i.e.* the event-detection mechanism, was therefore experimentally compared with other approaches. Experiments were carried out on a simulated Khepera robot (Michel, 1996) in an animal-like adaptation task.

Experimental results demonstrate that the proposed event-detection mechanism was competent and competitive, proving emotions helpful for the robot's success in its task.

Emotion Model

A large subset of theories of emotions is based on elaborate cognitive appraisal theories (*e.g.*, Lazarus, 1982; Power and Dalgleish, 1997) 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 without the need for high level reasoning processes (Zajonc, 1984).

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;

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.

In opposition to the traditional approach, a synthetic bottom-up approach based on the animat approach (Wilson, 1991b) was preferred for the current work which made the existing models inadequate, because they are over-designed and too complex (Pfeifer, 1994). Recently, models have been suggested that also follow a bottom-up approach (Velásquez, 1998; Cañamero, 1997; Foliot and Michel, 1998; Wright, 1996).

The most significant features of emotions that the model proposed in this document tries to capture are:

- Emotions have valence, *i.e.*, they provide a positive or negative 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 distorted 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 1 — is based on four basic emotions: Happiness, Sadness, Fear and Anger. These emotions were selected because they are among the most universal emotions and are adequate and useful for the robot–environment interaction afforded by the experiments (Gadanho, 1999).

The model determines the intensity of each emotion based on the robot's current internal feelings: Hunger, Pain, Restlessness, Temperature, Eating, Smell, Warmth and Proximity. These feelings are described below in the context of the experimental setup. Each emotion is defined by a set of constant feeling-dependencies and a bias. The values of the dependencies are carefully chosen to provide adequate emotions for the possible body states.

Based on what was suggested in (Damásio, 1994), the emotion state should also influence the way the robot feels. In the model, the body reactions aroused by emotions also give rise to

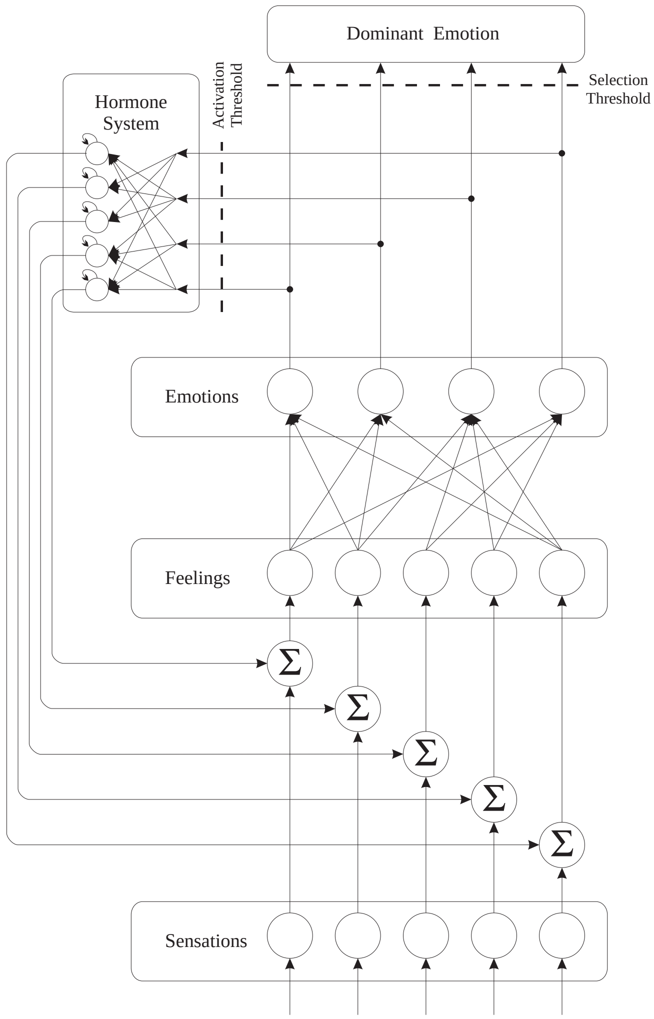

Figure 1: Emotions model.

the emotions that create them. Each emotion tries to influence the body state in such a way that the resulting body state matches the state that gives rise to that particular emotion.

When an emotion is active, *i.e.* its intensity value is significantly large, then it will influence the body through a hormone system. The hormone system in the model is a very simplified one. It consists in 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 provides competition between the emotions to gain control over the body which is ultimately what selects which emotion will be dominant. On the other hand, what the robot feels is not only dependent on its sensations but is also dependent on its emotional state.

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

The dominant emotion is the emotion with the highest intensity, unless no emotion intensity exceeds a selection threshold. 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 "good" are positive (only Happiness, in the set of emotions used), the others are considered negative.

In summary, the model of emotions described (a formal description is available in the appendix) provides not only an emotional state, based on simple feelings, that is coherent with the current situation, but also influences the body perception. Although the model is somewhat more sophisticated than those usually found in equivalent non-symbolic systems, it is based on a simplified hormone system and is far from the complexity of true emotions experienced by humans. In fact, it does not aim to model human emotions' complexity, but only to model simple emotions afforded by the agent's interaction with its environment.

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 body feelings, and not direct sensory input, 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 on a sub-symbolic approach do often try to model this feature (*e.g.*, Breazeal, 1998; Velásquez, 1998).

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). The next section describes the experiments done in this direction.

Experiments

The robot's task consists in collecting energy from food sources scattered throughout the environment. These food sources are actually lights so that the robot is able to distinguish them with its poor perception capabilities. The robot needs this energy to use during its functioning. It will use up energy faster if the velocity it demands from its motors is higher.

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. During this short period, the robot can acquire energy by receiving high values of light in its rear light sensors. This means that the robot must turn its back to the food source. To receive more energy the robot has to restart the whole process again 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. In time, the food source will recover its ability to provide energy again, but meanwhile the robot will be 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 the food source has no energy, the light associated with it is turned off.

The robot's 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.

Experiments were carried out in a Khepera simulated robot (Michel, 1996) within a closed environment divided by several walls and containing three lights surrounded by bricks — see figure 2. Specific implementation details for the experiments reported on this document can be found in Gadanho (1999).

The Emotion System

An emotion system was developed based on the emotion model presented previously and using feelings dependent on the following sensations:

- **Hunger:** The robot's energy deficit;
- **Pain:** High if the robot is bumping into obstacles;

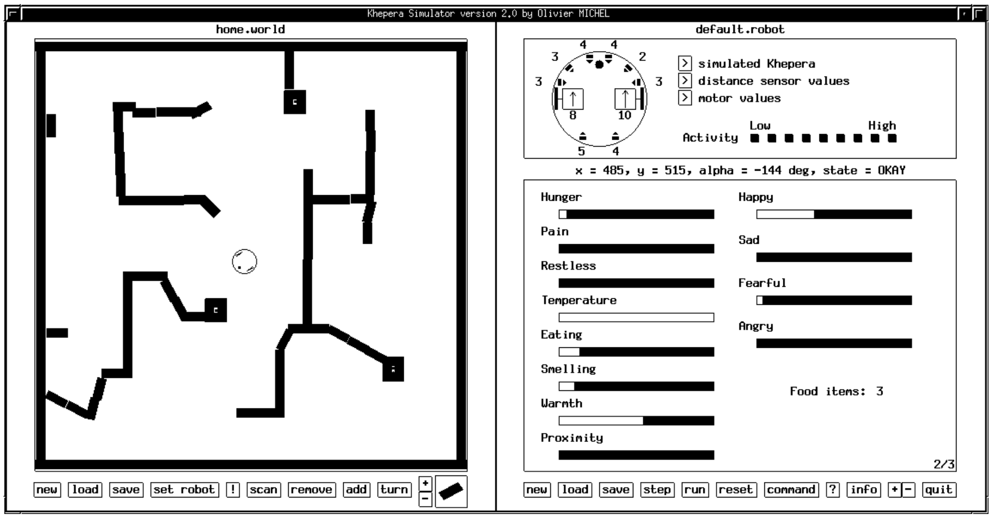

Figure 2: The robot and its environment.

- **Restlessness:** Increases if the robot does not move and 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 task, the emotional 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 the walls 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.

The Adaptive Controller

Reinforcement learning techniques have already been successfully used in the field of robotics and were therefore selected for the learning algorithm. The main problem with reinforcement learning is that learning can be very slow, particularly if the task is very complex. However, behaviour decomposition of the task can reduce significantly the learning time or even make the task feasible. Behavioural decomposition usually consists in 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 hard-wired to some extent (Mahadevan and Connell, 1992; Lin, 1993; Mataric, 1994). One problem in particular which is quite difficult and task dependent is determining when to change behaviour. This is not a problem in traditional reinforcement leaning 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). In our work we have chosen to have the primitive behaviours hand-designed and learn only the behaviour coordination in the hope that emotions might be useful in solving this problem. Three primitive behaviours were hand-designed:

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-follow — 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. However, the avoid-obstacles behaviour can easily help in these situations.

The controller developed — figure 3 — tries to maximise the evaluation received by selecting one of the three possible behaviours, taking into account the current robot feelings, and the

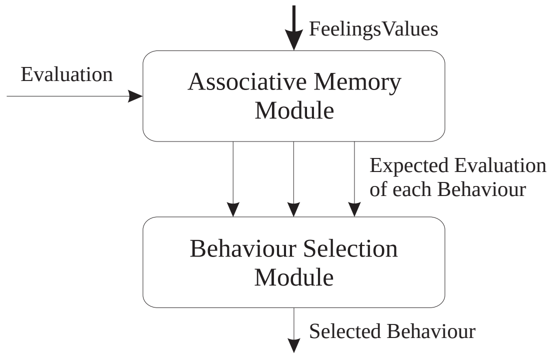

Figure 3: Adaptive controller.

previously received evaluations. It has two separate modules:

Associative Memory Module — This plastic module uses feed-forward networks to associate the robot feelings with the current expected utility of each one of the three robot behaviours. Q-learning (Watkins, 1989) was used in an implementation very similar to the one reported by Lin (1992). Neural networks learned by back-propagation utility functions that model $util(x, a) = R + \gamma eval(y)$. The value R is the immediate reinforcement. The function $eval(y)$ is the expected cumulative discounted reinforcement starting from state y reached by executing behaviour a in state x. The discount factor (γ) was set to 0.9. For each iteration, the target value $u = R + \gamma \mathrm{Max}\{util(y, k) \mid k \in actions\}$ is given to the network whose behaviour was used in the previous iteration.

Behaviour Selection Module — Based on the information provided by the previous module, this module makes a stochastic selection of the behaviour to take next. For a temperature T, the probability of selecting behaviour a whose value is v_a, is $P(a) = \frac{e^{\frac{v_a}{T}}}{\sum_{i=1}^{3} e^{\frac{v_i}{T}}}$.

Emotions and Adaptive Control

In robotics, the role of providing an evaluation of the state of the world is often attributed to emotions (*e.g.*, Albus, 1990; Wright, 1996). It is often assumed that human decision making consists in the maximization of positive emotions and minimisation of negative emotions (*e.g.*, Tomkins, 1984). Therefore a reward function was devised that extracts a value judgement from the emotion system by considering the intensity of the current dominant emotion and whether it is positive or negative. This value is the intensity of the current dominant emotion, or zero if there is no dominant emotion. If the dominant emotion is a negative one then its

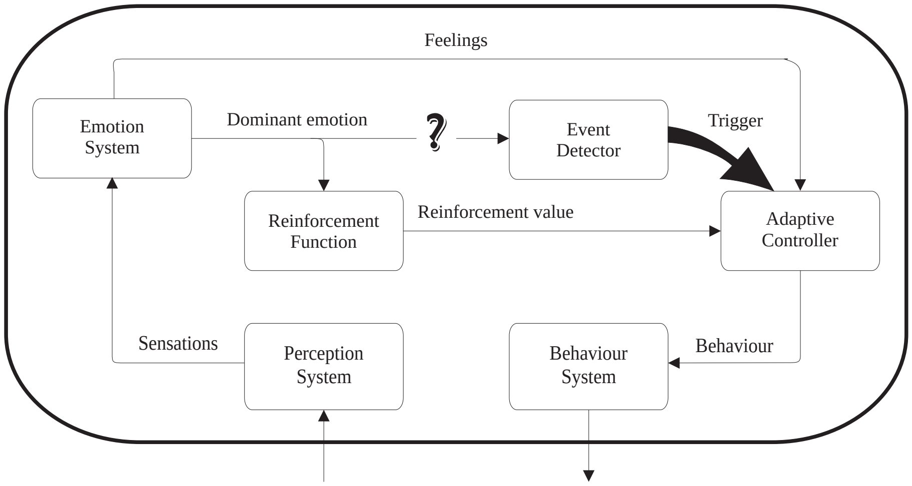

Figure 4: Emotions and control.

(positive intensity) value is negated. Although this reward function failed for a simple step by step action selection controller (Gadanho and Hallam, 1998), it proved quite satisfactory in a behaviour-based architecture and was therefore used throughout the experiments reported in this document.

In a robotic environment, a new system 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 new re-evaluation of a behaviour-based system every step by selecting a new behaviour and performing an evaluation on the previous behaviour is not wise. The problem is not so much one of too much computational waste, but mostly of not making a correct evaluation of the achievements of the behaviours. If the behaviour is evaluated and possibly replaced every step, then it will not have time to develop to its full potential. This will make it difficult for the learning system to understand what are the advantages of each 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. This work's added claim is that if the emotional intensity falls drastically, 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.

In this work, emotions directly influence perception through the emotion model used and provide reinforcement value, but the purpose of the experiments reported here was to explore *whether emotions can successfully fulfill the role of determining state transition* — see figure 4.

Testing the Hypothesis

To test the hypothesis 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. More specifically, 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 intensity 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 threshold and ξ times the standard deviation of the previous values, where ξ is a constant that was set to 2;
- A limit of 10 000 steps is reached.

If an event occurs, then the adaptive controller is triggered: an evaluation of the previous behaviour is made based on the current emotional state and a new behaviour selection is made 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 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 of very low standard deviation, imperceptible variations would be caught by the event detection mechanism.

In order to access the performance of the **Event-triggered** controller, three other controllers were designed and tested for comparison:

Interval-triggered — A simple alternative to having an event detection system is to trigger the adaptive controller at regular intervals. After extensive testing, 35 steps was found to be the most successful number of steps to have between two successive triggerings for the present task and environment. Finding the right time interval between consecutive control iterations was not trivial. If the number of steps is reduced, a proper behaviour evaluation becomes difficult, the system overall learning performance is lost and the robot is unable to maintain its energy. In particular, the inadequacy of establishing a state transition at every step, *i.e.* generating an evaluation and selection of a behaviour in every step, was shown empirically in Gadanho (1999). Results showed that a controller triggered every step can hardly learn anything useful, and its performance is not very different from that exhibited by random selection of behaviours. On the other hand, if the the number of steps is increased then the number of collisions increases, because it takes longer for the robot to notice obstacles. If increased enough the robot will also become incapable of keeping its energy, because its changes of behaviour will not be fast enough to allow acquiring new energy.

Hand-crafted — The purpose of hand-crafting a controller was to determine how much reinforcement a reasonably successfully controller would receive. To allow for a fair comparison with the other controllers, this controller only resorts to 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.

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, that allow the robot to lose sight of nearby obstacles very easily. This turned the design of a successfully non-learning controller into a slow and arduous cycle of test and redesign.

The hand-crafted controller uses the emotion-dependent event detection, with ξ equal to 1.5, *i.e.*, it uses a more sensitive event detection that is triggered by smaller variations in the emotion value. In fact, this controller performance is strongly influenced by the event triggering mechanism in use (see discussion for details).

Random — This controller simply selects a new behaviour at each step. It was included in the experiments to give a baseline to the results, 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 with time, making it difficult to see the real achievements made by the learning systems.

Four identical experiments were done, each using one of the different controllers. Each experiment consisted in having thirty different robot trials of three million learning steps. In each trial, a new fully recharged robot with all state values reset was placed at a randomly selected starting position. No distinction was made between a learning phase and a performance phase because, as a truly autonomous learning robot, the robot was designed to learn continuously. Instead the robot was evaluated while learning. For evaluation purposes, the trial period was divided in smaller periods of fifty thousand steps. For each one of these periods the following statistics were taken:

Reinforcement — mean of the reinforcement obtained during all the steps;

Event reinforcement — mean of the reinforcement obtained only during the steps in which the adaptive controller was triggered;

Energy — mean energy level of the robot;

Collisions — number of collisions;

Events — number of times the adaptive controller was triggered.

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

An average of the different statistics over the several trials is presented in figure 5, with error bars representing the 95% confidence interval. The last two statistics were presented as a percentage over the total number of steps in the period.

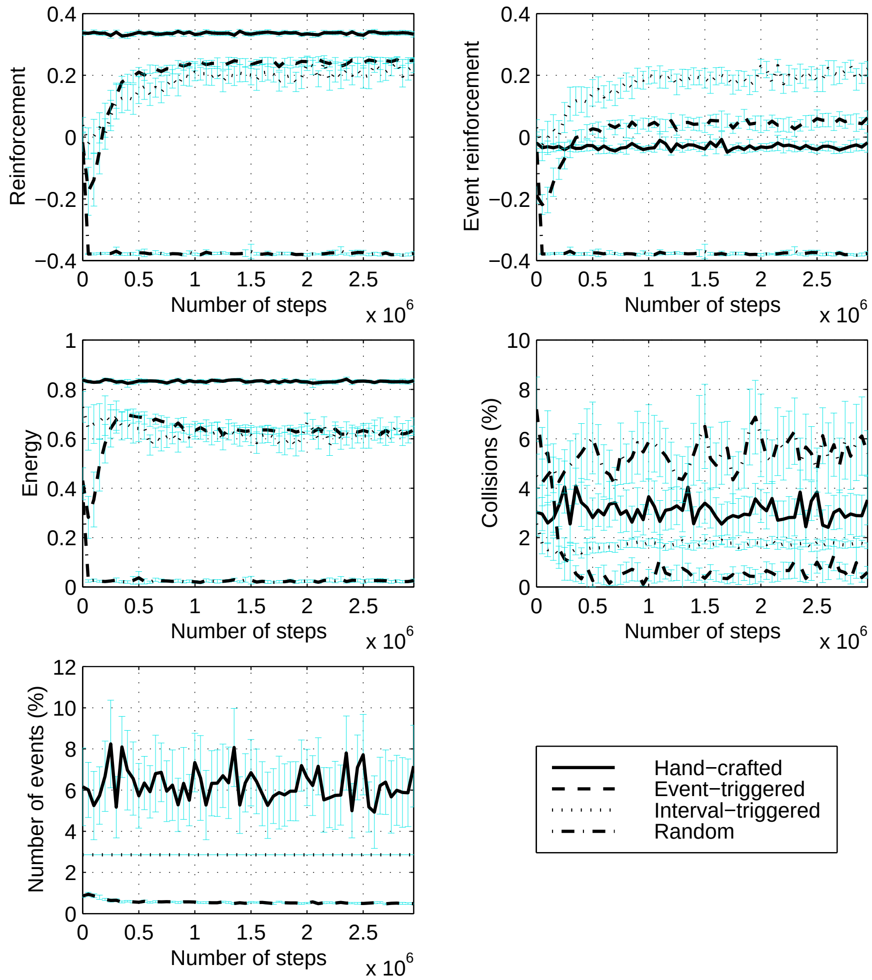

Figure 5: Value of the statistics recorded for each one of the controllers throughout the experiments.

In table 1, a summary of the results is given. Looking at the graph curves it can be safely assumed that, for every controller, learning has fully converged when a robot reaches the middle of its trial. This summary table presents the average of the values obtained from that point onwards.

Looking at the graphs in figure 5, one can see that the learning controllers do manage to learn their task. Their performance is much better than that exhibited by random behaviour selection. 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 later. In fact, in terms of obstacle avoidance the human designed controller performs worse. The lower energy is actually not a problem as long as the controllers are able to keep it relatively high above zero and this is done with success.

Controller	Reinforcement	Event Reinforcement	Energy	Collisions (%)	Events (%)
Hand-crafted	0.34 ± 0.01	-0.03 ± 0.02	0.83 ± 0.01	3.0 ± 0.8	6.1 ± 1.6
Event-triggered	0.24 ± 0.01	0.04 ± 0.02	0.63 ± 0.01	0.6 ± 0.3	0.5 ± 0.0
Interval-triggered	0.21 ± 0.03	0.20 ± 0.03	0.62 ± 0.04	1.7 ± 0.1	2.7 ± 0.0
Random	-0.38 ± 0.01	-0.38 ± 0.01	0.02 ± 0.01	5.6 ± 1.2	100.0 ± 0.0

Table 1: Summary of results obtained. The values presented are the mean of all the values obtained in the last half of the trials.

The hand-crafted controller having higher energy only shows that this controller acquires energy more often and can be at least partially attributed to the higher number of events it has available. In fact, changes in the triggering of this controller produce significant alterations in the results. When the controller was tested with $\xi = 2$, its energy level dropped to values similar to the ones found for the learning controllers. If, on the other hand, the hand-crafted controller is triggered at every step, it will eventually result in a trapped robot which is incapable of maintaining its energy. It is natural that the controller works better with the settings it was designed for in the first place, but 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 and a failed robot.

There is no significant difference in performance between the two learning controllers, apart from a slight difference in the number of collisions. In this respect the event-triggered controller does better, because the controller is triggered to deal with the obstacles that the robot finds in its way instead of having to wait until the next trigger point to deal with them. The difference in event reinforcement does not tell much apart from the fact that the event-triggered controllers are often triggered when something goes wrong: the event reinforcement of the interval-triggered controller is very similar to its overall reinforcement, because the event reinforcement values are picked from the rest at regular intervals and independently of their value; on the other hand, the event-triggered controllers are triggered in very specific situations that are often associated with negative evaluations. Typically, circumstances were that the current behaviour had to be changed, because it was not adequate anymore.

The event-triggered controller does not perform outstandingly better than its counterpart, but, it 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.

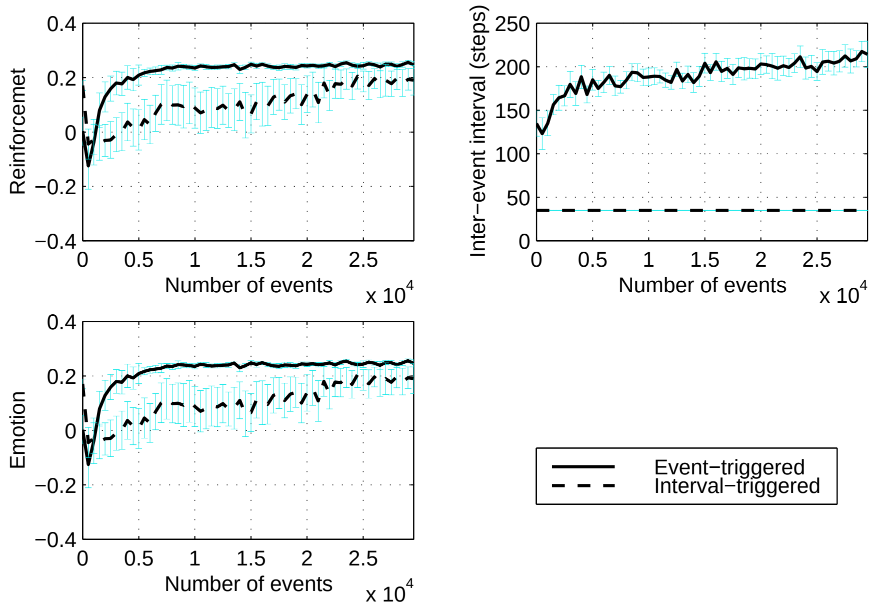

Figure 6: Comparing learning speeds in terms of events.

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 6 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 experiments 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 2, and slightly different values for the various trials of the event-triggered controller.

Controller	Total in millions	Relative to normal
Event-triggered	5.69 ± 0.08	190%
Interval-triggered	1.05 ± 0.00	35%

Table 2: Duration of trials in steps.

The graphs show that although the event-triggered controller has learned its task after one sixth 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.

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 interrupted as often as before.

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. It 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.

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

- The normal environment was replaced by the more demanding environment pictured in figure 7. 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

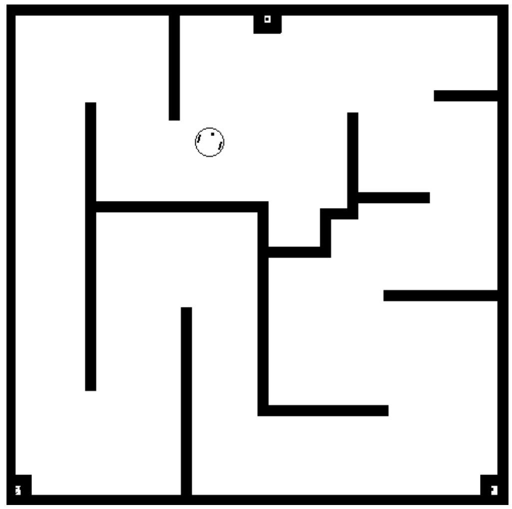

Figure 7: The robot in its more demanding environment.

energetic needs were increased. Furthermore, the advantage of not moving was removed by making the value of energy decrease independent of motor usage.

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 8 demonstrate the differences between the two controllers in this context. In this case, the advantages of the event-triggered controller are more evident.

Discussion

In this document a emotion model was proposed and integrated in a reinforcement learning architecture. The system was implemented and tested in a realistic robot simulator. Experiments showed that emotions can be used both as reinforcement and event detector in a reinforcement learning controller architecture based on behaviours. Furthermore, the emotion-dependent event detector allows drastic cuts in the number of triggerings of the learning controller which 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. The event-triggered controller learns its task with much less learning iterations and needs much less control iterations for successful robot performance.

It was empirically established 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. However, it is less flexible. The fact that intervals

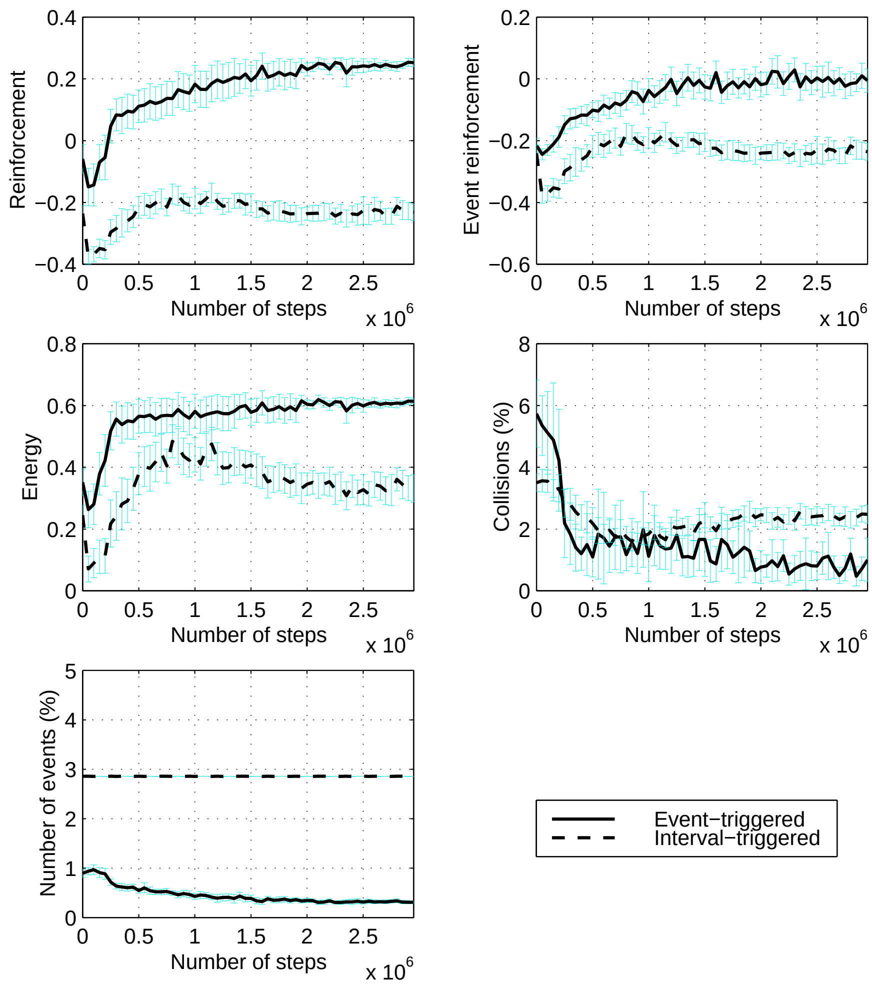

Figure 8: Comparing the different triggering mechanisms in the more demanding environment and with harder energy requirements.

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 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 a more flexible learner, *i.e.* it is 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, but 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 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 in 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 might even be beneficial (Gadanho, 1999).

Although not much importance is usually given to the problem of control triggering in the context of reinforcement learning and people usually resort to domain-specific solutions that artificially constrain the learning algorithm, experiments showed that the performance of the robot was very sensitive to the definition of the control triggering mechanism. Unlike other work in the field, the detection of changes in the input state proposed was dependent on the robot's dominant emotion and therefore intrinsically related with its reinforcement. The presented event detection mechanism 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.

The development of a emotion system for integration in a reinforcement-learning agent served as inspiration for the current research leading to the development of an innovative event-detection mechanisms which enhanced its learning capabilities.

References

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.

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.*

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.

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.

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.

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.

Charland, L. C. (1995). Emotion as a natural kind: towards a computational foundation for emotion theory. *Philosophical psychology*, 8(1):59–84.

Damásio, A. R. (1994). *Descartes' error — Emotion, reason and human brain.* Picador.

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.

Gadanho, S. C. (1999). *Reinforcement Learning in Autonomous Robots: An Empirical Investigation of the Role of Emotions*. PhD thesis, University of Edinburgh.

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.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Lazarus, R. S. (1982). Thoughts on the relations between emotion and cognition. *American Psychologist*, 37:1019–1024.

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 Melon University. Technical report CMU-CS-93-103.

Mahadevan, S. and Connell, J. (1992). Automatic programming of behavior-based robots using reinforcement learning. *Artificial intelligence*, 55:311–365.

Mataric, M. J. (1994). Reward functions for accelerated learning. In *Proc. of Conference on Machine Learning – 1994*, pages 181–189.

Michel, O. (1996). Khepera simulator version 2.0. 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.

Oatley, K. (1987). Editorial: Cognitive science and the understanding of emotions. *Cognition and Emotion*, 1(3):209–216.

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.

Power, M. and Dalgleish, T. (1997). *Cognition and Emotion*. Psychology Press.

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. Poster ?

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., 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.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning*. The MIT Press.

Toda, M. (1994). The urge theory of emotion and cognition: Chapter 2 Basic structure of the urge operations. SCCS technical report 94-1-01, Chuyko University.

Tomkins, S. S. (1984). Affect theory. In Scherer, K. R. and Ekman, P., editors, *Approaches to Emotion*. Lawrence Erlbaum, London.

Velásquez, J. D. (1998). A computational framework for emotion-based control. In *SAB'98 Workshop on Grounding Emotions in Adaptive Systems*.

Watkins, C. (1989). *Learning from delayed rewards*. PhD thesis, King's College.

Wilson, S. W. (1991a). The animat path to AI. In *From animals to animats*. The MIT Press.

Wilson, S. W. (1991b). The animat path to AI. In *From animals to animats — Proceedings of the First Conference on Simulation of Adaptive Behavior*. The MIT Press.

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.

Zajonc, R. B. (1984). On primacy of affect. In Scherer, K. R. and Ekman, P., editors, *Approaches to Emotion*. Lawrence Erlbaum, London.

Appendix: Emotion model

First, let us define the set of emotions ($\mathcal{E}$) and the set of feelings ($\mathcal{F}$),

$$\begin{aligned}\mathcal{E} &= \{\text{Happiness, Sadness, Fear, Anger}\} && (1)\\ \mathcal{F} &= \{\text{Hunger, Pain, Restlessness, Temperature, Eating, } \ldots\} && (2)\end{aligned}$$

and define the function $\mathrm{Th}_{[b_-,b_+]}(x)$ that bounds the value of x to the interval $[b_-, b_+]$.

$$\mathrm{Th}_{[b_-,b_+]}(x) = \begin{cases} b_- & \text{if } x < b_- \\ b_+ & \text{if } x > b_+ \\ x & \text{otherwise} \end{cases} \tag{3}$$

Taking this into consideration, the intensity value of emotion e at step n (I_{e_n}), depends on the intensity of the feelings in the following way:

$$\forall e \in \mathcal{E}, \forall n \in \mathbb{N}, \qquad I_{e_n} = \mathrm{Th}_{[0,1]}(B_e + \sum_{f \in \mathcal{F}} (C_{ef} I_{f_n})) \tag{4}$$

where B_e is a bias and C_{ef} is a coupling coefficient between the emotion e and the feeling f which are parameters of the system. To calculate the intensity of the feeling f at step n (I_{f_n}) the following functions are used.

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

$$I_{f_n} = \mathrm{Th}_{[0,1]}(C_h H_{f_n} + S_{f_n}) \tag{5}$$

$$H_{f_n} = \begin{cases} 0 & \text{if } n = 1 \\ \alpha_n H_{f_n} + (1 - \alpha_n) A_{f_n} & \text{if } n > 1 \end{cases} \tag{6}$$

$$A_{f_n} = \sum_{e \in \mathcal{E}:\ I_{e_n} > I_{th_a}} C_{ef} I_{e_n} \tag{7}$$

$$\alpha_n = \begin{cases} \alpha_{up} & \text{if } |A_{f_n}| > |H_{f_n}| \\ \alpha_{dn} & \text{otherwise} \end{cases} \tag{8}$$

The feeling's intensity calculation has to take into account both the influences provided by the hormone system (H_{f_n}), which are dependent of 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. Emotions only influence the hormone values if their intensity is above an 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}). 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.

If there are any emotions whose intensity is higher than a selection threshold (I_{th_s})

$$\exists e \in \mathcal{E} :\ I_e \geq I_{th_s} \tag{9}$$

then the emotion with the highest value is selected to be the dominant emotion.