Emotion-triggered Learning for Autonomous Robots

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.

Experimental work was done in a simulated robot that adapts to its environment through the use of reinforcement learning. Results suggest that emotions can be useful in detecting the relevant events of a learning task.

Introduction

In robotics, emotions are often used to modulate activity in a fixed controller (Cañamero 1997; Bates, Loyall, & Reilly 1992). 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, Ohkawa, & Tanie 1996)) or makes the robot capable of generating empathy emotions in people by creating an illusion of life in a believable character (Bates 1994).

The present research focus on how to use emotions in the control of an autonomous robot that adapts to its environment using reinforcement learning. Emotions seem to have solved in mammals many cognitive problems involved in autonomous systems with multiple goals and uncertain environments (Oatley 1987). It has been argued before in the field of Artificial Intelligence that emotions are the ultimate source of intelligence and might provide robots with the autonomy they need (Toda 1994).

The work was done under an animat philosophy (Wilson 1991), by building a biologically inspired complete agent where emotions form an integral part of the whole. The emotions are modelled by a recurrent network which incorporates the important computational features of Damásio's somatic marker hypothesis (Damásio 1994). Yet, the developed model is based on a simplified hormone system and is far from the complexity of true emotions experienced by humans. Experiments were carried out on a simulated Khepera robot in an animal-like adaptation task.

Emotions were used in the reinforcement learning framework to provide a reinforcement function and an event detection mechanism. The emotion-driven event detection mechanism was experimentally compared with other approaches and proved to be helpful for the robot's task.

In the next section, a description of the emotion model developed is presented. The experiments done with this model are reported in the following section.

Emotions

Contrary to prior assumptions, recent neurophysiological research suggests that emotions are helpful in human thinking. 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 (Damásio 1994).

Damásio makes a clear distinction between the concepts of emotion and feeling that will be used in the current work. 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.

Somatic markers are special instances of body feelings (visceral and non-visceral sensations) generated by emotions, that 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. This way, 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.

Inspired by the ideas that have been presented, an emotion model was developed that is described next.

Model

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 & Hallam 1998).

The model determines the intensity of each emotion based on the robot's current internal feelings. These feelings are: Hunger, Pain, Restlessness, Temperature, Eating, Smell, Warmth and Proximity.

Each emotion is defined by a set of constant feeling-dependencies and a bias. The values of the dependencies were carefully chosen to provide adequate emotions for the possible body states. For example, the sadness intensity will be high if hunger and restlessness are high and the robot is not eating.

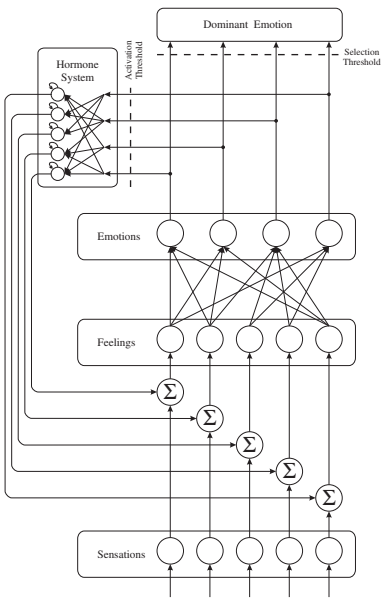

Figure 1: Emotions model.

Based on what was suggested in (Damásio 1994), the emotion state should also influence the way the robot feels. In general, the body reactions aroused by emotions also give rise to the emotions that create them. In this model 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 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, 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 provides not only an emotional state, based on simple feelings, that is coherent with the current situation, but also influences the body perception. In order to evaluate the role of emotions in reasoning, this state should be used for the actual control of the robot, determining its behaviour (Albus 1990; Wright 1996; Moffat, Frijda, & Phaf. 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.

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 getting energy at the moment.
- If the robot has very low energy and it is not acquiring energy, then it's state will be **sad**.
- If the robot bumps into the walls then the pain will make it **fearful**.
- 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. If the action is changed the value of restlessness is reset by the system.

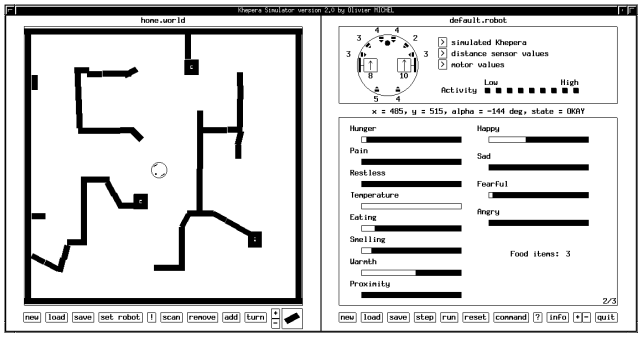

Figure 2: The robot and its environment.

All the experiments were carried out in a Khepera simulated robot (Michel 1996) within a closed environment divided by several walls and containing a few lights (figure 2).

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. Yet, 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 & Connell 1992; Lin 1993), behaviour coordination is much more difficult and is usually hard-wired to some extent (Mahadevan & 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. Yet, 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 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. Yet, 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 previously received evaluations.

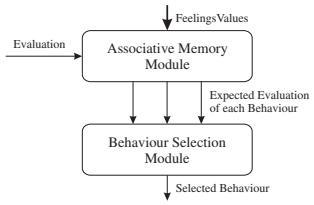

Figure 3: Adaptive controller.

The controller has two separate modules:

Associative Memory Module — This plastic module uses feed-forward networks to associate the robot feelings with the current expected value 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 (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 $u = R + \gamma \mathrm{Max}\{util(y,k) \mid k \in actions\}$ will be 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 $P(a) = \frac{e^{\frac{v_a}{T}}}{\sum_{i=1}^{3} e^{\frac{v_i}{T}}}$

The Role of Emotions

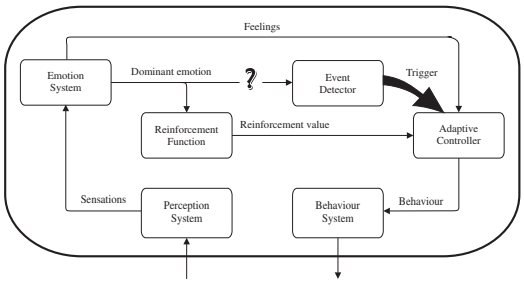

Figure 4: Emotions and control.

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 (positive intensity) value is negated. Although this reward function failed for a simple step by step action selection controller (Gadanho & 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 state can be found for 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[1]. 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.

The purpose of experiments reported in this paper is to explore *whether emotions can successfully fulfill the role of determining state transition.*

Testing the Hypothesis

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;
- 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 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 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.

In order to access the performance of the **Event Driven** controller, three other controllers were designed:

Regular Intervals — 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. 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. 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 designing a controller by hand 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

[1] In experiments made with the setup described previously, if a controller is triggered at every step it will converge to performance identical to that exhibited by random behaviour selection.

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.

This 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 Selection — 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.

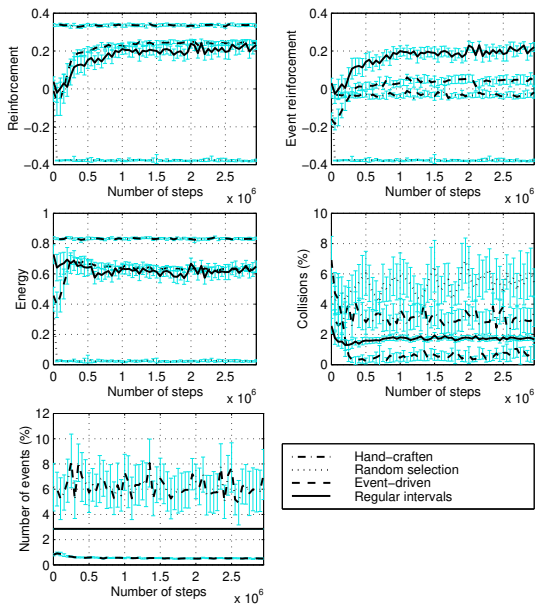

Figure 5: Value of the statistics recorded for each one of the controllers throughout the experiments.

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.

Controller	Reinf.	Event Reinf.	Energy	Collisions (%)	Events (%)
Hand-crafted	0.34	−0.03	0.83	3.0	6.15
Event-driven	0.24	0.04	0.63	0.6	0.52
Regular intervals	0.21	0.20	0.62	1.7	2.86
Random selection	−0.38	−0.38	0.02	5.6	100

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

Discussion

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 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 much of a problem, as long as the controllers 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 of-

ten 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. 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 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-driven 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 to deal with them. The difference in event reinforcement does not tell much apart from the fact that the event-driven controllers are often triggered when something goes wrong. On the one hand, the event reinforcements of the regular intervals controller is very similar to its overall reinforcement, because the event reinforcement are picked from the rest 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 were that the current behaviour had to be changed, because it was not adequate anymore.

The event-driven controller does not perform outstandingly better than is 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.

Conclusions

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 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.

Acknowledgements

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.

References

Albus, J. S. 1990. The role of world modeling and value judgement in perception. In Meystel, A.; Herath, J.; and Gray, S., eds., *Proceedings of the* 5^{th} *IEEE International Symposium on Intelligent Control*. Los Alamitos, California: IEEE Computer Society Press.

Bates, J.; Loyall, A. B.; and Reilly, W. S. 1992. 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. 1994. The role of emotions in believable agents. Technical Report CMU-CS-94-136, Carnegie Mellon University, School of Computer Science.

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.

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

Gadanho, S. C., and Hallam, J. 1998. Emotion-driven learning for animat control. In *From animals to animats 5*. The MIT Press. Extended version available as DAI Research Paper 881, University of Edinburgh.

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*. Ph.D. Dissertation, Carnige Melon University. Tecnical 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*, 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`.

Moffat, D.; Frijda, N.; and Phaf., R. 1993. Analysis of a model of emotions. In et al., A. S., ed., *Prospects for Artificial Intelligence*. IOS Press. 219–228.

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

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*.

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., eds., *Approaches to emotion*. London: Lawrence Erlbaum.

Watkins, C. 1989. *Learning from delayed rewards*. Ph.D. Dissertation, King's College.

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

Wright, I. 1996. Reinforcement learning and animat emotions. In *From animals to animats 4*. The MIT Press.