Increasing Behavioural Repertoire in a Mobile Robot*

Ulrich Nehmzow*, Tim Smithers‡ and Brendan McGonigle*

*Laboratory for Cognitive Neuroscience at the Department of Psychology
University of Edinburgh
Appleton Tower
GB - Edinburgh EH8 9LE
Telephone +4431 - 650 8449 Fax +4431 - 650 6512
E-mail:ulrich@castle.ed.ac.uk, B.O.McGonigle@ed.ac.uk

‡Artificial Intelligence Laboratory
Vrije Universiteit Brussel
Pleinlaan 2
B - 1050 Brussel
Tel +322 - 641 2965 Fax +322 - 641 3582
E-mail:tim@arti1.vub.ac.be

Abstract

This paper presents an investigation of the the suitability of the robot controller presented in [Nehmzow *et al.* 89] and [Nehmzow *et al.* 90] for a computationally cheap expansion of the behavioural repertoire of a mobile robot.

Experiments with mobile robots are presented that show that this is possible by simply adding further so-called *instinct-rules* without altering the controller itself: through the robot's interaction with its environment effective associations between sensors and actuators arise in an artificial neural network which serves as an associative memory.

1 Introduction

Designing intelligent controllers for autonomous mobile robots is a task often underestimated by the designer. Sensor signals turn out to differ from what is expected in theory, and actuators produce different effects than anticipated. Whilst many of these differences between expected and observed behaviour can be overcome by building actual robots, there remain uncertainties and variations that are due to temperature conditions, wear, battery charge etc., as well as more drastic changes in agent (failure of components) task or environment (changed external conditions). To deal with these through *a priori* defined strategies is (in practice) impossible.

One possible solution to the problem of coping with such unforeseen situations is the use of self-organising controllers that determine the effective wiring between sensors and actuators autonomously. Such an approach is described in [Nehmzow 92], for example. An additional property of the proposed architecture is discussed in *this* paper: that it allows an easy expansion of the robot's behavioural repertoire by adding so-called *instinct-rules*, without necessitating changes to the controller architecture itself. This increases the robot's flexibility to accomplish new tasks.

1.1 Related Work

Some work has been done on the acquisition of single competences in autonomous mobile robots, for example by [Mahadevan & Connell 91], [Maes & Brooks 90] and [Kaelbling 91]. Mahadevan and Connell and Kaelbling use reinforcement learning to control robots acquiring the competences of box pushing and phototaxis (moving towards a light source) respectively. Maes and Brooks use the correlation between a particular robot behaviour and positive reinforcement to establish 'relevant' behaviours; the likelihood of a behaviour being invoked then is proportional to its relevance. All of these architectures are designed particularly to facilitate the acquisition of *one* particular motor-sensory skill

*©U. Nehmzow, T. Smithers and B. McGonigle, Ocotber 1992 — Reprint from J.A. Meyer, H. Roitblat and S. Wilson (eds.), *From Animals to Animats 2*, pp.291-297, MIT Press 1993

only (box-pushing, moving towards a light, and walking, respectively). Contrary to this, the experiments described here aim to accomplish the acquisition of *various* motor-sensory skills, using the same controller architecture throughout.[1]

Addressing the question of automatic acquisition of behaviours, Brooks discusses first results of John Koza's genetic programming implementation ([Koza 90]) for the determination of computer programs. Here, LISP programs are determined by a genetic algorithm. This approach is promising, however the resulting programs are not yet complex enough to actually control robots (see [Brooks 91]).

2 The Controller Architecture

For all experiments described in this paper the same controller architecture was used. New behaviours are acquired by the robot by autonomously determining the effective wiring between sensory input and motor actions in a trial and error process.

2 bits	2 bits	2 bits	1 Bit	1 Bit
Left and Right Whisker	Left and Right Whisker	Left and Right Whisker	Forward Motion Sensor	On only if all other bits are off

"1" = Sensor on, "0" = Sensor off

Figure 1: The input vector.

The experiments presented here were conducted with the mobile robots Alder (shown in figure 2) and Cairngorm (shown in figure 8). Both are completely autonomous, Alder is controlled by an on-board ARC 52 controller (based on the Intel 8052 microprocessor) and uses two tactile sensors (whiskers) mounted at the front for the experiments discussed below, as well as a forward motion sensor. Controller card and whiskers are clearly visible in the photograph. Cairngorm is controlled by a Flight 68 k controller card, based on the Motorola 68000 processor; like Alder it is equipped with tactile sensors.

The controller architecture used in the experiments reported here is shown in figure 3. Its central element is a connectionist associative memory (Perceptron-like) which stores the effective associations between input signals and motor actions.

The performance of the robot is assessed through *instinct-rules*, which are fixed, predefined rules such as "Keep whiskers straight"[2]. Each instinct-rule is monitored by a dedicated sensor. The control mechanism is as follows[3]: as soon as the violation of an instinct-rule is detected by the monitor[4], an input vector (see figure 1) is generated and presented to the associative memory.

In this associative memory—a Perceptron-like, two-layer network—associations between the input signals and the four output units of the network are stored. These output units of the artificial neural network stand for four possible motor actions of the robot (left and right turn, forward and backward movement). The motor action associated with the output node responding most strongly to a particular input vector is then performed for about three seconds[5]. If, within this period, the violated instinct-rule is resolved again, the association between the original input vector and the respective motor action is confirmed to the network, if not, the action associated with the second strongest output node[6] is performed for a slightly longer period of time (the increment is two seconds). This process continues until appropriate responses by the robot are found and stored in the associative memory: effective associations between input signals and motor responses thus arise through the interaction of the robot with its environment.

[1] Investigations into reinforcement learning architectures that might be used for robot control are presented by many researchers, for example by [Kaelbling 90, Sutton 91, Prescott & Mayhew], but these have not been conducted using real robots.

[2] Strictly speaking, the robot only monitors the signals coming from the whiskers, the whiskers being off when they are straight and on when they are bent. There is no notion of a bent whisker being identical with an obstacle, for instance.

[3] For more details see [Nehmzow 92].

[4] If more than one instinct-rule is used instinct-rules are tested for violation beginning at the latest (the newest) and ending at the first instinct-rule.

[5] This value is mostly dependent on the velocity of the robot.

[6] At the beginning of the learning phase, when no associations are stored yet, motor actions are tried in turn, beginning with the first output node of the network and ending with the fourth.

Figure 2: Alder.

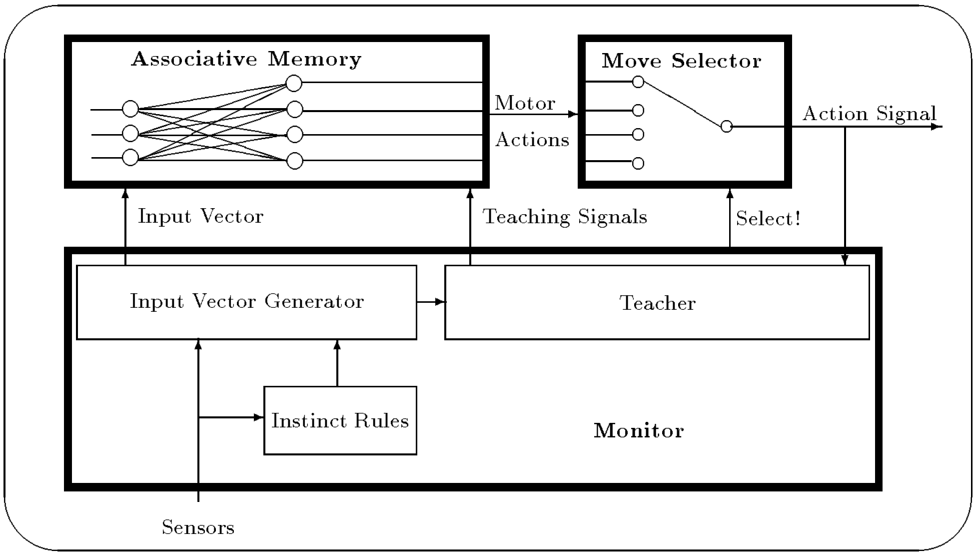

Figure 3: The controller architecture.

3 Experimental Results

3.1 Learning to Move Forward

Using the controller architecture presented in the previous section and one instinct-rule, the one shown in figure 4, Alder is able to determine the motor action that will make its forward motion sensor turn on.

1. Keep forward motion sensor on

Figure 4: The instinct-rule for learning forward motion.

A schematic diagram of the forward motion sensor of Alder is shown in figure 5. It is a pushbutton switch, operated by a cam that is attached to the caster wheel of the robot. If the caster wheel is aligned with the central axis of the robot (this happens after the robot has been moving forward for at least three seconds) the pushbutton switch is pressed, thus indicating that the robot is performing a forward motion.

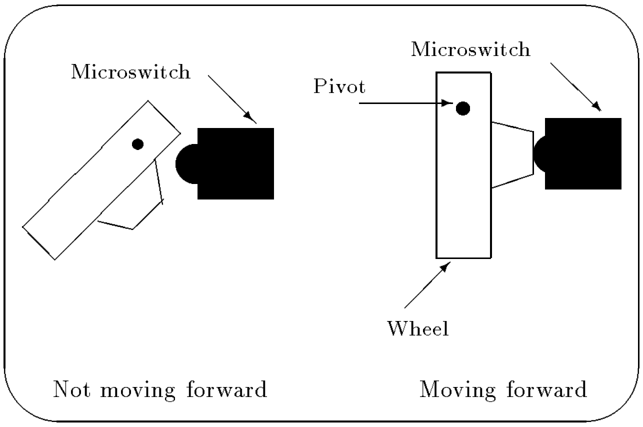

Figure 5: The forward motion sensor of Alder.

Independently from the initial wiring of the motors, Alder learns the "correct" output node of the network (the one that results in forward motion) after having found the correct node for the first time (one learning step). This takes between five and twenty seconds, dependent on how many other motor actions have to be tried until the forward motion node is found.

3.2 Additional Learning of Obstacle Avoidance Behaviour

The previous experiment resulted in a robot determining the effective wiring between its motors and its forward motion sensor in order to move forward, independently from the initial wiring of the motors.

It is possible to increase the behavioural repertoire by simply adding a further instinct-rule to the already existing one, without altering anything else in the controller architecture. Figure 6 shows the set of two instinct-rules that lets Alder learn how to move forward, and how to move away from obstacles.

1. Keep forward motion sensor on
2. Keep whiskers straight

Figure 6: The instinct-rules for learning forward motion and obstacle avoidance.

When started, the robot determines the correct motor action to move forward, as before. It will then start moving forward. As soon as any of the two whiskers emits a signal, instinct-rule 2 is violated and Alder will initiate motor actions to stop the whiskers signalling (that this coincides with the robot moving away from obstacles is not represented inside the controller; no symbolic representation of the world is used). Learning the wiring that accomplishes both forward motion and appropriate turn actions when an obstacle is encountered is achieved in approximately ten learning steps, requiring one to two minutes, dependent on the amount of perturbance from bouncing or caught whiskers, i.e., the consistency of sensor readings.

3.3 Additional Learning of Wall Following Behaviour

Again without changing anything but the set of instinct-rules, Alder is able to learn also to follow walls, as well as to maintain the two previously mentioned skills. The set of instinct-rules used for this experiment is shown in figure 7.

When Alder is started, it acquires the forward movement competence as before. After this, if a whisker gives a signal (i.e., touches something) Alder learns which way to turn in order to release the whisker. If no whisker signal is received within four seconds, the third instinct-rule initiates a wall-seeking movement. Provided the robot is near a wall it will eventually find it and thus learn which way to turn in order to satisfy the third

1. Keep forward motion sensor on
2. Keep whiskers straight
3. Make whiskers signal after 4 seconds

Figure 7: Instinct-rules for learning forward motion, obstacle avoidance and wall following.

instinct-rule. As soon as the wall is touched, of course, instinct-rule number 2 is violated and the robot will turn away again. The resulting behaviour is a movement along a wall, performing zig-zag motions towards and away from the wall every four seconds (see figure 8).

Figure 8: Cairngorm, learning to follow a wall.

To acquire all three skills takes about ten to fifteen learning steps (which take one to two minutes), however it can take longer if, in the course of acquiring the obstacle-avoidance competence the robot moves so far away from the wall that instinct-rule number 3 can no longer be satisfied straight away. This occasionally happens.

3.4 Additional Learning of Corridor Following Behaviour

Figure 9 shows the set of instinct-rules that make the robot learn how to follow a corridor, touching alternate sides of the corridor as it goes along. Again, the only component changed in this experiment, compared to the previous ones, is the fourth instinct-rule.

1. Keep forward motion sensor on
2. Keep whiskers straight
3. Make whiskers signal after 4 seconds
4. Make *alternate* whiskers signal

Figure 9: Instinct-rules for learning forward motion, obstacle avoidance and corridor following.

To acquire the corridor following competence (as well as the three previously discussed competences) requires approximately five minutes. It turns out that the robot is more dependent on reliable sensor signals in this experiment than in the previous ones. The forward motion sensor, for example, only signals if the robot *has been moving forward* for at least three seconds, because of the way it is built. Therefore, for example, pushing against one wall of the corridor—which can easily happen in the early learning phases—will result in the forward motion sensor not signalling, even if the robot is issuing the correct motor commands. This of course will lead to some 'confusion' which can (and often does) affect the rate of learning. Eventually, however, the correct corridor following behaviour *is* acquired, even under adverse conditions.

4 Discussion

4.1 How to devise Instinct-Rules

The expansion of Alder's behavioural repertoire was achieved through adding instinct-rules. How these can be determined is an interesting question, especially because it is conceivable that the determination of instinct-rules could be automated. An algorithm for finding instinct-rules has not been established yet, however some general guidelines can be identified:

- The violation of instinct-rules is detected by sensors dedicated to a particular instinct-rule. Sensors in this sense can be physical sensors (e.g., whiskers), internal sensors (e.g., timer), or values in memory locations ("Make alternate whiskers signal").
- Instinct-rules are not behaviours, but sensory conditions that have to be maintained. It would therefore be wrong to have an instinct-rule "Move towards the light", the correct rule would be "Increase the reading from a light sensor", or "Keep the light sensor on".
- As instinct-rules generate the desired behaviour, one question to be asked is "Which sensor signal can be associated with the desired behaviour?". The answer to this question is a guideline for deriving instinct-rules.
- It should be borne in mind that instinct-rules are checked for violation beginning at the newest rule and ending at the first instinct-rule. This ordering therefor defines priorities between instinct-rules.

4.2 Advantages

A number of properties make the self-organising controller presented in this paper interesting for robotics applications.

- The algorithm learns fast: effective connections between sensors and actuators are established within a few tens of learning steps, rather than several hundred or thousand as can be found in controllers using other kinds of reinforcement learning, see ([Prescott & Mayhew, Kaelbling 90, Sutton 91], for example. This is an important property, not so much because the operational period of a robot is short (batteries), but more because certain functions (like obstacle avoidance) *have* to be learned quickly in order to keep the robot operational at all.
- Because the effective wiring between sensors and actuators is determined by the robot, not the designer, setting up of a robot becomes less prone to error: sensors and actuators may initially be connected arbitrarily.
- Whereas acquired knowledge in reinforcement learning is often compartmentalised ([Kaelbling 91]), i.e., the robot is not able to generalise from experience, a benefitial side effect of using a connectionist associative memory is that such a generalisation does take place. For example will the robot turn left if *both* whiskers are on, even if it has only ever experienced the *right* whisker to be on.
- If the learning process has to be guided by the experimenter in the early stages, as is the case for some implementations ([Kaelbling 90]), this can diminish the robot's ability to cope with new situations. Here such initial guiding helps, but is not required.
- Because the controller is self-organising, it can not only determine the effective connections between sensors and actuators in the first place, it can also re-establish them in the event of unforeseen situations (swapped whiskers, swapped motor connections, change of environment etc., see also [Nehmzow 92]).

4.3 Comparision to Biological Systems

The system described thus far is certainly based on the designer's interpretation of a given niche within which the robot is supposed to work (office or laboratory environment with smooth, level floors, containing walls and box-like obstacles). This limitation is, we believe, an essential one: a general purpose robot does not exist.

In biological systems such behaviours as are described in this paper, based on taxes and tropisms, and selected over many generations, work robustly within the confines of well specified contexts. *Salmon* fry, for example, are positively phototropic when young and stay in bright (i.e., shallow) water, later they become negatively phototropic and seek dark (i.e., deep) water (the sea). As in the robot described here, the actual behaviour need not be specified in advance: behaviour selection can readily occur, ensuring the successful maintenance of the tropistic values set (the instinct-rules). Furthermore, in both cases the knowledge of the world is analogous and indirect: just as the wall is a whisker signal to the robot, so is the water route towards the sea associated with decreasing light intensity. In the salmon, a biological clock determines when the time is right to change the response to light and head for darker waters, which means that the overall behaviour is purposeful without requiring the agent to acquire any long term plan or represent a specific objective. One last similarity to the robot described is that the details of the behaviour are not specified: the agent can select those actions which best support the tropism.

Selecting specific behaviours from the total response set (as described here) is an example of procedural learning, an 'old' competence from an evolutionary perspective. Such competences can be dissociated both in phylogeny and in human clinical conditions (such as amnesia) from episodic, semantic and explicit forms of knowledge. There is no evidence, moreover, that declarative types of knowledge based on episodic or semantic forms of memory derive from a procedural motor (sub) system. On the contrary, as [Sherry & Shacter 87] have argued, separate multiple memory systems have evolved to cater for different types of achievement. It would be somewhat unwise, therefore, to propose a motor (based) learning system to support a general purpose kind of knowledge acquisition. The system presented here is designed to achieve a specific action-related 'representation' of an environment as in 'when to go', 'when to stop', 'when to turn' etc. Specifying *descriptive/declarative* systems in ways which yield actual robot implementations *remains* one of the major problem areas. To do this, we shall need complex biological agents as a model, for it is only in these that such competences are well developed — if they exist at all in simpler systems remains in doubt.

5 Conclusions

We have presented a controller architecture for mobile robots, using plastic components (connectionist associative memory) and fixed components (instinct-rules) that allows an easy expansion of the behavioural repertoire of a robot by merely adding further instinct-rules, leaving the controller itself unchanged.

Experiments with the mobile robots Alder (see figure 2) and Cairngorm (see figure 8) are presented: the robots autonomously determine the effective wiring between their sensors and actuators to achieve forward movement, obstacle avoidance, wall following and corridor following.

Besides offering an easy expansion of the behavioural repertoire of a mobile robot, as described in this paper, the architecture presented gives a higher degree of flexibility in unforeseen circumstances: the robot becomes able to cope with changes in its own morphology, changes of the task and changes in the environment (see [Nehmzow 92]).

Acknowledgements

The work reported here was supported by grant GR/F/5852.3 from the Science and Engineering Research Council. Other facilities and technical support were provided by the Department of Artificial Intelligence at Edinburgh University. John Hallam and Peter Forster were also involved in the funded project and we acknowledge their contribution. We thank Douglas Howie for producing the photographs of Alder and Cairngorm. Tim Smithers currently holds a visiting chair at the VUB AI Lab, Brussels.

References

[Brooks 91] Rodney Brooks, *Artificial Life and Real Robots*, Proceedings of 1st European Conference on Artificial Life, Paris 1991, published by MIT Press 1992

[Kaelbling 90] Leslie Kaelbling, Learning in Embedded Systems, PhD Thesis, Stanford Technical Report TR-90-04, 1990

[Kaelbling 91] Leslie Kaelbling, *An Adaptable Mobile Robot*, Proceedings of 1st European Conference on Artificial Life, Paris 1991, published by MIT Press 1992

[Koza 90] John Koza, *Evolution and Co-Evolution of Computer Programs to Control Independently Acting Agents*, From Animals to Animats, MIT Press 1991

[Maes & Brooks 90] Pattie Maes and Rodney Brooks, *Learning to Coordinate Behaviours*, National Conference on Artificial Intelligence 1990

[Mahadevan & Connell 91] Sridhar Mahadevan and Jonathan Connell, *Automatic Programming of Behavior-based Robots using Reinforcement Learning*, 9th National Conference on Artificial Intelligence 1991

[Nehmzow *et al.* 89] UIrich Nehmzow, John Hallam and Tim Smithers, *Really Useful Robots*, in T. Kanade, F.C.A. Groen and L.O. Hertzberger (eds.), *Intelligent Autonomous Systems 2*, Amsterdam 1989

[Nehmzow *et al.* 90] Ulrich Nehmzow, Tim Smithers and John Hallam, *Steps towards Intelligent Robots*, DAI Research Paper No. 502, Department of Artificial Intelligence, Edinburgh University, 1990

[Nehmzow 92] Ulrich Nehmzow, *Experiments in Competence Acquisition for Autonomous Mobile Robots*, PhD Thesis, University of Edinburgh 1992

[Prescott & Mayhew] Tony Prescott and John Mayhew, *Obstacle Avoidance through Reinforcement Learning*, to appear in J.E. Moody, S.J. Hanson and R.P. Lippmann (eds.), *Advances in Neural Information Processing Systems 4*, Sam Mateo, Morgan Kaufman, no year

[Sherry & Shacter 87] D.F. Sherry and D.L. Shacter, *The Development of Multiple Memory Systems*, Psychological Review, Vol 94, pp. 439-454, 1987

[Sutton 91] Richard Sutton, *Reinforcement Learning Architectures for Animats*, From Animals to Animats, MIT Press 1991