Exploring Internal Simulation Of Perception In Mobile Robots

Exploring Internal Simulation of Perception in a Mobile Robot

Masters' Thesis in Cognitive Science, HKGD17
Department of Computer Science
University of Linköping

LIU-KOGVET-D-0047-SE

Written by Dan-Anders Jirenhed
danji745@student.liu.se

Supervised by Tom Ziemke
Department of Computer Science
University of Skövde
tom@ida.his.se

April 2001

Summary

This thesis is concerned with exploring the possibility of an ‘inner world’ based on internal simulation of perception. Introspective and behavioral evidence suggest the existence of an inner world allowing for sensory experiences in absence of external stimuli. Traditional accounts claim that this inner world is realized by possessing an internal representational model of the external world. An alternative, non-representational account, suggests that the phenomenon may instead be realized by internal simulation of perception through associative activation of neural sensory structures. An empirical investigation was conducted in three sets of experiments using a simulated robot controlled by a recurrent artificial neural network. The design of the controller network was based on minimal assumptions about the necessary internal mechanisms. Using an evolutionary algorithm the robots were trained on increasingly complex tasks. In the first experiment, serving as a baseline, robots were simply trained to map sensory input to motor output such that they moved around in an environment without collisions. In the second experiment robots were additionally trained on predicting the next time step’s sensory input. In the third experiment, finally, the robot’s own prediction replaced the actual sensory input in order to investigate its capability to act ‘blindly’, i.e. in the temporary absence of external stimuli. The experimental results showed that behavioral abilities were acquired successfully, but learning to predict the sensory consequences of behavior proved to be a more complex problem than was initially perceived, thus affecting internal simulation abilities. Although only the first two experiments gave positive results, the conclusion is drawn that the experimental framework presented here should turn out useful in the investigation of more complex artificial neural models.

Acknowledgements

First, I would like to thank Tom Ziemke for being an excellent supervisor, for having the patience to answer my many questions and for continually providing ideas for improvement of this thesis.

Furthermore, I would like to thank Germund Hesslow for expressing the idea on which this whole thesis was based, for literature recommendations and for taking the time to clarify many details. I would also like to thank Arne Jönsson for all his help with making my interest in intelligent and adaptive behavior become a formal education in the end.

Finally, I would like to thank Mårten Szymanowski for providing many useful comments on the work presented in this thesis.

Contents

1. Introduction

Behavior and the causes of behavior have interested people for thousands of years, be it the behavior of humans or that of other animals. Numerous questions about the nature of the mechanisms underlying behavior have been raised and possibly even more accounts have been given. Numerous traditions of research – philosophy, psychology, ethology and neuroscience to name but a few – all offer a wide range of explanations. Each of these accounts in turn assumes a certain ontology – a set of explanatory entities. Two rivaling accounts for a common observed phenomenon may sometimes rely on very different ontologies and a choice between two explanations may thus become a choice of ontological commitments. Unfortunately, it is often far from obvious which entities are to be trusted and which are to be discarded.

The early 14th century philosopher William of Occam is perhaps best known for stressing the importance of making minimal assumptions in explanations. His principle of unnecessary plurality – "Don't multiply entities beyond necessity" – later came to be known as *Occam's razor* (Rey, 1997). The principle states that one should not introduce entities one may do without when explaining an observed phenomenon. All other things being equal, the theory necessitating the least assumptions tends to be the better one. A similar statement, specifically regarding explanations of behavioral observations, was later made by psychologist C. L. Morgan (1894):

> In no case may we interpret an action as the outcome of the exercise of a higher psychical faculty if it can be interpreted as the outcome of the exercise of one which stands lower in the psychological scale.

Similar to Occam, Morgan stressed the importance of not attributing the observed organism with any abilities for which there was no factual need. For example, if the moth's tendency to fly into the light could be explained in terms of something less extravagant than its *desire* to do so, then that was to be the preferred explanation. Morgan's work contributed to making comparative psychology an objective science which discarded previously common anecdotal, subjective and anthropomorphic accounts of behavior (Goodenough, McGuire and Wallace, 1993).

Experimental rigor and explanatory caution was also urged by behaviorists like J. B. Watson and B. F. Skinner. Both stressed the importance of dealing only with intersubjectively verifiable data, such as physical stimuli and behavioral responses. According to Skinner's philosophy of radical behaviorism, the observations of behavior and the environmental contingencies in which behavior occurred and changed were to be made and analyzed by psychologists, but the ultimate explanation of how internal mechanisms were involved in eliciting the observed behavior was to be given by physiologists. Internal mechanisms properly understood were physiological, not mental. Mental mechanisms could not be scientifically studied and thus, according to behaviorism, psychology was to be the scientific analysis of *behavior* – no more and no less. Mentalist concepts, such as ‘mind’ and ‘mental representation’ were dismissed as explanatory entities in accounts of behavior, since they were only possible to *infer from* behavior, verbal or other, and no other evidence in favor of their existence were available. Not much has changed since these arguments were raised the first time, except for the behavior of scientists. Cognitive psychology today abounds with accounts referring to mental mechanisms (Staddon, 1993; Skinner 1976, 1987).

A parallel to behaviorism may be found in the philosophical school of Eliminative Materialism (EM), most prominently represented by the works of Paul and Patricia Churchland (cf., e.g., Churchland 1986, 1989 or 1995). EM claims two things: First, the common sense conception of mental events and processes (or ‘folk psychology’) is a false and misleading account of the causes of behavior. Second, like other false conceptual frameworks from both folk theory and the history of science, it will be replaced by, rather than smoothly reduced or incorporated into, a future neuroscience. EM is physicalist in the classical sense, postulating some future brain science as the ultimately correct account of behavior. It is eliminative in the sense that it predicts the future removal of folk psychological entities, such as ‘will’ and ‘intention’ from our post-neuroscientific ontology. EM thus comes to the same conclusions as behaviorism about the appropriate framework for explaining the internal mechanisms underlying observed behavior.

Philosophical assumptions and preconceptions are always present in any scientific investigation. According to Inman Harvey (2000), this is true in a very determining way when conducting robotics research on behavior. In computational modeling, one’s philosophical position and preconceptions will strongly affect design decisions. All mechanisms put into the model will reflect both *explicit* and *implicit* assumptions made about which entities should be part of an account for the behavior being modeled. Those assumptions are then put to the test in an automated system in the real world. Hence, Harvey has appropriately called robotics research “philosophy of Mind using a screwdriver”.

1.1 An Inner World?

Introspective observation seems to tell us that we are able to have sensory experiences also in absence of external stimuli. This seems to be confirmed by experimental results of, e.g., Lee and Thompson (1982). In a series of experiments they demonstrated the accuracy with which humans can guide their behavior based solely on internally generated sensory experiences. A group of subjects were allowed to first look at their surrounding environment and direct specific attention to certain objects, such as marks on the floor and different obstacles. They were then asked to perform different tasks such as walking to the marked locations, avoiding the obstacles and throwing objects at different targets in the room. All tasks were performed with eyes closed. The subjects performed these tasks almost as accurately with eyes closed as when they were free to look. From these and other behavioral observations, it seems reasonable to propose the existence of an 'inner world' where sensory experiences can arise and where consequences of different behaviors may be anticipated. Some advantages of having an internal world were formulated in Kenneth Craik's (1943) *The Nature of Explanation*:

> If the organism carries a "small-scale model" of external reality and of its own possible actions within its head, it is able to try out various alternatives, conclude which is the best of them, react to future situations before they arise, utilize the knowledge of past events in dealing with the present and future, and in every way react in a much fuller, safer, and more competent manner to the emergencies which face it.

The probable existence of such an inner world has been argued from an evolutionary perspective by, e.g., Dennett (1978). Trial-and-error learning in the real world can be risky business for an organism and the consequences can be a matter of life or death. An ability to foresee lethal consequences should thus greatly improve the individual's chances of survival. From an evolutionary perspective then, the ability to foresee consequences of actions should clearly be advantageous. If such an ability were to appear at some stage in the evolution of organisms and if this ability, a physiological trait, would be genetically hereditary, then it would quite probably spread in the population over successive generations. Individuals with an ability to foresee the consequences of their behavior would (on average) better be able avoid dangers and survive longer and hence improve their chances of passing on their genes through reproduction. Conversely, individuals that were not able to foresee and avoid negative consequences would in comparison be less likely to pass on their genes. The net effect of evolution would thus be organisms with inner worlds. Note however that this does not say anything about the nature of the evolved inner world – what its exact properties are and how it is realized. It only supports a belief in its existence.

1.2 Motivation and Aim

The inner world certainly seems to be a very complex phenomenon, but do the assumptions in explaining the phenomenon need to be great? Skinner (1976) wrote:

> That a person may see things when there is nothing to be seen must have been a strong reason why the world of the mind was invented. It was hard enough to imagine how a copy of the current environment could get into the head where it could be “known”, but there was at least a world outside which might account for it. But pure images seem to indicate a pure mind stuff. It is only when we ask how either the world or a copy of the world is seen that we lose interest in copies. Seeing does not require a thing seen.

A hypothetical account for the evidence for an inner world, put forward by neurophysiologist Germund Hesslow (cf. Hesslow, 1996; Hesslow, 1994), argues that it should be understood in terms of *internal simulation* of perception and behavior. The account draws on neurophysiological observations of neural structures involved in perception and initiation of observable or *overt behavior*. It seems as if *covert behavior* and *imagery* is realized by neural processes very similar to those involved in observable behavior. In both cases, the very same neural substrates are often involved. Perhaps then, the inner world could be explained in terms of these neural structures, engaged in *internal simulation* of processes normally involved when interacting with the external world (cf. Section 2.5). This hypothesis of the inner world as internally simulated perception and behavior does not appeal to any assumptions of mind or mental entities, but instead relies on empirical data accountable for in purely materialistic terms. Although the full validity of the hypothesis is yet to be experimentally verified, it does draw on empirical evidence gathered from neuroscientific research and, perhaps more importantly for the current discussion, it is an attempt at explaining the phenomenon by making *minimal assumptions about the underlying mechanisms*. The hypothesis of the inner world as internally simulated perception and behavior (henceforth called *the simulation hypothesis*) is the main theoretical motivation for the experimental investigation presented in this thesis.

The specific aim of this thesis is to investigate to what extent a robot can develop an ability to internally simulate perception as sequences of sensorimotor interactions with its environment and thus realize some form of an artificial inner world. The empirical work reported in this thesis was a minor AI investigation of the simulation hypothesis, in the spirit of making minimal assumptions about the mechanisms underlying intelligent behavior. The internal mechanisms assumed to be necessary for control of behavior and the realization of an inner world were made explicit in the design of a recurrent artificial neural network controller for a mobile robot. In line with the simulation hypothesis, the same neural structures used in behavior control when interacting with the environment were used for the internal simulation of that interaction. No additional mechanisms were assumed for the realization of an inner world in the robot.

1.3 Overview

Chapter 2 will cover the theoretical and empirical background for the current work. An overview will be provided, starting with a brief presentation of the traditional cognitivist view on intelligence, the brain, inner worlds and how to model intelligence using computers, followed by a presentation of an alternative approach to intelligence and modeling of mechanisms underlying behavior. Chapter 3 will present the methodological and technical background of the empirical investigation made, as well as results obtained and observations made. Chapter 4 will provide a discussion of the experimental investigation and the obtained results and what they may say about an inner world and the method used to model the phenomenon.

2. Background

This chapter presents the central concepts and theories behind this thesis, concerning ‘inner worlds’ in natural and artificial systems. The traditional Cognitive Science view of a representing inner world and the concept of planning will be discussed in Sections 2.1 and 2.2, both as explanatory constructs for understanding the mechanisms underlying complex behavior and as a methodology for creating artificial systems. Problems with the assumptions underlying the traditional Cognitive Science explanatory framework and methodology (in relation to the issue of inner worlds) will also be discussed. An alternative, and more recent, approach to modeling intelligent behavior is discussed in Sections 2.3 and 2.4. Sections 2.5 and 2.6 then return to the central topic and discuss an alternative view of the inner world and a possible alternative to the traditional approach to endowing artificial agents with some kind of inner world.

2.1 The Representing Brain

The traditional Cognitivist view of mechanisms underlying intelligent behavior holds that within the neural structures of the brain there exists a *representational mental model* of external reality. All intelligent behavior presupposes this ability to internally represent the world as being certain ways. The inner world is seen as fundamental for intelligent behavior. The cognitive processes underlying intelligent behavior are seen as the rule-governed processing of mental representations in the inner world (cf. Varela, Thompson and Rosch, 1991). In perception the organism translates stimuli into internal representations. The representations are then manipulated by cognitive processes which derive new representations. These can in turn be translated back into actions (Russel & Norvig, 1995). As Skinner (1987) writes about the cognitivist perspective:

> For cognitive science the direction of action is from organism to environment. The perceiver acts upon the world, perceives it in the sense of taking it in. […] The world [has] to be taken in and possessed in order to be known.

The cognitive processes within the organism *control* behavior, i.e. the organism acts upon the world rather than interacts with it. The organism does not do this directly, but *via* its internal world model (cf. Varela *et al.*, 1991) or to quote Skinner’s (1987) description of this view: “we do not see the world, we see copies of it”. Obviously, this demands a very accurate inner world model if the organism is to control its behavior correctly and survive in reality. If one assumes a *representational relation* between the inner world model and the external world, as cognitivism does, then the proposed inner model *must* be accurate, since the assumed relation is very strong. In the words of one explicit representationalist, psychologist C. R. Gallistel (1990):

> […] I use the term [representation] in its mathematical sense. The brain is said to represent an aspect of the environment when there is a functioning *isomorphism* between some aspect of the environment and a brain process that adapts the animal's behavior to it. [emphasis added]
>
> An isomorphism exists when there is a procedure that maps entities, relations, and operations in the represented system [the environment] into entities, relations, and operations in the representing system [the brain] in such a way that two or more entities within the represented system are related in a given way *if and only if* […] there is a corresponding relation between their representatives in the representing system. [original emphasis]

A similar account of the relation between internal representations and the external world was that of Palmer (1978), who characterized the relation as including the following five aspects: (1) the represented world, (2) the representing world, (3) what aspects of the represented world are being modeled, (4) what aspects of the representing world are doing the modeling, and (5) what the correspondences between the two worlds are. The cognitivist view of the correspondences is one of "representation as an internal mirror of an observer-independent, pre-given external reality" (Ziemke, 2000). The relation can be illustrated as in Figure 1.

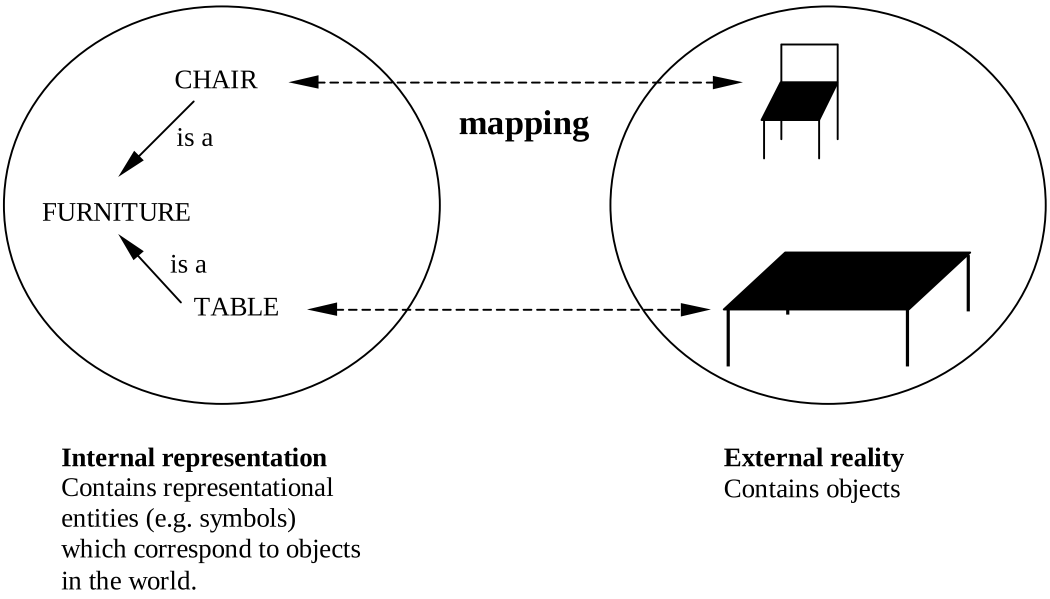


Figure 1: Representation as a direct mapping between internal representational entities and objects in the external world. (Adapted from Ziemke, 2000; Dorffner, 1997).

Cognitivism is predicated on the belief that internal mental representations are realized as symbols, at a distinct symbolic level of analysis. Symbols are arbitrary entities, which means that *any* single symbol (or compound of symbols) may be used to realize a certain mental representation. The semantics of a symbol follow only from the syntactic structures (i.e. the cognitive processes) working on it. Syntax *mirrors* or *parallels* semantics. Symbols are also discrete entities, which means that there is a one-to-one correspondence between each symbol and that aspect of the world which it represents. The rule-governed processing of mental representations, i.e. cognition, is at the symbolic level realized as computations on symbols. This view, that cognition essentially is computation (i.e. a syntactic process), is called *computationalism* or the *computer metaphor for mind*. The Mind is, according to this view, a program implemented on some arbitrary hardware (cf. Varela *et al.*, 1991). The computational processes of Mind may be implemented in many physically different ways and still be functionally equivalent. This is thought to be true because, according to *functionalism*,

> […] things in the world are what they are, not particularly by virtue of what they are *made of*, but by virtue of what *function*, or role, they serve in some kind of system (Rey 1997). [original emphasis]

Thus, it does not matter if the symbol manipulating routines are implemented in a brain or in a machine. The two different physical systems can both implement the same cognitive processes.

2.2 Traditional AI and Inner World Models

According to Newell and Simon (1976) the symbolic level is the proper level of analysis for all cognitive systems and with the Physical Symbol System Hypothesis (PSSH) they asserted that:

> […] a physical symbol system has the necessary and sufficient means for general intelligent action. By “necessary” we mean that any system that exhibits general intelligence will prove upon analysis to be a physical symbol system. By “sufficient” we mean that any physical symbol system of sufficient size can be organized further to exhibit general intelligence. By “general intelligent action” we wish to indicate the same scope of intelligence as we see in human action: that in any real situation, behavior appropriate to the ends of the system and adaptive to the demands of the environment can occur, within some limits of speed and complexity.

Thus, the brain of an organism and a computer with the correctly specified symbol-manipulating program may be regarded as equally intelligent, if both perform the very same thing – certain computations on *symbolic* representations.

The practical consequence of this approach to cognition was a methodology of Artificial Intelligence (AI) research where computers were programmed to solve problems or plan actions according to syntactic processes on representations explicitly specified by a programmer. A lot of effort was put into the creation of the right representations for the AI-system to work with. Choosing the right set of representations became the very key to solving problems (cf. Brooks, 1991a; Franklin, 1995). The specific mechanisms to be implemented could be acquired from, e.g., rational analysis of the task to be solved or verbal reports of human subjects solving the task to be modeled. The reported cognitive processes could then be automated in the computer and tested. If the displayed behavior (read symbol input-output mapping) of the computer conformed to the behavior observed in humans, the verbal reports given or some standard of rationality, then this could be interpreted as evidence supporting the validity of the hypothetical cognitive processes implemented. Newell and Simon (1976) referred to it as "Computer Science as Empirical Inquiry".

Traditional AI research investigated what the inner world of an intelligent agent had to look like in order to solve a task at hand. The inner world was created by making explicit implementations of the necessary internal representations of the outside world. AI was modeling the Mind with special focus on human intelligent behavior, e.g., verbal behavior and problem solving through logical inference. AI was not much concerned with how organisms interact with their environments through sensor and motor processes, but instead focused on more sophisticated problems. When modeling behavior control, agent-internal representations of the outside world were used for the *planning* of actions in order to achieve certain *goals* or *objectives*. According to a definition by Inder (1996) "*planning* refers to determining a sequence of actions you know how to perform that will achieve a particular objective". Since 'knowing' to cognitivists necessitates having internal representations, the existence of internal *goal representations* and *representations of actions* also follows. This knowledge is pre-given when *planning* is studied specifically and one needs only to assume that it is there in the system – in the form of representations. Systems that are thought to naturally control their behavior this way (or are explicitly designed to do so) belong to the class of *goal-directed* systems, since the explicit internal representation of the goal-to-be-achieved is instrumental in guiding the behavior (McFarland, 1989).

Strict divisions were made in order to isolate the really interesting 'cognitive' part of agent-environment interaction in what Brooks (1991a) calls the *sense-model-plan-act* framework for construction of intelligent agents. Perception was essentially seen as a process which produced internal representations. Thus one could go right to the next step and investigate how these representations ought to look and how they should be used to control behavior intelligently, through manipulations of the internal world model. And since behavior was merely a result of the process of translating representations into actions, this step could also be set aside for later. In this way the intelligent system was isolated from the environment. A famous example of control systems based on this approach was the STRIPS planner of Fikes and Nilsson (1971). It was originally designed to control the mobile robot Shakey in a real office environment, but ended up mostly involving simplified simulations instead. The STRIPS language has, since its introduction, become widely used in many planners. As illustrated in Figure 2, Shakey's simulated world was comprised of four rooms connected by a corridor. Each room had a light switch which could be flipped 'on' or 'off'. But, since Shakey was rather short, he had to find a box, move it over to the switch and climb up on it in order to reach the light switch. The STRIPS planner created action plans in order to fulfill goals, by method of means-end analysis. The goal to-be-achieved was given to the planner by an external observer in the form of an explicit representation of the desired state. The planner had a pre-given set of operators, e.g., *Go*(y) or *Climb*(b) could be used if certain preconditions were fulfilled (e.g., the operator *Climb*(b) could be used when the preconditions *On*(Shakey, Floor), *At*(Shakey, b) and *Climbable*(b) were fulfilled). Each operator also had a set of explicitly stated effects on the world. A *plan* was created by working backwards from the stated goal to the current situation. By examining the goal state (a representation of the desired situation) the subset of necessary operations needed to reach that state was determined. Those preconditions in turn became sub-goals to achieve using the same method. When a chain of operations was found that would take the system from the current state to the goal state, via the subgoal states, a complete plan had been found and could be executed. Thus, the generation of a viable behavior sequence was essentially equivalent to searching for the right action plan (Russel and Norvig, 1995; McFarland and Bösser, 1993).

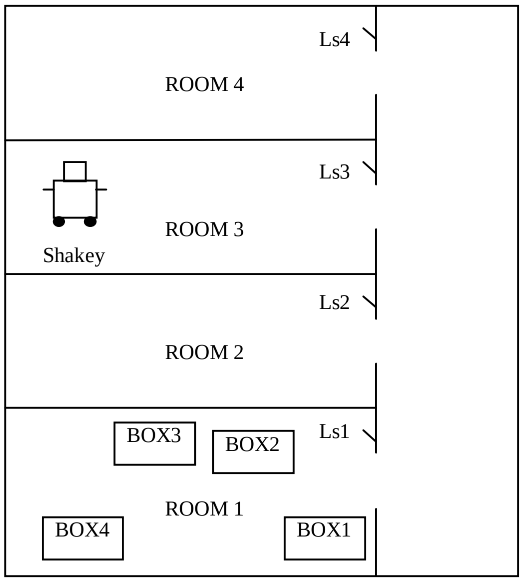


Figure 2: Shakey's world (adapted from Russel and Norvig, 1995).

As the STRIPS example shows, the representational planning approach was not so much concerned with the agent's physical sensorimotor interaction with its environment. In order to make planning problems feasible to solve, certain abstractions on the world were necessary. The key idea was therefor to represent only the really relevant aspects of the world and leave out all unnecessary details. The experimenter responsible for the implementation was the one making the necessary abstractions. But, as Brooks (1991b) argues:

> […] this abstraction process is the essence of intelligence and the hard part of the problem being solved. Under the current scheme, the abstraction is done by the researchers, leaving little for the AI programs to do but search.

It is further worth noting that, after having performed the necessary abstractions and implementations, no further distinction was made between what was *inside* the agent and what was *outside*. The inner world of the agent was assumed to be the same as the outer world – i.e. the agent had been given a 'God's eye view'. The controller (planner) had, in one single step, gained full access to an objective description of the world and it had the ability to work on it directly (Brooks, 1991b). However, as Dorffner (1997) argues, the 'objective' description was only a description of the world provided by the designer who implemented it. Thus, the resulting internal model in a traditional AI system "does not really mirror the world, but the conceptual view of the world by the designer" (Dorffner, 1997), as illustrated as in Figure 3.

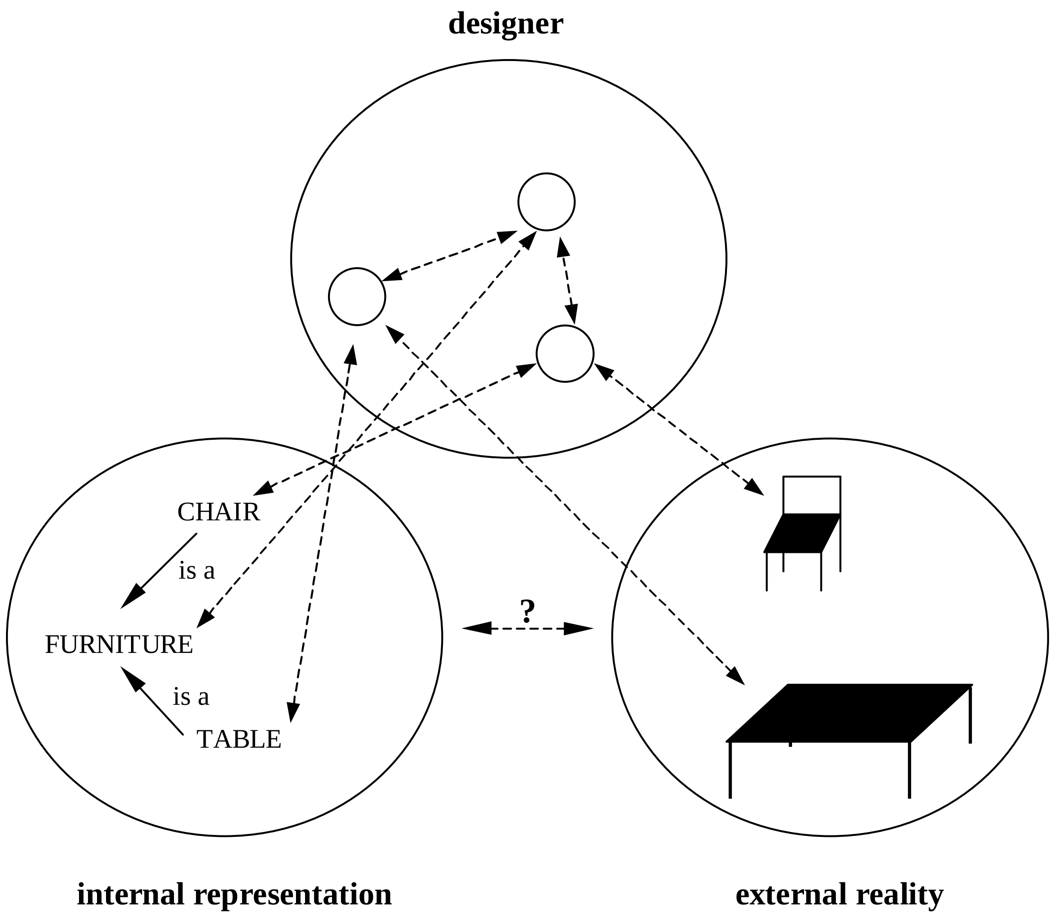


Figure 3: "What 'really' happens in traditional AI representation" (Dorffner, 1997). There is a relation between the designer's concepts and the external reality, and there is also a relation between the designer's concepts and the implemented representations in the AI system. There is, however, no *designer-independent* correspondence between the representations in the AI system and the external reality. (Adapted from Ziemke, 2000; Dorffner, 1997).

Since behavior was abstracted from physical interaction and formalized as a set of operators, most problems of continuously coordinating sensorimotor behavior evaporated from the planning process. Perfect perception and perfect action in a more or less static world was assumed. Once the problems of handling the simpler case of a static environment like Shakey's world were solved the work was to be extended to tackle the more difficult case of behaving in a dynamic environment. This was at least what researchers hoped for in the 1970s and early 80s. But, the practical consequences of these abstractions appeared when the planning systems were ported to reality via physical robots. A growing amount of empirical evidence points to the fragility and inflexibility in this kind of systems. As Beer (1990) describes the results:

> Their performance is extremely sensitive to the representational choices made by their designers, and brittle in the face of inevitable deviations of the real world from these abstractions. They are incapable of flexibly coping with contingencies not explicitly foreseen by their designers.

In the real world, perception in the cognitivist sense is rarely perfect, rendering difficulties in constructing a correct internal model. Similar problems can appear due to unforeseen consequences of actions. It has been argued from the 'New AI' community (e.g. Brooks) that in the traditional assumptions about perception, action and representations lie the weaknesses of the traditional approach to understanding the inner world.

2.3 A New AI Approach

The traditional Cognitivist theoretical framework and approach to understanding intelligence could be described as 'top-down' in its assumptions of Mind, mental representations and symbols as the starting points for further investigation. In contrast, a number of radically different 'bottom-up' approaches have over the last 15 years grown stronger, advocating the situated nature of natural intelligent systems as the fundamentals for understanding intelligence. Cognitivist notions like 'Mind', 'mental' and 'representations' are either reinterpreted, carefully avoided or dismissed all together. Nowadays many researchers doubt that these terms *necessarily* play a role in explaining and understanding intelligence. The new perspective is articulated in Inman Harvey's words on the brain's neural mechanisms and their relation to the outside world:

> […] there is no *a priori* reason why all parts of the brain should be in such a modular form that representation-talk is relevant. […] This does not rule out the possibility that in some circumstances representation-talk *might* be useful, but it is an experimental matter to determine this (Harvey, 2000). [original emphasis]

The focus of attention is instead put on the situated nature of intelligent behavior, the properties of nervous systems and what it means for an animal to be adapted to its environment.

Contrasting with traditional cognitive science and traditional AI, a view of intelligence as *adaptive behavior* is advocated by, e.g., Beer (1990) and Balkenius (1994). Beer argues for an AI methodology which he calls *computational neuroethology* – computer modeling of the neural mechanisms underlying intelligent behavior. Adaptive behavior, as defined by Beer, is "behavior which is adjusted to the environmental conditions". Intelligence as adaptive behavior thus means that the animal's (or the robot's) ability to change its behavior appropriately, when conditions in the world change, determines to what extent it may be called intelligent. Balkenius (1994) uses the example of a squirrel collecting nuts in apparent anticipation of the winter. The observed behavior fits well the squirrel's present and future needs and an external observer may conclude that the behavior is intelligent, since it apparently is adapted to the environmental conditions. Balkenius points out that this is the idea behind the words of Brooks (1991a): "intelligence is in the eyes of the beholder". The term *intelligence* is thus a qualitative judgement on behavior, not an intrinsic property of the mechanisms behind it (e.g. symbol manipulation). If the squirrel's conditions are changed by, e.g., filling up the store of nuts so that it has more than enough for the winter, and if the squirrel still continues to collect more, then one may question its intelligence. But if it, on the other hand, would change its behavior to adapt to the new circumstances, then we would still think of it as intelligent in some sense. Balkenius concludes, from this, that intelligence seems to be equivalent to the capacity to learn. In the vocabulary of Maturana & Varela's biological theory of cognition (cf., e.g., Maturana and Varela, 1987) adaptive behavior is the result of a *structural congruence* between internal and external dynamics. The dynamics of the internal neural mechanisms are such that they, in interaction with the environment, trigger appropriate behavior, but the mechanisms need not necessarily qualify as corresponding representations of the environmental aspects to which they are adapted.

Sharkey and Heemskerk (1997) have proposed making a distinction between *distal* and *proximal* descriptions and explanations of behavior, in order to properly understand the mechanisms behind the observed intelligent behavior. Distal descriptions refer to the observed behavior and these descriptions are provided by an external observer who has categorized the behavioral observations into some chosen constituents, often using intentional terminology (e.g. 'searching for food', 'avoiding obstacles'). Proximal descriptions or explanations of behavior instead refer to the sensorimotor mechanisms that *underlie* the observed behaviors. Just as there is not necessarily a representational relation between internal mechanisms and aspects of the environment, there need not be a one-to-one mapping between the two types of descriptions, i.e. categorized *behaviors* may not be easily mapped to specific mechanisms. Instead, the behaviors can *emerge* from the interaction of internal mechanisms, and the interaction between congruent neural mechanisms and the environment (cf., e.g., Ziemke, 2000).

It has been argued by several researchers in this field (e.g. Balkenius, 1994; Beer, 1990; Brooks 1991a, 1991b) that adaptive behavior should be modeled using *complete agents* or *whole systems* (preferably physical mobile robots) in interaction with a dynamic (preferably real and complex) environment. By doing this, one works with artificial systems that resemble real organisms in that they will, in some sense, be *situated* and *embodied*. According to Brooks, these terms may be characterized as follows:

> **[Situatedness]** The robots are situated in the world – they do not deal with abstract descriptions, but with the here and now of the world directly influencing the behavior of the system.
>
> **[Embodiment]** The robots have bodies and experience the world directly – their actions are part of a dynamic with the world and have immediate feedback on their own sensations (Brooks, 1991a). [original emphasis]

Furthermore, the long term goal is to have *autonomous* agents, which in the case of AI essentially means that they are *independent from human design* (cf. Ziemke, 1997). As illustrated in Figure 3, design choices made by the experimenter impose *a priori* restrictions on the artificial agent's internal mechanisms and thus its ability to develop structural congruence with its environment. A 'truly' autonomous system, such as a living organism, works under no such designer influenced restrictions based on *a p riori* assumptions. The multi-facetted nature of the term *autonomy* and the current limitations on all of its aspects, imposed by reasons of designer influence and practical constraints, have been thoroughly discussed by Ziemke and Sharkey (cf., e.g., Ziemke, 1999a; Sharkey and Ziemke, 1998). However, even if the ultimate fulfillment of all these goals remains, the very fact that they *are* working goals has effect on this new methodology in AI.

Beer (1990) and Brooks (1991b) have also proposed starting with modeling *simpler systems* and *simpler behaviors*. There are two main reasons for this. Due to the behavioral complexity that comes from interactions between even a fairly simple agent and its environment, there is a need to understand the fundamental characteristics of these interactions. Insights gained from working with models of simpler systems and behaviors can be applied to modeling slightly more complex ones. This cautious advice comes mainly from practical considerations. But there are also evolutionary motivations for working incrementally from simpler to more complex systems and behaviors. As Beer (1990) writes:

> It is a striking testament to human conceit how little effort in AI has been expended on modeling the behavior of simpler animals. While some of our higher cognitive functions appear to be unique among the animal world, there is no reason to believe that they are completely discontinuous with the abilities of simpler animals. After all, human beings did evolve from simpler animals in the first place. Most scientific and technological endeavors seek to understand and construct simpler systems before tackling the most complex ones.

Furthermore, as Brooks (1991b) notes, the majority of time elapsed in the evolution of life has been spent on developing abilities like locomotion and foraging. Only for a very short time in the history of biological life on this planet have human skills like language use and symbol manipulation been around. All evolutionary solutions of a later date are built upon previous ones. So, to fully understand the nature of modern human abilities one must understand the longer biological and evolutionary history behind it.

2.4 Adaptive Neuro-Robotics

Much bottom-up AI research, aimed at understanding the proximal mechanisms behind intelligent behavior, has since 1990 made use of mobile robots with control systems capable of learning and self-organization. Often the control system is some form of Artificial Neural Network (ANN) used as an '*artificial nervous system*' connecting the robot's receptors and effectors. The neural mechanisms are developed through self-organization using lifetime and/or evolutionary adaptation techniques. Ziemke (2000) uses the term *adaptive neuro-robotics* in referring to this line of AI-research.

To understand fundamental characteristics of adaptive behavior, Dorffner (1997) argues for an approach to AI, which he calls Radical Connectionism (RC). It is a "neural bottom-up approach" which advocates the connectionist formalism of ANNs as its basis for modeling of intelligent behavior in artificial agents. The main difference, when compared to traditional connectionism, is the abandonment of using input and output representations whose semantics are given and interpreted by an external observer. Instead, the ANN should make sole use of *sensorimotor interfaces* – inputs are immediate *sensory stimuli* and outputs are *motor responses*. The modeled systems are to interact with an environment, thus making motor outputs and their consequences on sensor inputs intimately coupled in order to make the agent embedded and situated in its environment. The internal structures necessary for appropriate behavior should further develop through self-organization instead of design. The general approach to guiding this self-organizing process is to use some form of learning method. The key feature of the 'New AI' approach presented here lies in the efforts to make *minimal assumptions* about the internal mechanisms underlying intelligent behavior. This is especially relevant in the case of RC, since the approach to a high degree relies on self-organization rather than design.

Artificial Neural Networks (cf. Figure 4) are computational models that capture some coarse aspects of natural nervous systems. ANNs are built up of small units which can have some level of activation, e.g., a value between 0 and 1. These units can form networks via weighted connections that resemble synapses in that they can propagate the activation of one unit to other units, with different degrees of strength as given by the connection weight. The effect of propagated activation can be either excitatory (increase activation) or inhibitory (decrease activation) on the receiving unit, depending on if the connection is weighted negatively or positively. The network receives input in the form of a vector of activated units. This activation vector is then propagated over weighted connections to receiving units at a layer above input. After the activation vector has been propagated through all layers of connection weights and units, the network delivers output in the form of an output vector of activated units at the output level. Each unit receiving activation – either from other units in a layer below or as input from sensors – works by summing up the incoming activation levels and passing that summed activation level through an activation function which determines whether or not the incoming activation will make that unit active or not (Plunkett and Elman, 1997).

Feedforward networks, such as the one depicted in Figure 4a, are reactive systems and thus always deliver *one specific response to a specific stimulus*. Recurrent Neural Nets (RNNs) are augmentations of this architecture, with self-recurrent feedback connections (cf. Figure 5). In RNNs, unit activations at

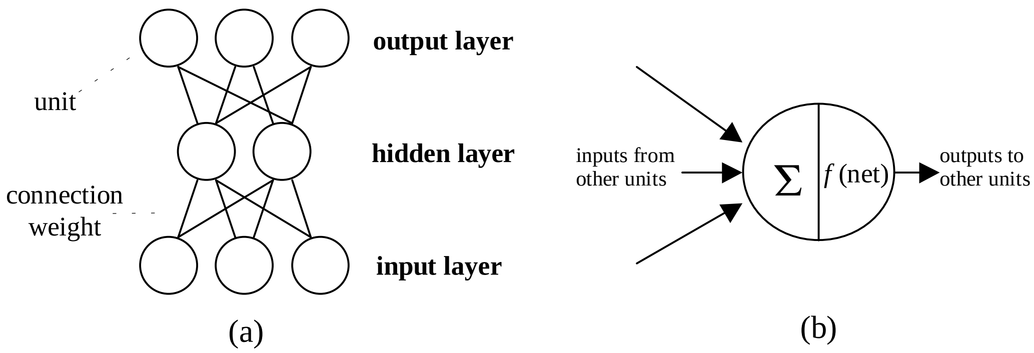


Figure 4: (a) A typical feed-forward ANN, with units and weighted connections. (b) A single unit with an activation function *f*(net), where net is the sum of all incoming activations (adapted from Plunkett and Elman, 1997).

previous timesteps are able to influence responses to future input, thus realizing a form of *internal state* or *context*. This internal context adds behavioral dynamics such that the network can deliver *different responses to identical stimuli*, depending on the current internal state of the network.

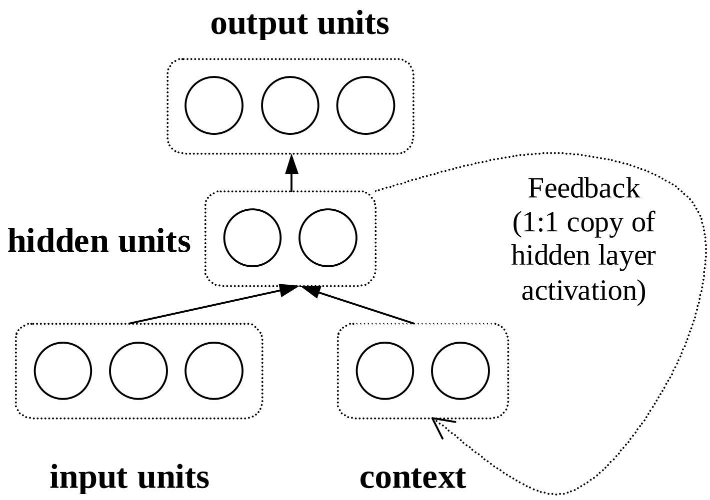


Figure 5: Recurrent artificial Neural Network (RNN). The Simple Recurrent Network architecture (Elman, 1990). Each layer of nodes is fully connected to the layer above. Hidden layer unit activations are copied to the input layer at the following timestep, thus giving context to the input.

Neural networks can be trained with a wide range of learning methods and algorithms to make Stimulus-Response associations. A common approach to associative learning is to adapt the connection weights of a fixed topology network. This means that the network architecture is designed and fixed by an experimenter and that the architecture is not subject to change. Only the weights of the network connections are adapted. Feedforward network weights are often adapted using variations of the back-propagation of error algorithm (or ‘backprop’) of Rumelhart, Hinton and Williams (1986). The algorithm usually demands that the set of input-output pairs to be associated by the network is specified *a priori* by the experimenter. However, this form of supervised learning is typically not used when training behavior in autonomous robots, controlled by RNNs. There are two reasons for this (cf. Ziemke, 2000): Firstly, as has been argued and experimentally shown, e.g., by Meeden (1996), connection weights are in many cases more successfully adapted through reinforcement learning, using Evolutionary Algorithms (EAs), for example a Genetic Algorithm (GA) (cf. Mitchell, 1996). Secondly, feed-back on behavior, when using an EA, is less influenced by design when compared to back-propagation methods. No set of specific input outputs pairs at each timestep needs to be pre-specified for training. Actually, this kind of moment-to-moment guidance is typically not even possible to give the learning robot when it is interacting with its environment, since there is not necessarily just one correct action for a given situation. And even if there is, it would not typically be known *a priori* (Meeden, 1996). Instead of explicitly guiding behavior learning, an EA is typically employed to reinforce behaviors in terms of more abstract goals, from an observer's distal perspective, while allowing the proximal mechanisms to self-organize.

EAs are basically search algorithms, inspired by the process of Darwinian evolution, i.e. adaptation through variation and non-random selection of hereditary traits. When applied to learning in ANN-controlled robots, the algorithm can be used to search for mechanisms (e.g. connection weights) which make the robot produce some desired behavior in its environment. The desired behavior can be captured in an abstract goal description, expressed as a mathematical function to maximize, e.g., if the abstract goal is 'move as fast as possible', the function to maximize could be the measured velocities of wheel rotation that the robot produces, or the distance traveled in a certain time. A common procedure is to have a population of individuals (e.g., matrices of connection weights in an ANN controller) evolve over a large number of generations, by repeatedly selecting individuals which measure well (i.e. make the robot behave well) according to the fitness function and have them reproduced to form a new generation, by recombinations and slight mutations. Mechanisms (e.g. combinations of connection weights) that contribute to 'good' behaviors tend to be selected and spread in the population over successive generations, thus improving the average population fitness over time (cf., e.g., Nolfi and Floreano, 2000). In this thesis, the term *learning* will be used in the 'wide' sense of *adapting* behavioral mechanisms, thus including artificial evolution as a learning method.

An important aspect of evolutionary robotics (a particular case of adaptive neuro-robotics) is that the evolutionary process works by determining the viability of *observed robot behaviors* in order to perform selection. In the case of evolving connection weights, each individual set of connection weights is tested by 'implanting' it in a robot and then determining the resulting quality of the robot's behavior when interacting with its environment. This process works at the level of *distal descriptions of behavior*, although not using intentional terminology, but instead mathematical functions describing the desired observable behavior. But all the adaptive changes of that behavior take place at the level of *proximal mechanisms*, e.g., in the form of connection weight mutations. These low-level changes in mechanisms are only implicitly guided by the distal goal description employed in selection at the behavioral level. There is no direct human design influence on the selection of neural mechanisms that participate to produce the behavior selected for, since that selection process works on a separate level of description to which there is no one-to-one mapping. Furthermore, if the goal is described in sufficiently abstract terms, then there is no direct influence on the selection of specific successful behaviors either, since for many problems it is possible to perform equally well, measured in abstract terms, when employing different behavioral repertoires. Furthermore, both the observed behaviors and the proximal mechanisms behind them, may be quite different from the most intuitive solution. Adaptive neuro-robotics in general, and evolutionary robotics in particular, realizes the wanted aspects of embodiment and situatedness in artificial systems to some extent, by combining the use of robots in real environments and neural network controllers adapted through self-organization. The following sub-sections present some relevant examples of experimental work within the field.

2.4.1 Emergent obstacle avoidance and homing behavior

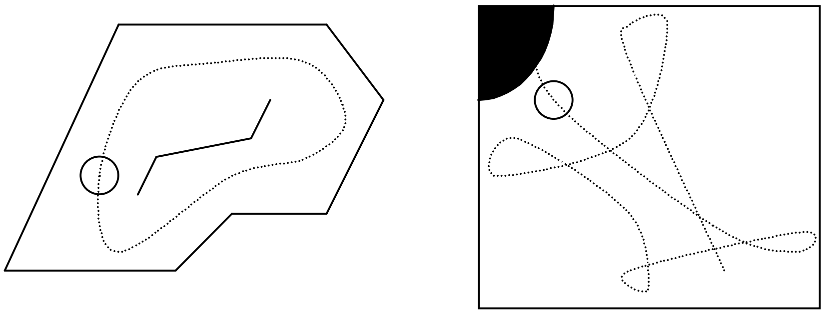
Floreano and Mondada (1996) carried out two experiments illustrating how neural network controllers could be successfully trained to meet abstract behavioral goals, through self-organization of proximal mechanisms. In both experiments, fixed topology ANNs were used and the connection weights were evolved using a genetic algorithm. The robot (cf. Figure 14 in Section 3.2) was controlled by a neural network with eight input units, directly coupled to each of the robot's eight IR-sensors, and two self-recurrent output units, directly controlling the robot's left and right motors respectively.

In the first experiment, a population of 80 ANN robot controllers was trained to navigate and avoid obstacles (walls) in a small looping maze environment (cf. Figure 6a). The fitness criterion Φ to maximize, described the task to be learned, as a function of three variables,

$$\Phi = V\,(1-\sqrt{\Delta v})\;(1-i)$$

where V was a measure of the average rotation speed of the two wheels, Δv was the absolute value of the algebraic difference between the speeds of each rotating wheel and i was the activation value of the highest activated IR-sensor. This function says, in other words, 'keep moving as fast as possible in any one direction, with as little turning and sensor activation as possible'. It says nothing about *how* to do this. In the looping maze environment this goal can be realized by making laps around the inner walls. In less than 100 generations the robots had learned to do this smoothly and reliably and avoid the obstacles, while keeping a cruising speed of approximately 75% of maximum available speed.

In the second experiment Floreano and Mondada wanted to investigate whether the robot could discover the presence of a battery recharging area and adapt its global behavior using an even simpler fitness criterion. As depicted in Figure 6b,



(a) (b)

Figure 6: Example trajectories in (a) the looping maze environment (approximately 80 * 50 cm), (b) the battery recharge environment (40 * 45 cm). Adapted from Floreano and Mondada (1996).

a rectangular environment was used with one of the corners painted black to represent a recharging area. When the robot happened to be over the black area, its simulated battery became instantaneously recharged. The robot was to move around without collision in the non-black part of the environment, but could only do this as long as it had power in its battery. The fitness criterion Φ to maximize, described the task to be learned, as a function of only two variables,

$$\Phi = V(1-i)$$

where V was a measure of the average rotation speed of the two wheels and i was the activation value of the highest activated IR-sensor. This function only says ‘keep moving as fast as possible in any one direction, with as little sensor activation as possible’. It says *nothing* about battery recharging, light or lifetime. A population of 100 robots was trained on the task for 240 generations. Analysis of the resulting behaviors showed that a range of strategies had evolved which solved the task quite elegantly. Individuals were able to repeatedly time their behavior so that they turned to the recharging area to recharge, with only 2 seconds left before complete battery discharge. After recharge, they quickly left the black area again and only returned whenever complete discharge was near.

2.4.2 Emergent ‘planning’

To move the connectionist approach to robot control beyond the level of reactive behavior Meeden (1996) proposed an “incremental approach, beginning with simple reactive behavior and gradually building up to more memory-dependent behavior" to eventually reach the level of planned behavior.

Meeden (cf. Meeden, McGraw and Blank, 1993; Meeden, 1996) conducted a series of robot experiments using *Carbot*, a modified toy car with primitive sensor and motor capabilities. It had two digital touch sensors in the front and two in the rear and two analog light sensors directed 30 degrees to each side. It had one steering motor for left and right turns of the front wheels and one motor for backward and forward rotation of the rear wheels. As depicted in Figure 7, a low-complexity static environment was used, consisting only of a relatively small rectangular box with a light source in one corner. The robots behavioral task had two levels: keep moving and avoid walls (rudimentary navigation level); seek or avoid the light, depending on the current goal (goal level). The environment was designed to limit Carbot's freedom of motion, so that it could not make a 180 degree-turn in one single step. In order to switch between fulfilling the 'approach' and 'avoid' goals respectively, Carbot was forced to co-ordinate coherent *sequences* of forward and backward turning motions.

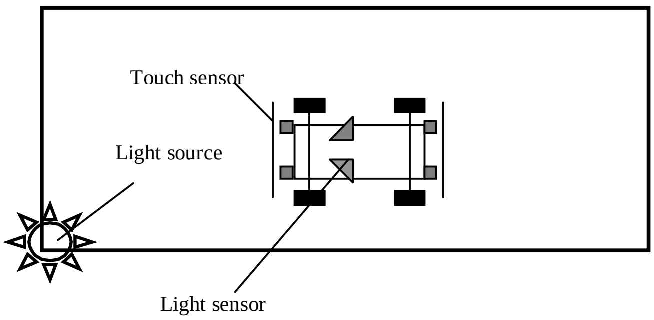


Figure 7: Schematic drawing of Meeden's Carbot and its environment. (Adapted from Ziemke, 2000).

The controller network (cf. Figure 8) was similar to a Simple Recurrent Network (Elman, 1990), with self-recurrent connections at the hidden layer, thus allowing for a short-term memory of its own encodings of the past input states, and giving context to the sensor input. Input consisted of sensor activation (digital and analog). The two output units controlled the left/right steering motors and two units controlled forward/backward motion. In Meeden (1996) an additional goal-node was given at the input layer and used in two of the experimental conditions. When used, the goal node was set to +1 when the goal was to seek the light, and –1 when the goal was to avoid the light. Once a goal was reached it changed to the opposite. Thus, the robot was required to alternate between two opposing behavioral repertoires, depending on the currently given goal to achieve.

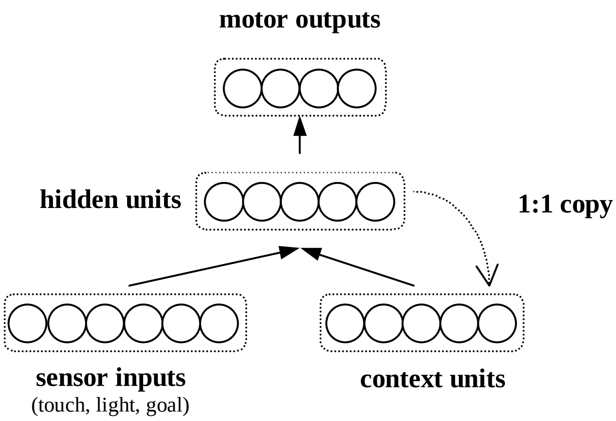


Figure 8: The controller network architecture used in Meeden (1996). Each layer of nodes is fully connected to the layer above. Hidden node activation is copied back to the input layer as internal context for sensor input at the following timestep.

Three experimental variations, giving different amounts of feedback in the adaptation process, and two different reinforcement methods were used to adapt the connection weights in the network. The two training methods were (a) a modified version of the backpropagation algorithm and (b) an evolutionary algorithm. The experimental results showed the evolutionary algorithm to be superior in solving the more demanding conditions and resulting in more generally stable solutions.

As depicted in Figure 9, several evolved behavior strategies were observed. The simplest solution was a forward/backward semi-circle strategy for approach and avoidance alternatively. A more complex two-point turn strategy, utilizing sensor input from hitting a wall to change direction also appeared. Finally, a ‘many-point turn’ strategy was also observed, which allowed for wall avoidance through a ‘star-like’ motion pattern near the center of the environment. The two more complex strategies were more than simply following the rewarded light gradient. Instead, momentary motions *away* from the ‘beacon’ (the light source) were executed, in order to pursue the overall goal to ‘approach’ and the opposite, i.e. momentary motions *towards* the light source, were executed in order to fulfill the ‘avoid’ goal. Carbot was thus able to co-ordinate *sequences* of actions, instead of simply reacting to environmental stimuli. But, at the same time, it was able to use sensor input in obstacle avoidance. Meeden *et al.* (1993) argued that Carbot’s behavior was *plan-like* in the sense that (a) it associated abstract behavioral goals with sequences of primitive actions, (b) the behavior could be described in hierarchical terms by clustering behaviors into categories, e.g., a *seek light* category comprised of the sub-categories *orient towards light* and *go to light* which in turn were realized as primitive actions of moving and turning, (c) the robot maintained its overall strategy even when reacting flexibly to the environmental conditions. On the other hand, the behavior was not plan-like in the traditional sense that the robot would explicitly anticipate future situations in order to control behavior. However, in Meeden *et al.* (1993) experiments were conducted where Carbot was trained on the secondary task of predicting the next sensor state. This was done in order to investigate the effects on behavior learning. However, no analysis was presented regarding the quality of those predictions made.

(a)
(b)
(c)

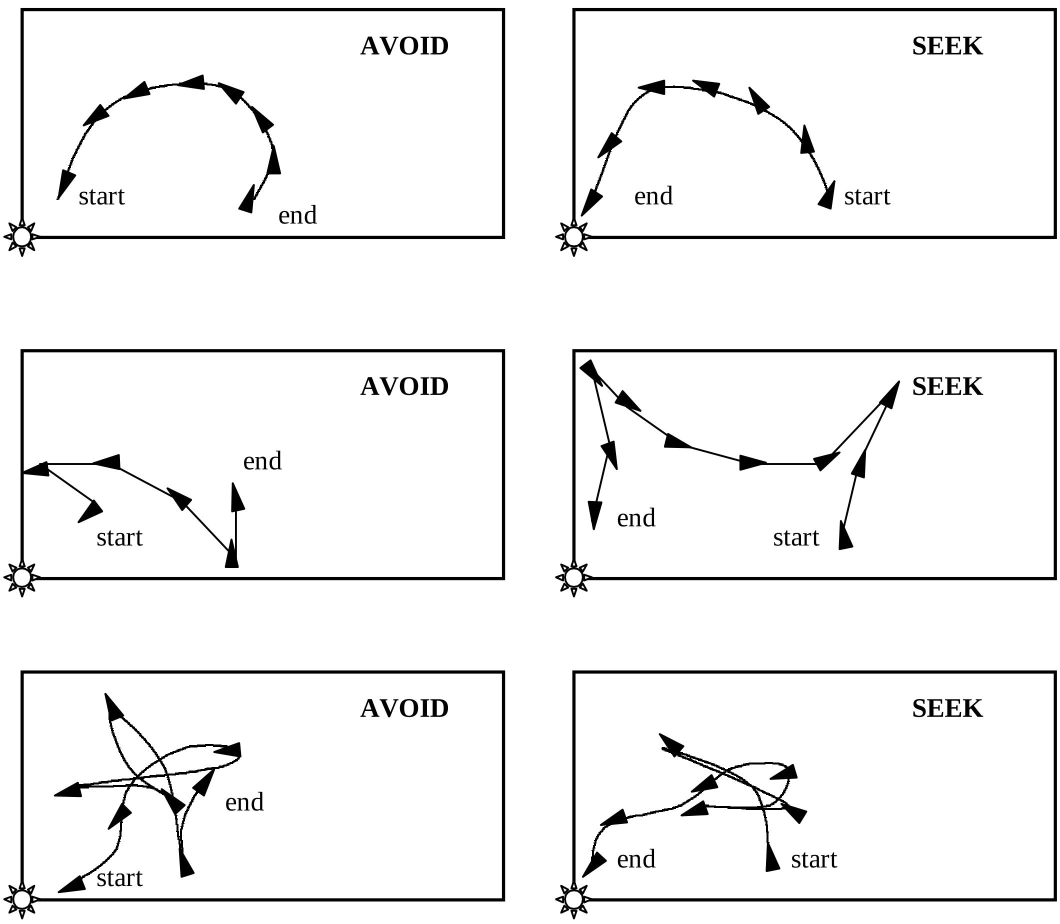


Figure 9: Schematic illustrations of evolved strategies in the Carbot experiments. (a) Semi-circle strategy. (b) Two-point turn strategy. (c) Many-point turn strategy. Adapted from Meeden (1996).

2.4.3 Concept formation through prediction

Tani & Nolfi (1999) investigated how an agent could learn “an internal model of the world” from the agent’s own perspective, by developing internal dynamics *congruent* with aspects of its environment. The key idea was this: An agent moving about in an environment is exposed to a continuos flow of sensor activations. The flow of sensor activations is dependent on the dynamical interaction between (a) the environment and (b) the agent’s own movements. If the agent is given the task of predicting its future sensor states, then the agent’s *internal dynamics* will have to be adapted to the *dynamics of the sensory flow* that comes from the agent-environment interaction in order to be correct. For example, in order to correctly predict the appearance of a corner on the left side after ten time steps of straight motion in a corridor, the agent must have developed some internal dynamics that are congruent with these environmental characteristics. In having to predict future sensory states, the robot is forced to find regularities in its sensorimotor flow, thus (through adaptation) making its internal mechanisms congruent with the changes in the perceived world when acting upon it.

A hierarchical architecture of RNN modules (cf. Figure 10) was trained on different levels of prediction, while the robot moved around in its environment. Five RNNs at the bottom level continuously competed in making predictions of the next sensor state. Each of the lower level networks could thus (through learning) become experts in dealing with certain aspects of the environmental interaction; one may become attuned to the dynamics of corner behavior and another may be better at predicting the sensory-motor flow when following a straight corridor. The lower level expert RNN making the best predictions was said to be ‘active’. Five higher level RNNs were in turn trained to predict which of the lower level RNNs would be active, thus adapting the higher level RNNs to find structure in the sequences of perceived environmental characteristics at the lower level. The networks received sensor and motor information from a simulated robot with 20 laser range sensors which was pre-programmed to perform collision-free movement in an environment of two rooms connected with a closable door (cf. Figure 11). The two rooms were of the same size, but with wall structures of different arrangements creating different corridors. The robot traveled around one room for three laps, then the door was opened and it traveled three laps in the second room. The learning continued for a total of five encounters with each room.

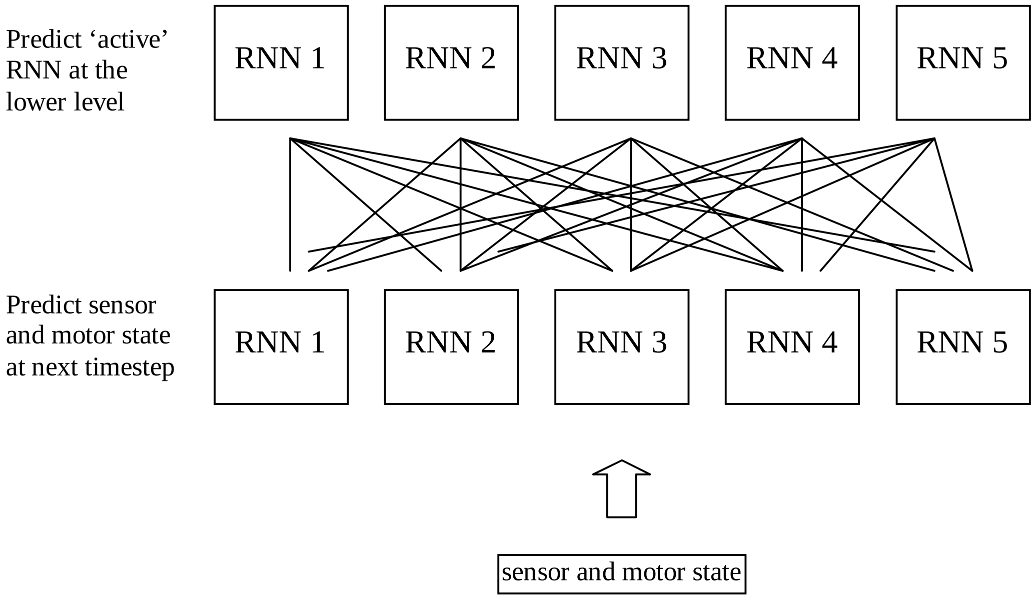


Figure 10: Hierarchy of Recurrent Artificial Neural Networks (RNNs) in Tani and Nolfi (1999). Each of the lower level RNNs receives the current sensor and motor state and competes at predicting the next sensor and motor state. Each of the higher level RNNs competes at predicting the 'active' (winning) lower level RNNs.

Earlier work in traditional AI on machine learning, e.g., in landmark navigation, has been based on an approach where designers have decided beforehand what should constitute a landmark for the robot. These landmarks are not necessarily "intrinsic to the perception of a robot", as Tani and Nolfi argue. Instead, these concepts should be generated from the robot's experience – its sensory-motor flow in interaction with the environment. This experiment focused on the construction of *for the agent meaningful* 'concepts' for aspects of its world, e.g., what a 'corner', a 'wall' or something else is, in terms of the sensory-motor dynamics as perceived by the agent. But, since these concepts are based on the agent's sensor readings from the world, the formed concepts are perhaps not meaningful to us as external human observers. The concepts (to be constructed by the lower level RNNs) could be used as parts in a hierarchy describing more complex entities, e.g., 'walls' and 'corners' together may form 'rooms' in the higher level RNNs.

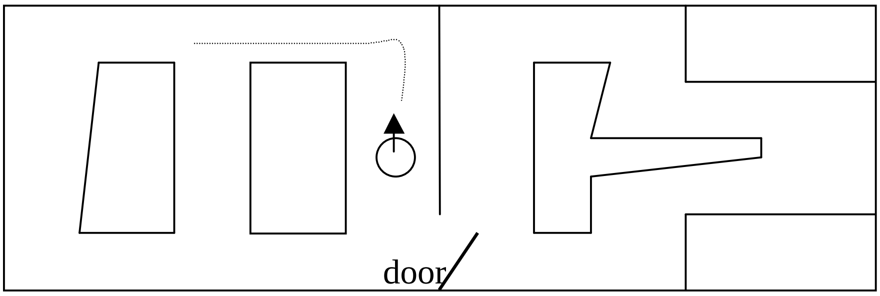


Figure 11: Robot in the simulated environment consisting of two rooms connected by a door. (Adapted from Tani and Nolfi, 1999).

After training, four types of meaningful concepts had been generated at the lower level, using four of the five available modules. Concepts for ‘following a straight wall’, ‘making a right turn at a corner’, ‘making a left turn at a corner’ and finally, ‘passing a T-junction’, were formed in the respective modules. In the higher level RNNs (after stabilization at the lower level), two of the five networks generated concepts for ‘room A’ and ‘room B’ respectively, thus forming a hierarchical ‘world model’ of two rooms consisting of differently arranged sequences of walls, corners and junctions. Concept formation in a module should here be understood as a *tuning* of that module, so as to have intrinsic dynamics corresponding to the dynamics of interacting with a certain part of the environment, thus allowing that module to make correct predictions of future states.

2.5 Internal Simulation

How may the 'inner world' be understood in terms of something other than a small-scale representational model of the outside world? Germund Hesslow (cf. Hesslow, 1996; Hesslow, 1994) argues that it could be understood in terms of *internal simulation* of perception and behavior. The approach is basically behaviorist and avoids talk of manipulations of (mental) representations and it thus fits well with the presented 'New AI' approach to modeling intelligent behavior, using neural networks trained to associate stimuli and responses. The following approach to describing the underlying mechanisms is assumed (Hesslow, personal communication): An animal placed in an environment is exposed to a number of external stimuli (S), such as objects reflecting visible light which affects receptors on the animal's retina. These physical phenomena causally give rise to some neural activations in the animal's nervous system, which are propagated to the brain, resulting in an activation (s) in the relevant sensory area of the brain, e.g., in the primary visual cortex. The activation (s) spreads to different neural structures in the brain via various connections and eventually results in an activation of a motor area. This activation of a motor area, e.g., the premotor cortex, can be understood in terms of a motor response preparation (r). It is a neural activation pattern which, if propagated further to primary motor cortex and to the peripheral nervous system and muscles, will cause a motor response (R).

As a result of evolutionary adaptation and/or life-time learning, certain stimuli become *associated* with certain responses, such that if the animal is exposed to a relevant stimulus it will produce an appropriate motor response, much like the robots described in previous sections. Motor responses to stimuli can be observed from the outside as *overt behavior*. As illustrated in Figure 12, we could describe an agent's overt behavior when interacting with its environment, in a simplified way, as a sequence of stimulus-response pairs. In the initial situation a stimulus (S) triggers a response (R), which changes the situation as seen from the animals perspective such that a new stimulus (S´) triggers another response (R´), which in turn causes S´´ and R´´ and so on. All these steps go *via* the inner processes of sensory states (s) causing motor response preparations (r).

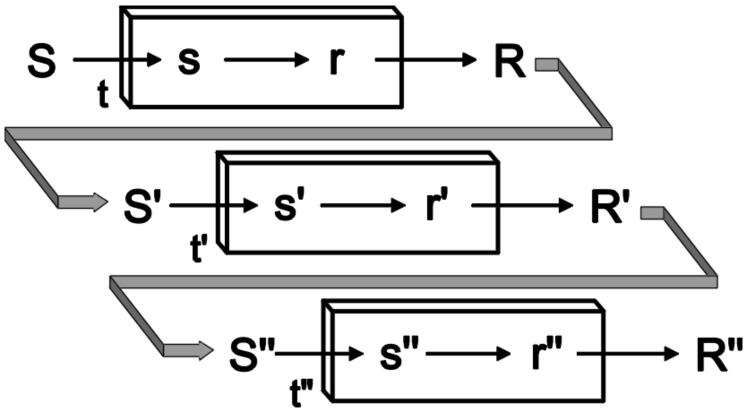


Figure 12: A stimulus-response sequence during overt behavior. At time t, the stimulus (S) causes a response (R), which, at time t´ causes a new stimulus (S´) and a new response (R´) which in turn causes (S´´) and (R´´) at time t´´. (Courtesy of Germund Hesslow.)

According to the description above, we have assumed that an internal process takes place in the time between exposure to the stimulus and the observed response, where the sensory activation (s) gives rise to a motor response preparation (r). We could call this generation of a motor response preparation, *covert behavior*. It is the same neural process as for overt behavior, except that the neural activations do not propagate all the way out to the peripheral nervous system and muscles thus causing observable motor responses. The motor response (preparation) stays internal. Next, assume the existence of a *sensor reactivation* or *imagery* mechanism, which allows for internally generated activation of sensor areas in the brain, so as to produce the simulated experience of a stimulus, but without the presence of the external stimulus. The sensory area could, e.g., be activated via connections from some neural structure shaped by previous experiences (i.e. a memory). For example, if someone says the word *apple*, the sensor reactivation mechanism could cause a listener to not only have the auditory experience of the uttered word, but also, via connections shaped by previous experiences of apples, have the visual experience of an apple, even though that visual stimulus is not present. Finally, assume the existence of an *anticipation* mechanism, i.e. an ability to predict the sensory consequences of a motor response (a bodily movement). For example, anticipation of a perceived size increase when stepping closer to an object. Support for these assumptions can be found in several discussions on neuroscientific research. Covert behavior and imagery (sensor reactivation) and data supporting their existence are discussed in, e.g., Jeannerod (1994). Imagery is further discussed in Kosslyn, Behrmann and Jeannerod (1995). Finally, in a number of articles (Miall, Weir, Wolpert and Stein, 1993; Miall, 1997; Miall and Wolpert, 1996; Wolpert, Miall and Kawato, 1998; and also Thach, 1998), the cerebellum is considered a prime candidate module involved in sensory prediction and anticipation, e.g., in guiding of motor responses.

With these three mechanisms (i.e. covert behavior, sensor reactivation and anticipation), it would (in principle) be possible to *internally simulate* the overt behavior sequence described above, in the following manner (cf. Figure 13): In the initial situation, a stimulus (S) causes a sensory activation (s) which triggers a motor response preparation (r). But instead of causing the overt response (R), the motor response preparation could, by influence of other neural activations inhibiting the motor response, cause a new sensory activation (s´) via a mechanism for anticipation of the sensory consequences and, in turn, a sensor reactivation mechanism. This new (internal) sensor activation triggers another motor response preparation (r´), which in turn causes a new sensory activation (s´´) which triggers a motor response preparation (r´´) and so on. Instead of overtly interacting with the environment, the agent internally simulates this interaction covertly. It behaves in its inner world.

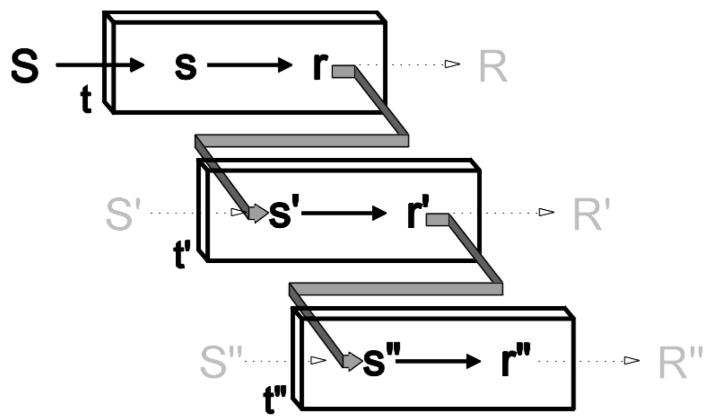


Figure 13: A sequence of internal simulation. At time t, the stimulus (S) causes a response preparation (r), which, at time t´ causes a new stimulus (s´) via anticipation and sensor reactivation. This internal sensation causes a new response preparation (r´) which in turn causes (s´´) and then (r´´) at time t´´. The overt responses R-R´´ are not elicited and thus the environmental situations S´ and S´´ are only simulated. (Courtesy of Germund Hesslow.)

Experimental AI research on robot navigation, with control systems inspired by neuroscientific empirical findings and explanatory constructs, may prove internal simulation to be important in sensorimotor interaction. By drawing on psychological, physiological and neuroanatomical data on the properties of the cerebral cortex, Gross, Heinze, Seiler and Stephan (1999) constructed a computational neural model (called the Model for Anticipation based on Sensory Imagination, or MASIM) allowing to handle the problem of anticipatory search and internal simulation in order to explain perception and generation of behavior in 'brain-like systems' at the level of sensorimotor intelligence. Gross *et al.* present a detailed and quite complex computational model based on an extensive background of evidence for anticipatory systems in the brain. Anticipation and prediction, in their analysis, involves several cortical and sub-cortical systems and a number of different processes in natural neural systems. The functional role of the cerebellum for anticipation and prediction is especially interesting in this case. Furthermore, anticipation and prediction seem to play a central role in sensorimotor interaction with the world. The details of the MASIM-architecture will not be discussed here, due to its complexity and limited relevance to the experimental work conducted here. But, the main characteristics of the approach and the experimental results are certainly interesting. The objective of their approach was "to demonstrate the general idea and the advantages of an anticipation-based perception in the light of several evolvable and observable behaviors of a real sensorimotor system". And since "observable behaviors are the only indicators to evaluate and compare the perceptual performance of sensorimotor systems as a whole" they compared the obstacle-avoidance performance of an anticipating mobile robot to that of a reactive mobile robot in several different environments. The robots used optical flow fields as sensory input. The experimental results showed the "perceptual superiority" of the anticipatory approach, in that the anticipating systems showed better navigation performance with higher speed and far fewer collisions. Most importantly, the anticipating systems were able to turn, in order to avoid obstacles, long before coming close to them. In contrast, the reactive system often tried to drive straight forward as long as possible before it started some hard turns, thus resulting in more collisions.

2.6 Internal Simulation in a Robot?

Could internal simulation be an alternative to the traditional planning approach which presupposes a representational inner world model? The hypothesis investigated in this thesis is that no internal symbolic world model is needed for an autonomous agent to internally simulate sequences of interaction with the world and thus generate its own perceptual input. However, the empirical investigation of this hypothesis will not be an attempt to construct a planning agent using neural networks. Instead, it will be a test to see if appropriate *sequences of sensory consequences of behavior* may be generated through internal simulation within a neural network controller. 'Appropriate' here means that these sequences are such that they conform to the agent's interaction with the environment. They should be sufficient to successfully control overt behavior. If they are, this does not show that the agent can 'plan' in the traditional sense, but instead it would show that it is possible to create a kind of minimal 'inner world' where consequences of actions can be foreseen. This inner world would exist in the robot without explicit implementation of any representations corresponding to objects or relations in the environment. What would be observed is only a self-recurrent system, capable of generating its own perceptual data in accordance with the dynamics in the agent-environment interaction. The internal mechanisms for generation of sensor data do not involve any representations (in the traditional sense) of objects, actions or causal relations. Rather, the mechanisms are merely *tuned to a congruent relationship* with the external world and its dynamics. The primitive behavioral mechanisms are of minimal complexity (i.e. stimulus-response associations). Sensory situations are associated with actions, and also the sensory consequences of those actions. The ANN controller used to implement these mechanisms is similar to that used by Meeden. It is a single network and not a complex multiple module architecture such as those of Tani and Nolfi (1999) or Gross *et al.* (1999). The specific input-output mapping functions are further left to self-organize through learning from interaction with the environment. This investigation of the simulation hypothesis is thus strongly influenced by Occam's razor (cf. Section 1), since it aims to make *minimal assumptions* about the mechanisms necessary to account for the ability to viably control behavior sequences in absence of external stimuli. If the experiments prove successful, the adequacy of these assumptions are supported. On the negative side, no conclusive evidence *against* the hypothesis would be offered if the experiments prove unsuccessful. To achieve this, more experimental evidence is needed, since this matter is intimately connected to another experimental question – if the chosen network architecture has the necessary dynamics to be congruent with the external environment. Bearing these issues in mind, an experimental investigation was made to see if a robot could develop the internal dynamics necessary for internal simulation of interaction with its environment.

3. Experimental investigation

This chapter presents the experiments that were conducted as an initial investigation of the validity of the simulation hypothesis. The following sections present the details on the experimental strategy and methods, the materials used, the experimental conditions and algorithms. After the description of each conducted experiment, the observed and analyzed results are presented. These results will be discussed further in Chapter 4. This empirical investigation should be considered as limited on a number of accounts: First, a highly simplified computational model of an autonomous agent was used in these experiments – it is not a detailed model of any known organism. Second, the design of the agent's control system is merely *drawing on the principal thoughts* laid out in the hypothetical account for complex behavior (cf. Section 2.5). It is not in any real sense a neurophysiologically motivated computational model, rather it is a theoretically motivated model with minimal assumptions about underlying mechanisms. Third, the level of complexity of the control system, the properties of the agent's body, the environment, the agent's behavioral repertoire and the processes of adaptation are kept very low, as in the experiments of Floreano and Mondada (1996), Meeden (1996) and Tani and Nolfi (1999), discussed in Section 2.4. Obviously, real nervous systems, bodies, environments, behavioral repertoires and adaptation processes are often of vastly higher complexity.

3.1 Strategy

Three specific questions were set out to be answered under three successive experimental conditions, using a simulated robot controlled by a neural network with the connection weights adapted using a Genetic Algorithm:

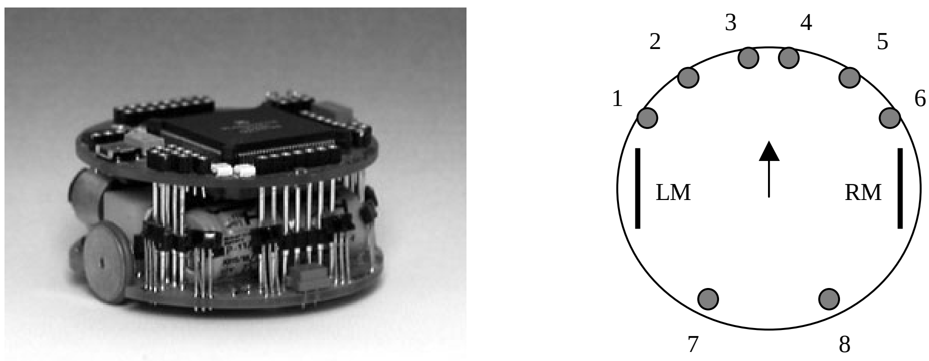
(1) *Can agents with rudimentary navigational abilities be evolved?*
(2) *Can agents able to predict their next sensory states be evolved?*
(3) *Will these agents be able to generate viable sequences of behavior through internal simulation of perception?*

The first step in investigating the hypothesis was to train agents with a navigational task requiring a set of rudimentary sensorimotor abilities. Step (1) could be expected to be successful in light of the looping maze experiments carried out by Floreano and Mondada (1996) and the emergent planning experiments of Meeden (1996) and Meeden *et al.* (1993). The second step was to have agents not only navigate in the environment, but also predict their next sensor state. In step (2), successful prediction abilities could be expected in light of the results of Gross *et al.* (1999), Tani and Nolfi (1999) and to some extent also Meeden *et al.* (1993). The third and final experimental step was to investigate the predicting agents' ability to use their own sensor state predictions for behavior control (i.e. navigation). This experimental condition was thus the one most directly concerned with investigating the simulation hypothesis, but this could not be done without answering the first two questions. Before internal simulation in the agents could be investigated, it had to be confirmed that the agents were able to make viable predictions of their next sensory states, otherwise there would be no point in using them in behavior control during internal simulation. Likewise, for predictions to be meaningful and potentially useful, the appropriate behavioral abilities had to be present. An agent unable to viably move around in its environment (e.g. always standing still or rotating in one place) has nothing useful to predict since the sensor state would be identical at all times or varying only in a trivial fashion. If agents proved capable of making good predictions of sensor states over time (e.g. anticipating and simulating the sensor profiles of a corner) and furthermore able to use these predicted sensor states to control behavior viably (e.g., make a successful turn at that corner) then one may interpret this as evidence supporting the hypothesis that no explicit representational internal world model is necessary for generation of complex (non-reactive) behavior sequences.

In order to be able to predict the next sensor state, two fundamental properties have to be present in the agent. The first required property is a means for creating an agent internal contextual reference for the current sensor reading. This is needed in order for the controller to be able to identify where in a sequence of sensor states the current state belongs. If no such properties were present, then every sensor state would be (from the controller's point-of-view) the first one, so to speak. It would have no relations to any previous sensor states. In order to, for example, be able to predict the appearance of an obstacle at the end of a straight corridor where the agent is currently located, the controller has to be able to (in some sense) identify its current relative position in that corridor, by means of relating to a history of sensor states. The second required property for prediction is to have at least an implicit sense of what action is taken at the moment, so as to be able to appreciate the sensory consequences of that action. The next sensor state depends not only on the current state and the previous states, but also on what action is taken at the current moment. The SRN-architecture (Elman, 1990) used by Meeden (1996) was chosen since it meets both these requirements (cf. Section 2.4.2). The self-recurrent hidden layer allows for internal states that depend both upon a history of exposures to input vectors as well as the previous internal states, thus meeting the first requirement. The hidden node activation vector (i.e. the internal state) further reflects the output activation vector since this vector is a function of the hidden node activation vector, thus meeting the second requirement.

3.2 Materials

The miniature mobile robot Khepera manufactured by K-Team (www.k-team.com) is of cylindrical shape with a diameter of 55mm, a height of 30mm and a weight of 70g (cf. Figure 14). It is equipped with eight infra-red proximity sensors with a range of approximately 50mm. Six of them are distributed in a semicircular way at the front and two are placed at the back. The sensors are capable of detecting ambient infrared light and the reflected infrared light emitted from the robot itself. Depending on how far from a sensor an object is the intensity of reflected infra-red light will vary, as will the activation in that sensor. The two laterally placed wheels are driven by two stepper motors which can rotate the wheels forward or backward independently. The Khepera robot is currently widely used in the autonomous systems research community, due to its for experimental purposes convenient size, its affordability and flexibility. Extra hardware features like sensors, digital cameras or a gripper module can easily be added to the robot, but that was not the case in the experiments documented here (cf. Mondada, Franzi and Ienne, 1993; Miglino, Lund and Nolfi, 1995; Floreano and Mondada, 1996 for further details on extra hardware).



(a) (b)

Figure 14: (a) The Khepera robot (picture from www.k-team.com). (b) Schematic drawing of the robot with IR-sensors (1-8) and left and right motors (LM and RM). Direction of forward motion in the reported experiments is indicated by the arrow.

A major drawback when conducting robot experiments in the real world is the amount of time necessary for agent-environment interaction. As an example, in the battery recharge experiment of Floreano and Mondada (1996), discussed in Section 2.4.1, a population of 100 individuals were evolved for 240 generations and each individual interacted with the environment for approximately 20 seconds. This experiment took a total of 10 days to perform. Real-time experiments soon become unmanageable if one whishes to replicate experiments and vary different parameters. A speed-up of the process is thus necessary and for this purpose a number of robot simulators have been developed. The robot simulator used in this project was a slightly adapted version of the one presented by Miglino *et al.* (1995). The simulator's realism, which derives from the fact that it is based on sensor measurements obtained with a real physical Khepera robot, has been shown sufficient to allow transfer of controllers trained in the simulator to a real robot in a number of papers (e.g. Nolfi, 1997). For this project modifications were made to allow for the designed ANN controller architecture, specifically the addition of a sensor prediction layer and feedback of predicted sensor activations. Modifications of the Genetic Algorithm were also made to allow for a new two-step selection routine. The following sections will describe the details of the designed experimental setup, starting with the environments and learning tasks.

3.3 Environments and Learning

As in the looping maze experiments of Floreano and Mondada (1996) (cf. Section 2.4.1), the behavioral task of the agents in these experiments was to navigate and avoid obstacles in a small environment. Accordingly, a performance measure similar to the one in the looping maze experiments was also used for learning here. For the purpose of investigating prediction and simulation abilities two somewhat more complex environments were designed. The motivation was to enhance variation in sensor sequences, so as to reduce the risk of getting stuck at the suboptimal solution of always predicting the same sensor state. If the world almost always looks the same then it is a good strategy to always make the same prediction. If the world gives varying (but predictable) sensor patterns over time then this should prove a less advantageous strategy.

The two simulated environments (cf. Figure 15), of differing complexity with respect to size and the number of left and right turns, were designed to lessen the risk of specific environment effects on the experimental results. The larger world (from now on referred to as the 'h-world') had the global measurements 400 mm * 400 mm and the smaller one (from now on referred to as the 'T-world') 300 mm * 400 mm. The T-world had five corners to turn - two of 90 degrees and three of 180 degrees. The h-world had six corners to turn - two of 90 degrees and four of 180 degrees. The inner and outer walls were approximately 3cm high and at the corners of the inner walls small round objects with a diameter of 2.5cm were placed.

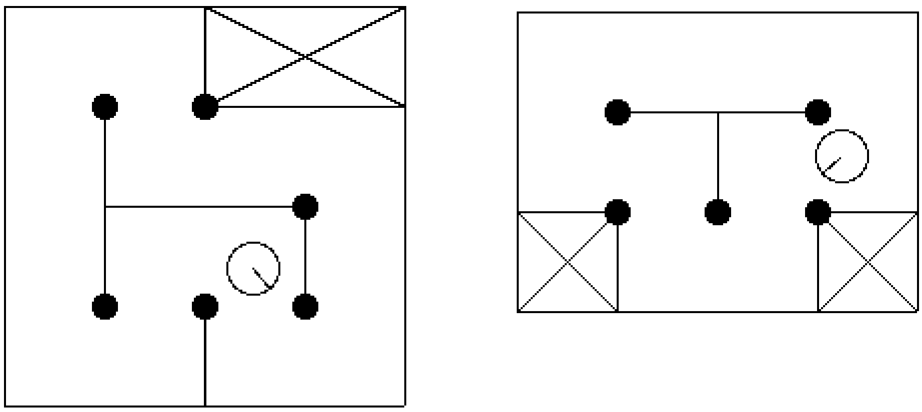


Figure 15: The h-world and the T-world, each with a simulated Khepera robot inside. The robot's heading is indicated by the orientation of the line inside each circle.

A series of three experiments (discussed in Sections 3.4, 3.5 and 3.6), corresponding to the three steps discussed in Section 3.1, was conducted for each of the two environments, making it a total of six experimental conditions. The two first experimental conditions (Experiment 1 and 2) were learning experiments, i.e. agents were supposed to adapt their behavior and predictions to the environment they were trained in (i.e. either the T-world or the h-world). The third experimental condition (Experiment 3) was a test of the behavioral and predictive abilities acquired in Experiment 2.

Training of behavior and prediction was realized by evolving the connection weights of the fixed topology SRN network controllers using a Genetic Algorithm very similar to those used by Nolfi (1997) and Ziemke (1999b). The Genetic Algorithm was designed to run for G generations on a population of P individual weight matrices, where each individual was a set of real valued connection weights in a weight matrix. Connection weights were encoded such that each real value was represented by a binary string of 8 bits. Each weight matrix was thus represented as a sequence with a length of $8W$ bits, where W was the number of connection weights in the network. The length of each bitstring individual was fixed within each experiment, since the number of connection weights was static for each controller architecture. The evolution of weight matrices started with creating an initial population of individuals with random bitstring values. Each bitstring individual was then decoded into a real valued weight matrix which was used to control the behavior of a simulated Khepera robot in its environment. The performance of that individual was then determined quantitatively using a fitness function, specified such that it at each timestep rewarded or punished the behavior by giving positive or negative fitness points to that individual. Each individual thus received a total fitness score after having been evaluated. After all individuals in a generation had been evaluated, the N best (i.e. highest scoring) individuals were selected. These N individuals were then used to create a new generation of the same population size P by making P/N mutated copies of each selected individual. The mutation rate – or the probability of a 'genetic bitflip' – was set to 1.5%. No recombination mechanism (i.e. blending of two parent weight matrices) was used in reproduction, since it has been argued by several authors (e.g. Meeden, 1996) to be more destructive than constructive when evolving neural network connection weights.

The learning experiments (Experiment 1 and 2) consisted of evolutionary runs. A single run consisted in having a population of 150 individuals (i.e. weight matrices) evolve for 500 generations. Each run was replicated ten times, each time using a different seed for randomization of the weight matrices of the controllers in the initial population and the starting point in the environment. One additional replication of each experimental condition was run for 1000 generations to test if any significant improvements were to be expected after the first 500 generations. During learning agents were allowed to interact with their environment for two trials of 500 timesteps (50 seconds) each, each trial starting at a random position. One lap in the smaller environment (T-labyrinth) would take the robot approximately 200 timesteps to complete and one lap in the larger environment (h-labyrinth) would take approximately 250 timesteps. The simulator's sensor readings from the environment provided input for the controller network. The sensor activation values ranged from 0 (object far away) to 1 (object at close range) with uniformly distributed random noise in the range of +/-10% added to all sensor readings. The activation of the output nodes directly controlled the speed and direction of each of the Khepera's two motors. Activation values were in the range from 0 to 1, where 0 initiated full backward rotation, 1 initiated full forward rotation and 0.5 initiated no rotation at all. Each timestep (i.e. the sampling of sensor input from the environment and the duration of action execution) was set to 100ms. A sigmoid unit activation function $f(x) = 1/(1+e^{-x})$ was used for the output units and hidden units, and the connection weights between units were in the range from –10 to 10.

3.4 Experiment 1: Rudimentary Navigation

To answer the first of the three experimental questions (i.e. investigate rudimentary navigational ability) a fixed topology SRN controller was designed (cf. Figure 16) with eight input nodes receiving activation values from the eight IR-sensors, a layer of three self-recurrent hidden nodes and two output nodes controlling the left and right motors respectively. Each layer was fully connected with the next layer.

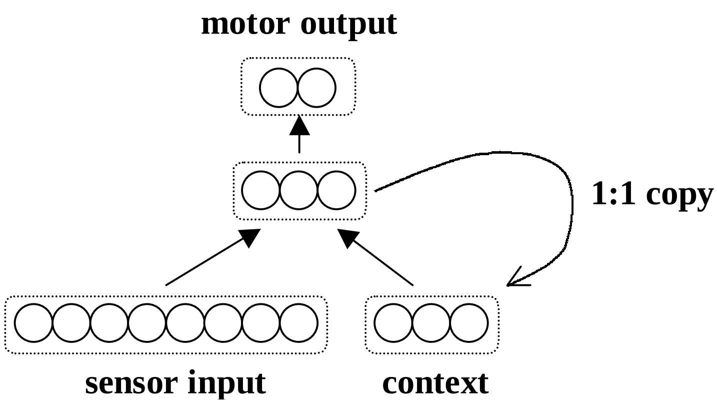


Figure 16: The SRN controller network architecture used in Experiment 1.

The behavioral task, as reflected by the fitness function to maximize, was to perform rudimentary navigation in the environment defined as obstacle avoidance and forward motion as fast a possible. The quality of behavior was determined quantitatively in each time step after performance of an action not resulting in a collision. The fitness function used was defined as:

$$\Phi(a_i(t)) = \Sigma_{(s=1 \text{ to } 1000)}\ 2 * ((v1_s + v2_s - 1) + (1- r_s))$$

where $\Phi(a_i(t))$ is the behavioral fitness value for individual i at generation t, s is the evaluation timestep, $v1_s$ and $v2_s$ are the signed speed values of the left and right motors (between -1 and +1) and r is the activation value of the proximity sensor with the highest activity. The first component ($v1_s + v2_s - 1$) yields a value between –3 and +1, with a positive value for average wheel rotations above 0.5 and negative below this speed. This component is maximized by straight motion at full speed in the positive direction of wheel rotation. The second component ($1- r_s$) yields a value between 0 and 1 and is maximized by obstacle avoidance. To enhance individual differences in the population, the sum of these factors is multiplied by 2, yielding a fitness value between –6 and 4. Fitness was measured at every time step over the two trials (of 500 timesteps each) that the individual was in the environment. Each evaluated individual received a fitness score Φ after evaluation corresponding to the sum of its collected fitness points. After all individuals in a generation had been evaluated, the 30 highest scoring individuals were selected for reproduction. Individual data was collected automatically during learning, including score from the fitness function used in evaluation, average moving speed, maximum moving speed and the number of collisions with obstacles. The different data types were chosen because they each reflect different aspects of the developing abilities during learning. The graphs presented below are based on values averaged over the ten replications of the evolutionary runs.

Behavioral observations of evolved agents confirmed that navigational abilities had developed already within 100 generations in 5 of the 10 evolutionary runs in both environments (cf. Figure 17 for example trajectories). After 500 generations all populations, in both environments, had produced individuals with successful behavior.

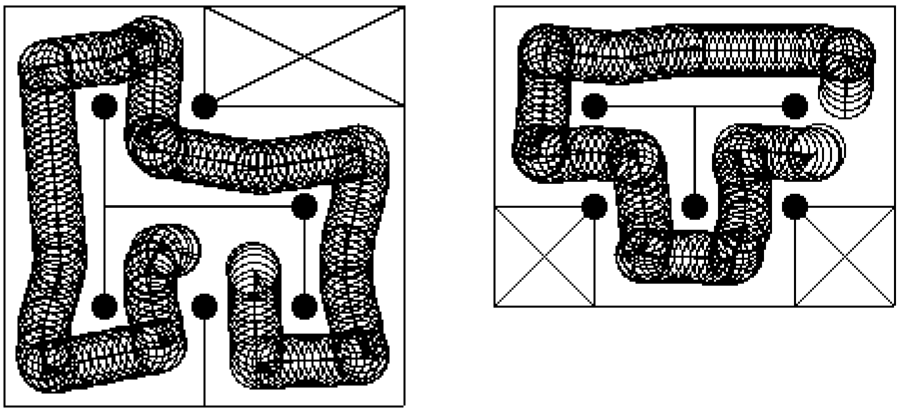


Figure 17: Example trajectories of agents with successful behavior, one in the h-world moving forward counter-clockwise and one in the T-world moving forward clockwise. Both individuals appeared in populations before generation 100. Each circle with a line depicts the Khepera robot's position in one timestep. The robot's heading is indicated by the orientation of the line inside each circle.

Comparisons of behavior fitness development with development of navigational speed and obstacle avoidance (measured as the number of collisions during evaluation) confirmed the validity of the behavior fitness measure. Graphs illustrating the development of these variables are presented in Figure 18.

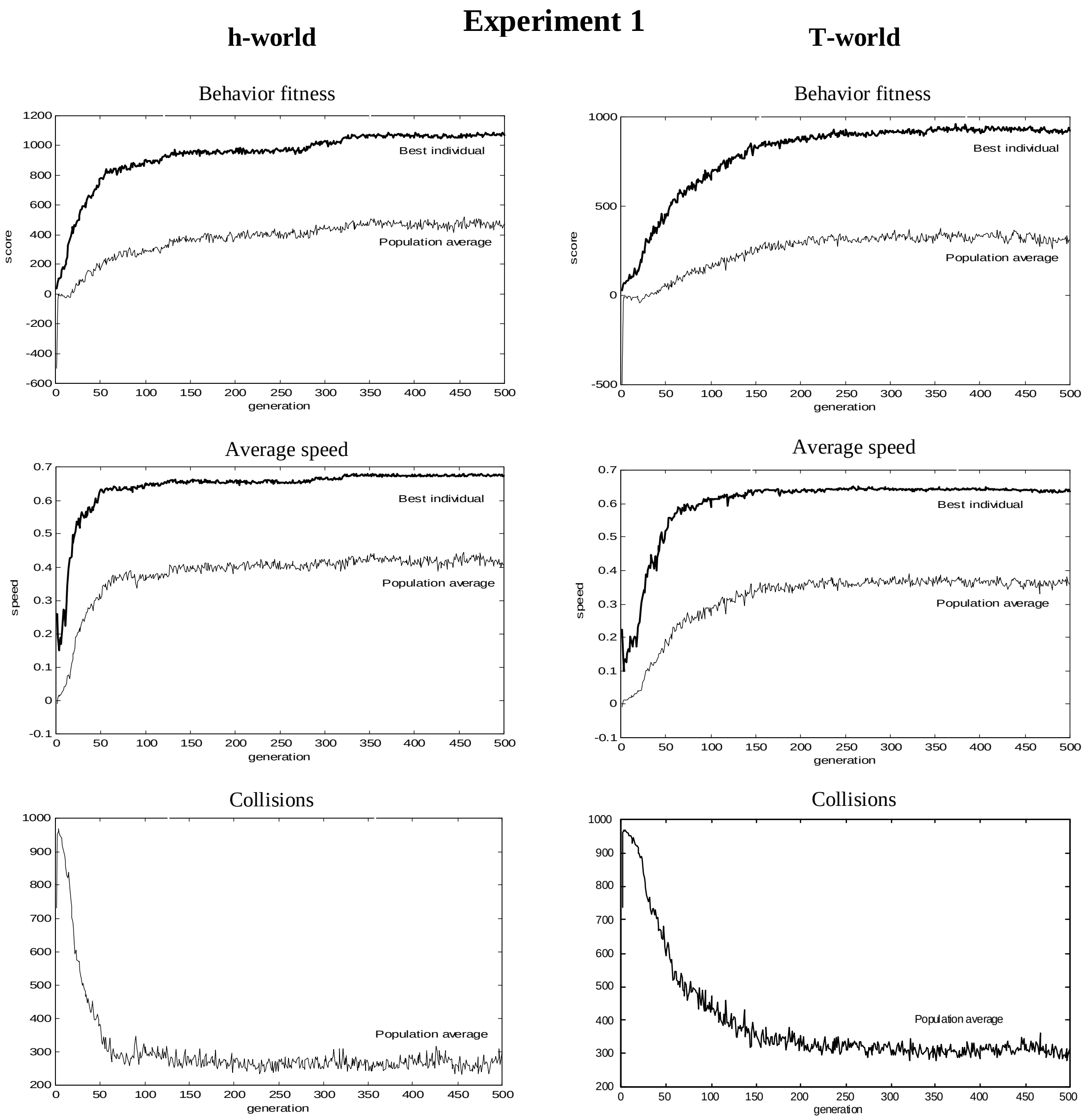


Figure 18: Average development of behavior fitness scores, average moving speeds and number of collisions over 500 generations of evolution. The left column is based on results from the h-world and the right column is based on results from the T-world.

As behavior fitness levels increased the average navigational speed increased and the number of collisions decreased. Individuals navigating perfectly, thus achieving zero collisions, appeared already within one hundred generations and were from thereon reliably present in the following generations. The average number of collisions in the population stabilized at a level of approximately 300 in 1000 timesteps. This may seem unsuccessful, since it could be interpreted as one collision every third timestep on average. However, collisions were detected at every timestep, which means that an individual unable to move away from an obstacle it had collided with *once*, kept collecting collisions until the 1000 timesteps were up. Each individual could thus collect a maximum of 1000 collisions. A population average of 300 collisions could thus be obtained, e.g., in a population where 70% of the individuals navigated *perfectly* and only 30% got stuck. This would not be an improbable case since each generation (except for the initial) consisted of random mutations of the previous generation. Such a case would furthermore be regarded as successful, even though the average number of collisions may seem high, since perfect navigation had been actually achieved by a number of individuals. It was thus impossible to determine whether obstacle avoidance learning was successful from the average number of collisions alone. This matter had to be determined by making behavioral observations. However, the fact that the collision level *decreased over time* indicated that learning did take place in the populations. Learning could be regarded as successful even if only a very small number of individuals capable of perfect navigation reliably appear over a number of replications. The behavioral observations confirmed that such individuals did appear and that navigation learning thus was successful. The successful individuals from both worlds had all received behavior fitness scores of approximately 800 points and negotiated obstacles at average speeds above 60% of full forward speed. Analysis of results from the longer 1000 generation runs confirmed that no further improvements occurred in behavior fitness scores after the initial 500 generations of evolution.

3.5 Experiment 2: Prediction of the Next Sensor State

The second experimental condition was designed to investigate prediction ability. In Meeden *et al.* (1993) agents were trained with a behavioral task and a prediction task in parallel using a backpropagation algorithm. In Meeden (1996) agents were trained with the same behavioral task using an Evolutionary Algorithm, but without the prediction task. In the present study both behavior and prediction were trained using an Evolutionary Algorithm (i.e. the Genetic Algorithm discussed in Section 3.3 above). The controller network architecture from Experiment 1 was augmented with a sensor prediction layer of eight nodes at the output level (cf. Figure 19). This layer was trained with a prediction task where the target output vector (at time t) for these eight nodes was the sensor input vector of the next timestep (at t+1). Training with the prediction task was done in parallel with training on the behavioral task which was the same as in Experiment 1. Performance at both the behavioral task and the prediction task was evaluated in parallel in every timestep, using the behavior fitness function described above and one of the two prediction fitness functions described below.

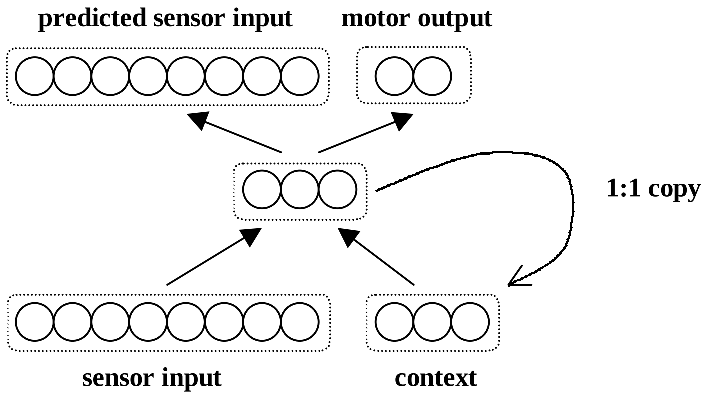


Figure 19: The controller network architecture used in Experiment 2. The original controller network has been augmented with a sensor prediction layer.

Two different prediction fitness functions were designed in order to investigate the effects of specific scoring details on learning. Both functions had the following in common: At each timestep *t* the controller network produced a prediction vector for timestep *t*+1 as well as a motor output. The prediction thus had to accommodate (implicitly) for the action taken at that time in order to be correct. The predicted sensor vector was saved at *t* and evaluated at *t*+1 by comparing it to the current actual sensor readings. When comparing the real and the predicted vectors, two different methods were used for scoring of the prediction quality.

The first prediction fitness function is referred to as 'weighted', since the prediction scoring was weighted with respect to the degree of change in sensor state from one timestep to the next. The weighted fitness function was defined as

$$\tau(a_i(t)) = \Sigma_{(s=1 \text{ to } 1000)} \, (0.1 * \Sigma_{(n=1 \text{ to } 8)} \, (p_{sn} * \Delta s^2_{sn}))$$

where $\tau(a_i(t))$ is the prediction fitness value for individual *i* at generation *t*, *s* is the evaluation timestep. The factor *p* is a measure of the quality of prediction at one node in the layer. The value of *p* is determined from the absolute difference between the target value and the actual value, such that a node activation difference of less than 0.1 units yields the value 5, a node difference of greater than or equal to 0.1 and less than 0.2 yields 4, a node difference of less than 0.3 yields 3, a node difference of less than 0.4 yields 2, a node difference of less than 0.5 yields 1, and a node difference greater than or equal to 0.5 yields 0. This factor is maximized by making correct predictions of node activations. Δs is a measure of change in the sensor state at a node. The value of Δs is defined as the absolute difference of activation at a sensor (node) between the current reading (at time t) and the previous reading (at time t-1). This factor is maximized by large sensor state changes. Its influence is such that predictions in situations of minor sensor state change give small rewards, while predictions in situations of major sensor state changes (if correct) will give large rewards. This factor was raised to the second power to amplify differences between unchanging and changing situations. The total prediction fitness value $\tau(a_i(t))$ is at each time step in the interval between 0 and 4. However, as a consequence of the agent's movements in the environment, the target vectors of the prediction task will vary dynamically with these movements. This means that the best possible fitness value varies from one time step to another, i.e. it is not possible to get the maximum reward (of 4 points) at all timesteps. Also, since the environments and the agents' maneuverability were constrained in space (in order to continuously give sensor activation) and since the sensor sampling interval was short (100ms), the Δs factors above would rarely (if ever) all be equal to 1.

The second prediction fitness function was called 'unweighted', since no weighting with respect to degree of change in sensor state was made. The unweighted fitness function was defined and used as described above, but without the Δs factor:

$$\psi(a_i(t)) = \Sigma_{(s=1 \text{ to } 1000)} (0.1 * \Sigma_{(n=1 \text{ to } 8)}\ p_{sn})$$

A modified version of the Genetic Algorithm, allowing for a two-stage selection procedure, was used. To attain a selection of 30 individuals in each generation, first the 60 individuals with the highest behavior fitness score Φ were selected and from these the 30 individuals with the highest prediction fitness scores τ or ψ were selected for reproduction. These selection parameters were determined experimentally in initial experiments. The modified selection routine was designed with the aim of promoting evolution of prediction ability, but not at the expense of behavioral abilities. A total of four prediction learning experiments were thus made (and replicated ten times), as a result of testing two prediction fitness functions in two environments. Individual data was collected automatically during learning including score from the fitness function used in evaluation, average moving speed, maximum moving speed and the number of collisions with obstacles. The weight matrix of the best individual from each generation (in each replication) was further saved for use in Experiment 3. The graphs presented below are based on values averaged over the ten replications of the evolutionary runs.

Analysis of the results showed that evolution converged to stable levels for both the weighted and the unweighted prediction fitness functions in both environments (cf. Figure 20). The development of the prediction fitness score is also very similar for the two functions, though not numerically comparable due to the effects of weighting versus not weighting rewards. The populations trained with the unweighted function produced some individuals with scores up to 3600 points of the maximum total of 4000 points, i.e. collecting 90% of the possible rewards. The populations trained with the weighted fitness function produced some individuals with scores around 900 points. Due to the weighting, with respect to the magnitude of sensor state change, a meaningful comparison with the maximum reward could not be calculated. Again, analysis of results from the longer 1000 generation runs confirmed that no further improvements occurred in either of the two prediction fitness scores or in the behavior fitness score after the initial 500 generations of evolution.

As can be seen in Figure 20, the behavior fitness development was virtually unaffected by the choice of prediction fitness function in the T-world. Both prediction fitness functions allowed for almost identical development of behavior fitness. For both functions successful behavior was evolved in 9 of the 10 evolutionary runs. In the h-world the unweighted prediction fitness function seemed to allow for slightly better behavior fitness development than with the weighted function. With the unweighted prediction fitness function successful behavior was evolved in 9 of the 10 evolutionary runs, but with the weighted function the success rate was down to 7 out of 10 runs. Behavior fitness development suffered on average in Experiment 2 when compared to Experiment 1 (cf. Figure 18 and Figure 20). Between generations 200 and 500 in Experiment 2 (i.e. after fitness development had slowed down) the best individual behavior fitness levels were between 300 and 400 points below the corresponding levels in Experiment 1 for both environmental conditions. The population average levels had suffered approximately 200 points.

Experiment 2

h-world **T-world**

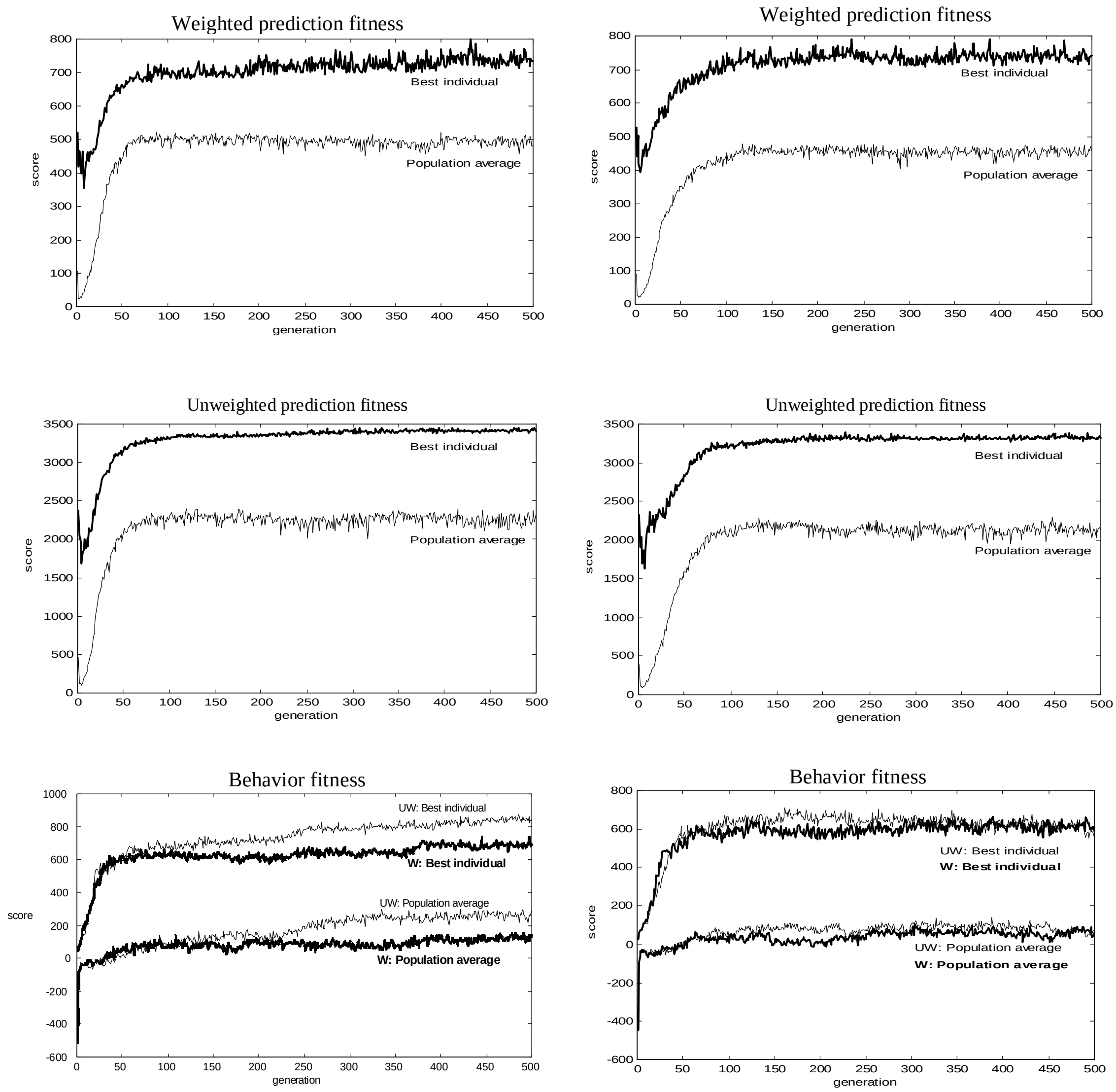


Figure 20: Development of prediction behavior fitness scores over 500 generations of evolution. The left column is based on results from the h-world and the right column is based on results from the T-world. W refers to the weighted prediction fitness function and UW refers to the unweighted prediction fitness function.

The weighted prediction fitness function did not prove to give better prediction abilities than the unweighted, but more importantly it produced worse behavior. This fact was indicated by the development of fitness scores discussed above and was further supported when the 40 best individuals from Experiment 2 were tested in Experiment 3. Their behavioral abilities were generally less developed, often only able to turn either left or right. As a consequence, navigation was often restricted to a smaller portion of the environment. The results presented below are therefor based on analysis of individuals evolved with the unweighted fitness function. Two individuals were chosen for further qualitative analysis. The first individual, depicted in Figure 21, was evolved in the T-world and was found in generation 85 in the first evolutionary run. The second individual, depicted in Figure 22 was evolved in the h-world and was found in generation 323 in the first evolutionary run. They were chosen since they displayed the most interesting observable behaviors during internal simulation when compared to the other individuals in the set of the 40 best predictors from Experiment 2. As both robots' navigation trajectories show in Figures 21a and 22a, their behavioral abilities were not perfectly developed. However, in both cases almost complete

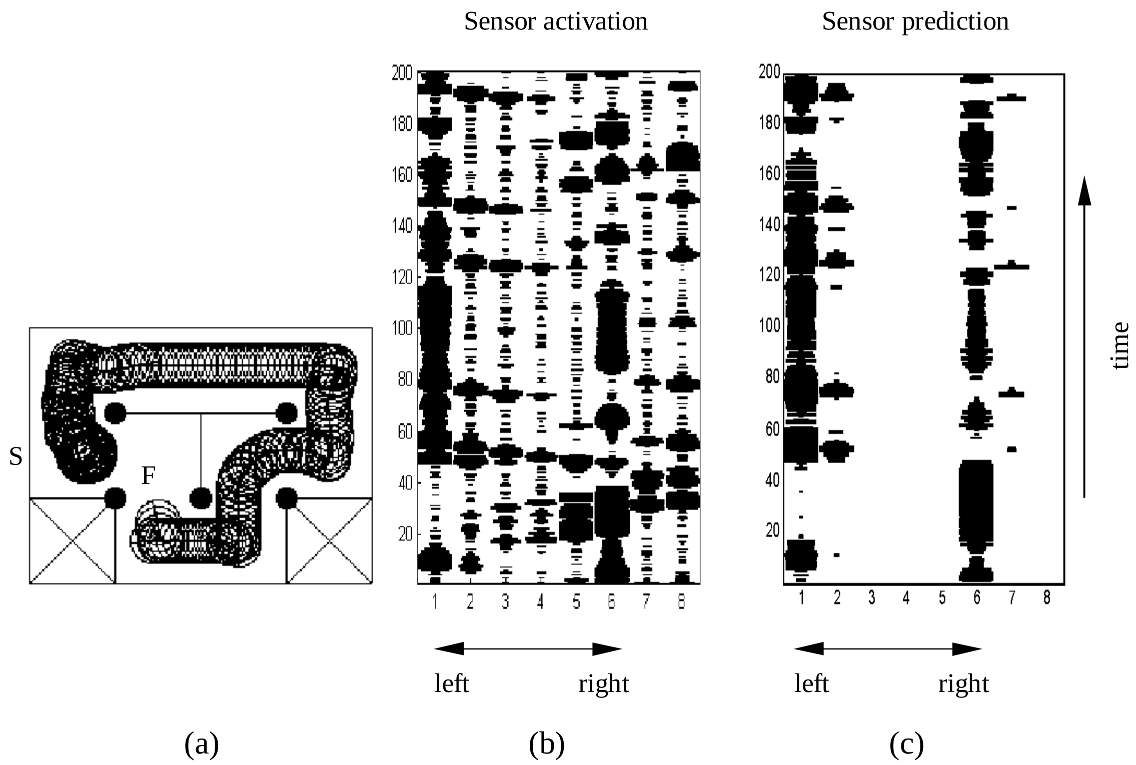


Figure 21: (a) Individual 1's trajectory over 200 timesteps in the T-world, starting at position **S** and finishing at position **F**. (b) The robot's sensor state development over time while navigating. (c) The sensor state predictions made by the agent. Predictions were always made one timestep before the actual sensor activation.

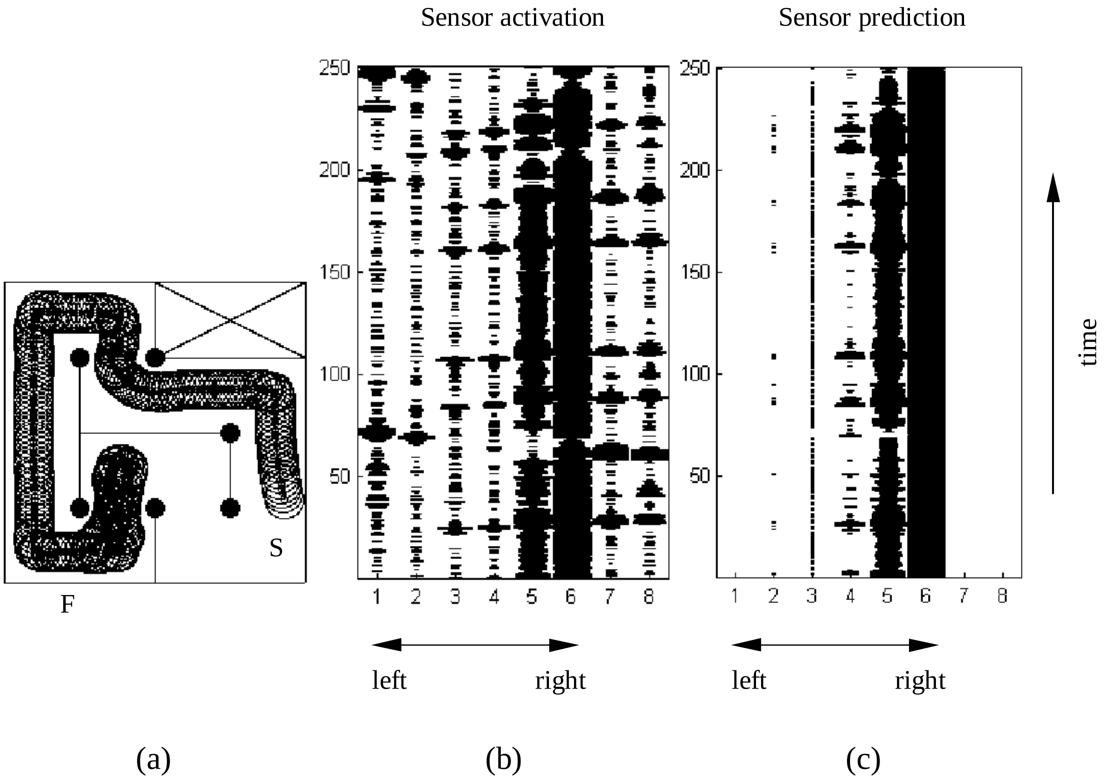


Figure 22: (a) Individual 2's trajectory over 250 timesteps in the h-world, starting at position **S** and finishing at position **F**. (b) The robot's sensor state development over time while navigating. (c) The sensor state predictions made by the agent. Predictions were always made one timestep before the actual sensor activation.

laps could be performed, thus allowing the robots to be exposed to a majority of the relevant sensor situations in their environments.

The logged sensor activations were used to create activation plots, thus giving a description of the world as seen from the perspective of the robot (cf. Figure 21b and Figure 22b). The magnitude of sensor activation is reflected by the width of the black line in the plot. High sensor activation in a timestep is depicted as a wide black line. Plots describing the sensor predictions were also created using the same method. As the sensor plots show, the paths traveled gave rise to continuous and varying sensor activation states. A wall-following strategy can be noticed in individual 2's behavior from looking at the navigation trajectories and the distribution of sensor activations. As the trajectory and the sensor plot show, the robot is always moving close to the wall on its right side giving rise to continuously high activation of the fifth and sixth sensors, both located on the robot's right side (cf. Figure 14). They are kept at an almost constant level, thus keeping the robot at a constant distance away from the wall to avoid collision. When the activation levels drop this indicates to the robot that it has lost contact with the wall which in turn initiates a turning behavior, gradually rotating it clockwise while moving forward to regain contact with the wall. The trajectory and sensor activation plot of individual 1 show signs of an analogous strategy, but with the robot always keeping right in between both walls instead of keeping close to one. This can be seen in the trajectory, but most clearly from looking at the sensor activations. Activation levels were evenly distributed from left to right, due to the equal distance to the walls on both sides. This distribution of activation (and the distance) was sustained by the robot through navigation.

The sensor prediction plots confirm that the two individuals also learned to predict some of the dynamics in the sensor state changes. Most notably, their predictions are strongly related to the adopted behavior strategy. As illustrated in Figure 21c, in the case of individual 1 prediction of sensor activations in sensors one and six (the far right and far left sensors) clearly have been learned with rather high accuracy. The second sensor was also predicted with good accuracy, but only in timesteps when activation levels were high. Prediction of activation in the seventh sensor was only sporadic and rarely correct. No or very little activation was ever predicted in any of the other sensors, thus being incorrect in most timesteps. As illustrated in Figure 22c, in the case of individual 2 the activations of the sensors on the robot's right side were predicted the best. But in comparison with individual 1, sensitivity to small changes in activation levels were less well developed. For instance, activation in sensor six was always predicted to be equal to 1.0, also when it in fact dropped to 0.5 on three occasions (around timesteps 60, 210 and 240). Prediction of sensor five was better with respect to this aspect, since the predicted activation levels corresponded quite well to the actual levels. Just as with individual 1 (in the case of sensor two), prediction of sensor four was correct only when the activation levels were high. Predictions of sensor three were almost constantly around 0.5 and did not vary much in accordance with actual sensor readings, thus making these predictions mostly wrong. No or very little activation was ever predicted in any of the other sensors, thus being incorrect in most timesteps.

To conclude: Sensors that normally were highly activated were often (or always) predicted to be active. Sensors that rarely were highly activated were only rarely (or never) predicted to be active. This could be interpreted as positive results for the prediction experiments. The next section will discuss in further detail to what extent prediction ability was developed and sufficient for internal simulation.

3.6 Experiment 3: Internal Simulation of Perception

The third experimental condition involved no learning. Instead a selection of the best individuals produced in Experiment 2 was used for further testing of internal simulation abilities. A total of 3,000,000 individuals had been evolved and evaluated in Experiment 2 (from 4 combinations of environments and fitness functions in evolutionary runs replicated 10 times, each run using a population size of 150 individuals and lasting for 500 generations). From each of the 40 evolutionary runs, the best predicting individual (i.e. the individual that had received the highest prediction fitness score) was selected for testing, giving a total of 40 individuals for Experiment 3 (10 individuals from each environment and prediction fitness function). They were tested for their internal simulation abilities in the environment they were adapted to, i.e. either the T-world or the h-world.

Each individual was tested using the following routine:

i) Starting from a random position in the environment, the agent was allowed to behave according to its acquired abilities with access to sensor information about the environment for 200 timesteps (in the T-world) or 250 timesteps (in the h-world). This allowed the agents to complete one whole lap, given that they had developed sufficiently good behavioral abilities. This way, an internal context could be built up in the agent from interaction with the environment, giving information useful for making predictions.

ii) After the initial context building phase, the agent was put in a state of internal simulation of perception for 10 timesteps with no access to sensor information about the environment. This was realized by replacing the controller's input vector from sensor readings with the previous sensor prediction vector (cf. Figure 23). All behavior in the environment was thus (for 10 timesteps) based on the agent's own predictions of sensor states. Correct predictions would allow for correct behavior, while incorrect predictions could affect behavior negatively.

iii) After the simulation phase, the agent was once again given information about the surrounding environment from its sensors for 40 timesteps. If the agent had made incorrect predictions – putting it in a bad location in the environment and in an inaccurate internal state – then this time period could set it on the right track again and establish a more correct internal state context.

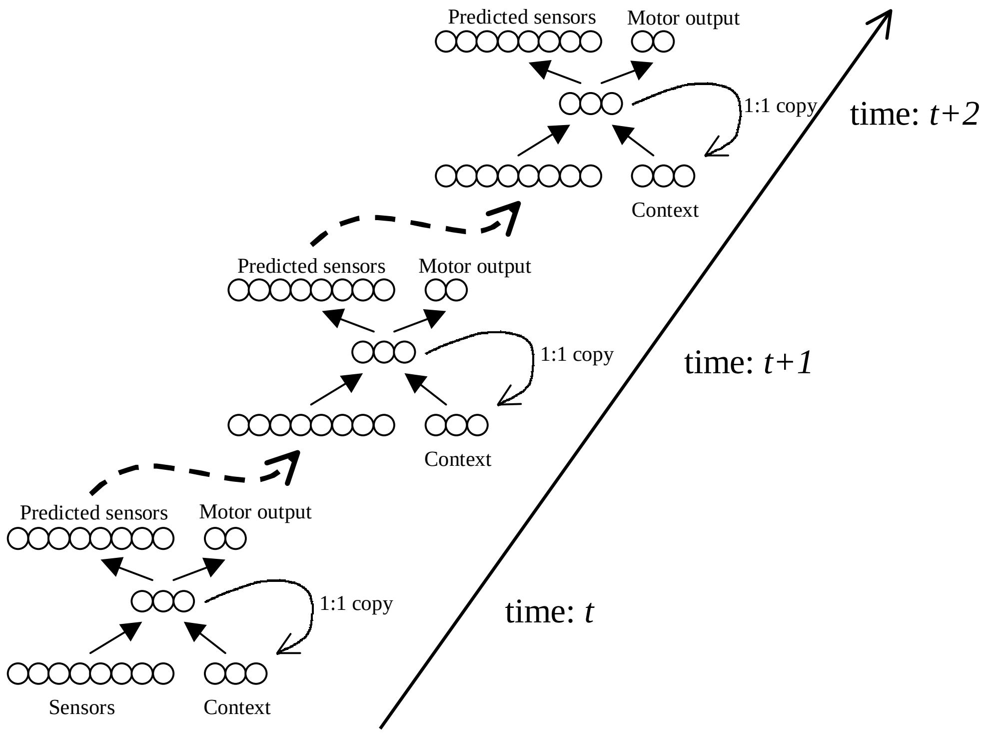


Figure 23: Internal simulation of perception during three timesteps. Instead of real sensor readings the previously predicted sensor activation vector was used as input in each timestep. All behavior during a simulation phase (lasting for 10 timesteps) was thus solely based on the agents predictions about the environment.

After the 40 timesteps of real sensor input, another phase of simulated perception was started (as described in step ii). Switching between real and simulated perception was repeated so that each agent, after the initial context building phase, was tested for a total of five simulation phases.

Simulation ability was determined by making observations of behavior during testing. Data was collected in the form of snapshot picture series of the agent and its trajectory in the environment. Behavior was judged to be viable if it was obstacle avoiding and in compliance with the characteristics of the environment, e.g., if the agent started/continued turning at corners and moved straight in corridors. Additional data such as sensor readings, predicted sensor states, motor activations and hidden node states were also recorded automatically in each timestep during testing. This allowed for comparison of predicted sensor vectors with real sensor vectors and comparisons of external observations of behavior with the internal state changes.

All 40 selected individuals were tested. From this set two representative individuals, previously analyzed in Experiment 2, were chosen for analysis of their internal simulation abilities. Figures 24 and 25 show series of eleven snapshots, taken of Individual 1 and 2 while navigating in their respective environments, along with graphs of activation patterns in the controller networks. In both cases, snapshot 1 shows the context building phase. After that, all odd numbered snapshots show the robots’ behavior during the 40 timesteps where they had access to sensor information and all the even numbered snapshots show the robots’ behavior during internal simulation of perception. Below the picture series are the hidden and output node activation levels in the controller networks presented as graphs divided into segments corresponding to the time periods depicted in the snapshots.

As illustrated in Figure 24, individual 1 managed to avoid collisions in four of the five simulation trials. Only in the third trial (frame 6) did the agent keep moving straight ahead without anticipating the obstacle, thus crashing into it. However, it is clear from the observed behavior in frames 2, 4 and 8 that the agent did not make correct simulations of perception even though it avoided collisions. In frame 2 it is clear that the agent could have moved straight ahead since there were no obstacles in its way. In frame 4 the agent does not anticipate the wall and therefor does not start turning before it actually has come very close to it and just barely avoids colliding with it. In frame 8 it is clear that the agent makes the wrong predictions and starts turning in the opposite direction from the desired one. Only in frame 10 does the internally simulated perception produce behavior which conforms with the situation, making the agent turn left at the corner.

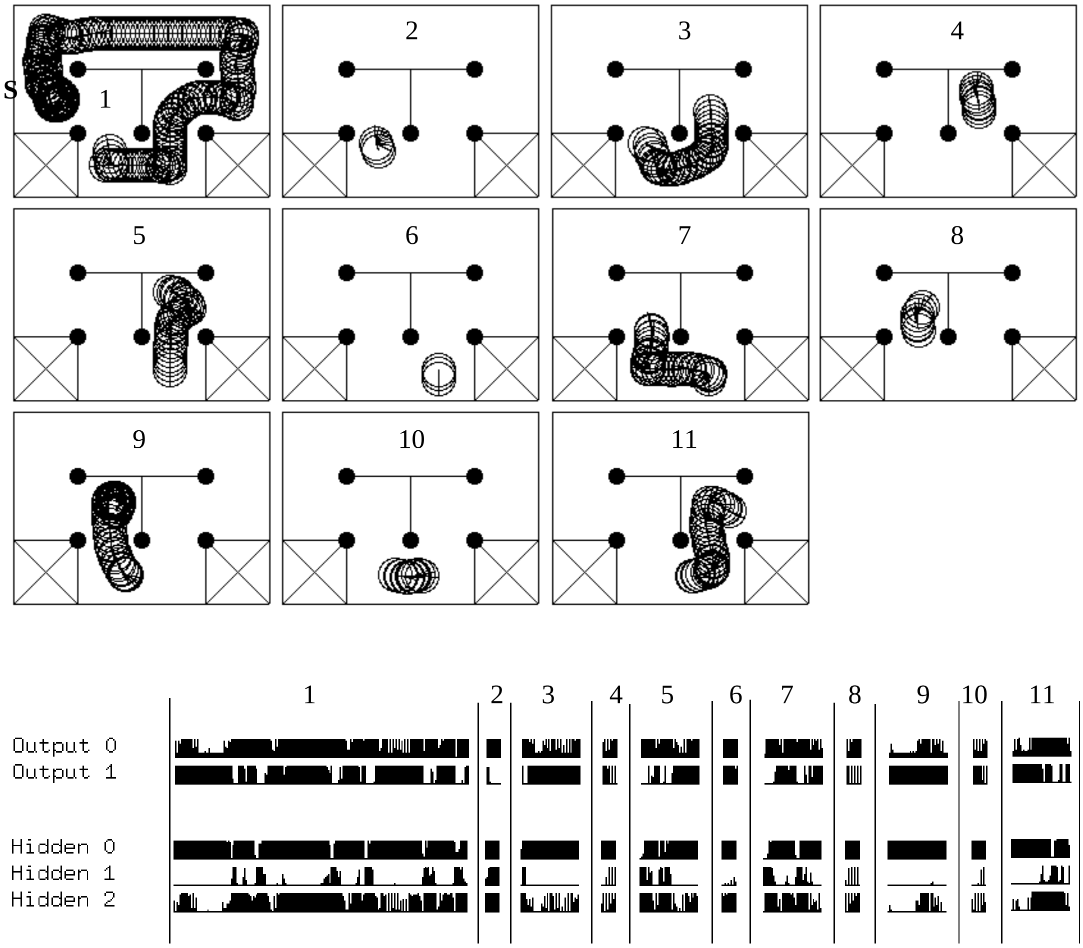


Figure 24: (Top) A snapshot series of individual 1 while navigating in the T-world with access to sensor information (odd numbered frames) first in the context building phase, thereafter in periods of 40 timesteps and also during internal simulation of perception (even numbered frames) in periods of 10 timesteps. (Bottom) The activation levels of hidden and motor output nodes in the controller network divided into segments corresponding to the time periods depicted in the snapshots.

As illustrated in Figure 25, individual 2 managed to avoid collisions in all five simulation trials. In frame 2 the produced turning behavior fits very well with the agent's situation, smoothly turning right at the corner. In the long corridor (cf. frame 6 and 10) the agent navigated straight and correct for the first five or six timesteps, but then initiated a slow right turn behavior almost resulting in collisions with the wall. Finally, in frames 4 and 8 the agent shows continued turning behavior when straight motion would be preferred. From these behavioral observations it was not possible to conclude that the internal simulation abilities were satisfying in general. In fact, these observations indicate the opposite.

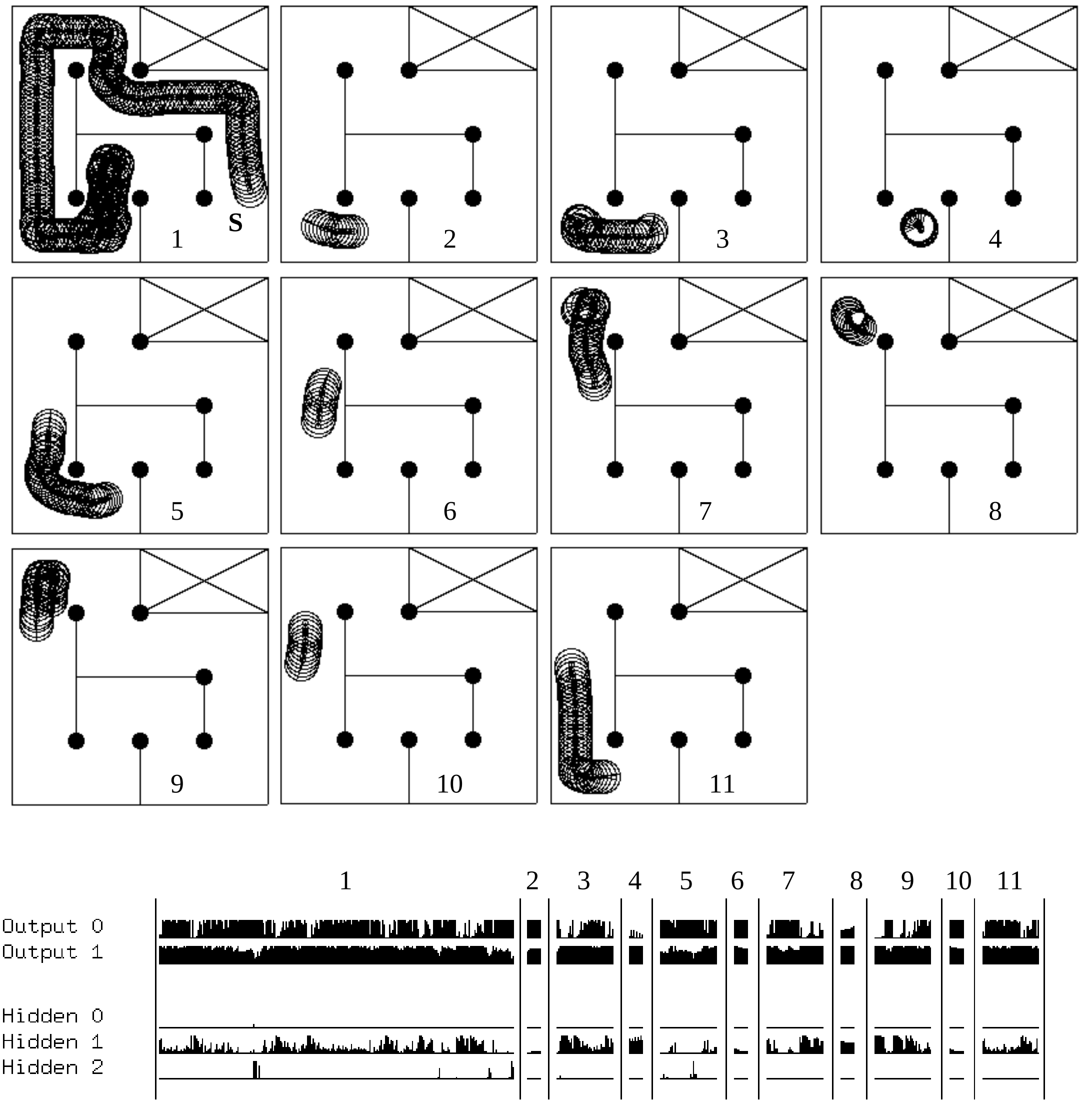


Figure 25: (Top) A snapshot series of individual 2 while navigating in the h-world with access to sensor information (odd numbered frames) first in the context building phase, thereafter in periods of 40 timesteps and also during internal simulation of perception (even numbered frames) in periods of 10 timesteps. (Bottom) The activation levels of hidden and motor output nodes in the controller network divided into segments corresponding to the time periods depicted in the snapshots.

Observation of the internal state changes while simulating perception (cf. Hidden node activations in Figures 24 and 25 corresponding to the evenly numbered frames) reveal no definite lack of internal dynamics in the controller. Changes in activation levels could be found in the second and third hidden nodes in both controllers and changes were observed both in the positive and the negative direction. It is thus clear that the agents do have a capacity to change their internal states during internal simulation. However, as the behavioral observations show, these internal state changes were not sufficient and in some cases even maladapted, thus leading to incorrect motor responses.

3.7 Summary of Experiments

The observations made in Experiment 1 clearly showed that the appropriate rudimentary navigational sensorimotor abilities were developed. Several of the observed individuals were capable of forward motion at more than 60% of full speed, while perfectly avoiding obstacles in both environments. The first experiment thus clearly proved successful and laid a good foundation for the two following experiments.

Training on prediction ability in Experiment 2 proved to be more problematic. No individuals with perfect prediction ability were observed, but the best predicting individuals, as measured quantitatively with the unweighted prediction fitness function, collected approximately 90% of the possible reward points. The qualitative analysis of prediction abilities showed that sensor state predictions were highly selective in the sense that activations in only a few of the eight sensors were predicted. The predicted sensor activations did however conform quite well with the real sensor readings. Thus, the overall pattern of activation in the most significant and informative sensors (specifically the sensors on the robot's left and right sides) had been learned. Variations in 'preference' of sensors to predict also showed a clear correspondence with the behavioral strategy employed to solve the navigation task. For example, Individual 2 which had developed a right-wall-following behavior strategy had also developed a preference to predict activation the right side sensor. This kind of fit between behavior and prediction was probably due to the co-evolution of these abilities, using the two-step selection procedure. A clear negative effect on navigational abilities from the parallel prediction training was however evident. The best individual behavior fitness levels in Experiment 2 were between 300 and 400 points below the corresponding levels in Experiment 1 in both environments. Behavioral abilities were, however, developed well enough to allow the agents to move around in most of the environment.

The 40 best predicting individuals from Experiment 2 were chosen for the third and final experiment which investigated the agents' ability to internally simulate perception through sensor prediction and to use those predictions for behavior control (i.e. navigation). Behavioral observations showed that prediction abilities clearly suffered when sensor input was simulated. The internal dynamics (i.e. the ability to change the internal state) were not sufficiently adapted to reliably produce appropriate behaviors without access to environmental cues via the sensors. Instead agents tended to repeatedly predict the same sensor state during the whole simulation phase. In some cases this was appropriate, as in the case of continuously moving along a straight corridor, but in several cases this was not successful, e.g., when corners appeared and changes of behavior were necessary to avoid collisions. Reliable internal simulation abilities could however not be shown in this experiment.

4. Discussion and Conclusions

The empirical investigation presented in the previous chapter set out to answer three questions (cf. Section 3.1). However, as the results showed after analysis, the empirical data was not always clear in answering these questions. In this chapter the method of empirical investigation and the data collected will be discussed and further elaborated on, including some identified problems that were not anticipated beforehand. Finally, some conclusions will be drawn and some recommendations for future work will be made.

4.1 Behavior

The first set of experiments successfully confirmed the hopes of developing appropriate behavioral abilities in the agents (cf. Section 3.4), so when the details of the prediction learning experiments were worked out, much effort was put into ensuring that development of behavior did not suffer as a consequence. Since prediction of sensor states without the appropriate behavioral abilities would be of little use when the agents were to internally simulate perception, it was important that this problem be solved. One strategy which was considered initially was to train evolved 'well behaving' agents on the prediction task afterwards. This alternative was however (perhaps erroneously) discarded without further empirical investigation. A theoretical argument against this method of two-step learning was that once a network has had its internal dynamics adapted to solve one particular problem of environmental interaction – in this case navigation – the evolved internal state space dynamics would quite probably not be well suited for development of a second problem solving ability – in this case sensor state prediction. The learning algorithm would, in step one, have found an (at least local) optimum for solving the first task, in the space of all possible solutions (i.e. connection weight matrices). In learning a second task, this would mean a non-random starting point for the search algorithm. It was suspected that this would probably be a less advantageous situation than any random starting point, since the new search goal for the algorithm had to be an optimum with respect to, not only the first criterion, but also the second (i.e. both navigation *and* prediction). Such a solution might very well be far from the starting point. Choosing to start at a random position for training on both tasks (i.e. starting 'from scratch') was not thus necessarily better, but unwanted negative consequences from starting at the non-random point could at least be avoided. The practical argument against the two-step learning approach was that a substantial amount of work would have been necessary in order to empirically investigate this matter and to confirm or disprove the suspicion. The arguments (however speculative) thus led to the choice of another approach to solving the problem, where the key ingredient instead was a two-step selection procedure (cf. Section 3.5). This specific method of reinforcement was not previously encountered in the literature, and as such, its virtues or shortcomings had not been validated in other problem domains. Thus, not only the investigation of internal simulation abilities was novel, but also an important aspect of the method of investigation which may have had some unclear impacts on the investigation of the problem domain itself.

The two-step selection method had the attractive quality (at least in theory) of adapting the internal network dynamics to the two tasks simultaneously, instead of having restrictions imposed from the first task upon learning of the second. Agents were reinforced in parallel for their behavioral and prediction abilities, in hope of avoiding the problem of unwanted restrictions on the evolving dynamics of internal state space. This was however not successful. The behavior fitness levels reached in Experiment 2 were substantially lower than in Experiment 1 and the scores of the best individuals in the populations suffered in particular. Behavioral observations further confirmed the quantitative indication of suffering behavioral abilities. The preliminary conclusion which may be drawn is that the negative behavioral effects were consequences of the prediction task training. Unfortunately, it is hard to be more specific, since a total of three technical changes were made with the introduction of the prediction task. First, the network architecture was augmented with eight new output nodes, thus giving 24 new connection weights (not including the eight new bias weights) that had to be adjusted in learning. This meant that the learning algorithm had to search a larger space of possible weight matrix configurations, thus increasing the complexity of search. Second, an additional selection criterion (i.e. the prediction fitness) was introduced and third, a new two-step selection procedure was used. Both of the introduced prediction learning mechanisms interacted with the selection for behavioral abilities. These factors may all have contributed in different degrees to the negative effects on behavior development.

A specific property of the learning algorithm may also have caused problems in general. The mutation routine for random generation of new connection weights may at times have worked coarsely. The probability of mutation (i.e. alteration) of bits coding for connection weights was evenly distributed over the whole length of the bitstring. In these experiments, there was a probability of 1.5% that any one of the bits would be altered. This could be considered a rather small value considering the total bitstring lengths used which was always eight times the number of connection weights. However, whenever a single bit was altered, e.g., from 1 to 0, this may have taken place at any one of the eight locations coding for a connection weight value. This means that one single mutation could change the strength of a connection weight value with anything from 0.4% to

50%. Thus, although weight changes may have been sparse, they may have been great in size. A possible positive effect is that this strategy may avoid getting stuck at local optima in solution space and thus find better solutions otherwise impossible find through gradient descent methods such as backprop. On the negative side, a solution with overall good features may not be selected due to the detrimental effects of one single, but effective, mutation. Thus, the ‘fine tuning’ of network weights may possibly have been disturbed from time to time due to this lack of sensitivity in search.

4.2 Prediction and Internal Simulation

The prediction learning experiments were initially intended to have agents make good predictions in general, but they were soon particularly aimed at producing good predictions in situations when the world changed substantially as a consequence of motor responses, e.g., when turning corners. This proved to be harder than expected and the problem motivated the design of the two environments finally used in the experiments. Both environments had as many turns as possible, both left and right, while still allowing for use of a behavior fitness promoting forward motion. This made turns a statistically well represented feature of the environments, thus increasing the importance of learning to predict their sensory effects. Context sensitivity in the network controller was also early on identified as a necessary prerequisite for making correct predictions, using internal states influenced by a history of sensory situations. Both environments, as seen from the robots’ perspective, essentially consist of sequences of only a few different sensory situations (i.e. straight corridors, left turns and right turns). The sequential character of the task motivated the chosen network architecture (cf. Figure 19). The internal state and context space had to be sufficient to handle the sensor state sequences produced while moving around one lap in each of the environments. Predictions should also receive an appropriate balance between influence from context and influence from the current sensor input. Thus, internal states reflecting the sensory and behavioral history of the agent should influence predictions primarily, but environmental cues, as perceived through sensors, should also play a role. Otherwise sensor predictions would play little role when copied back during internal simulation. An appropriate balance between these factors was necessary for producing correct simulation behavior. If these matters had been resolved successfully it would automatically have led to good simulation ability, since the correctly predicted sensor state, when copied to the input layer, would produce the learned motor response. Given that behavioral responses were learned correctly, context sensitive prediction ability was thus essentially equivalent to good simulation ability.

Prediction abilities initially seemed promising. The plots of predicted sensor states (cf. Figures 21 and 22) showed that significant aspects of actual sensor input were successfully reproduced by agents trained in both worlds. Relations found between behavioral strategies and predictions further indicated that the meaning of ‘significant aspects’ was fundamentally dependent on the specific environmental cues used in obstacle avoidance and navigation. Sensor activation states which were behaviorally important also seemed to be predicted well most of the time. A relation between statistical distribution of occurring sensor states and predictions also seemed present. For example, in one agent (Individual 2) the right side sensor was highly activated most of the time and the agent consequently learned to predict that sensor to be active and it did this always. However, the sensor was not really active at all times. Actually, what mattered most for successful navigation in this agent’s case were those instances in which this sensor was *not* active. Those instances indicated that a corner was reached and that it was time to make a right turn, thus regaining the appropriate right sensor activation level. Since it had almost perfectly learned this relationship behaviorally it did turn when supposed to, at least most of the time. But when it came to prediction, these instances were not handled successfully. Instead it kept predicting the same activation always. This example points to an important characteristic of ANN reinforcement learning – generalization. In behavior learning, an abstract measure of performance was used in the sense that there was no explicit goal motor activation vector (i.e. behavior) for each moment. Rather, behavior was reinforced from the consequences it had on achieving an overall goal, in this case fast forward motion. Generalizations of sensor states were accordingly made by the agents *at this level of reinforcement*. The agents learned to handle situations behaviorally on the basis of their consequences in fulfilling and abstract goal. Contrarily, prediction learning involved explicit goal vectors. There was always one correct prediction to make at every timestep and every correctly predicted detail (i.e. single sensor state) in that vector gave fitness points. In the case of Individual 2 then, since the right side sensor almost always was active, a good strategy, guaranteeing many fitness points over 1000 timesteps, was of course to predict that this sensor would be active. In relation to the total lifetime, only quite few timesteps did *not* involve an active right sensor and they were thus ‘lost in the sensor history’, so to speak. A good general strategy had also been found for prediction, but *at another level of reinforcement*. Unfortunately, this was rather unwanted since it did not serve internal simulation behavior well. Internal simulation depended on the ability to identify such details and when environmental cues were no longer present, predictions failed to be reliably consistent with the changing environment. Most behavior observed during internal simulation tended to be static, in the sense that it was continuously repeated through mere reproduction of the input vector at the prediction level. Also, the internal state was quite static and showed only slight changes in activity from one timestep to another. These observations may be explained by the generalizations made in learning the respective tasks, as discussed above. They may also have been due to an insufficient network architecture, not allowing for development of congruent internal dynamics. The prediction task might also in itself be far more complex than initially perceived, thus demanding vastly more refined technical solutions for successful training. It would be interesting to compare these experimental results on sensor prediction, using an RNN architecture, with other findings on this matter. As mentioned in Section 2.4.2, prediction training was carried out in Meeden *et al.* (1993) but no analysis was presented regarding the quality of those predictions. Similarly, Tani and Nolfi (1999) trained their RNN modules on making predictions (cf. Section 2.4.3) but again, no such results were presented. In both cases prediction training served different purposes compared to the present study, but it would nonetheless have been interesting to see if their RNNs managed to predict anything of value.

4.3 Theory and Future Work

Why did prediction learning and thus internal simulation not work perfectly? One or several of the following factors may have contributed to the results. (1) The network architecture designed for these experiments may have had inherent characteristics limiting the internal dynamics possible to develop in learning. In particular, the internal state space and the contextual influence on predictions may have caused the ‘copying’ of input at the prediction layer. (2) Sensor state prediction might also be a tougher problem than initially perceived, even when dealing with such short time periods as 100 ms. (3) The results might also be traced to problems of global reinforcement. The method used may, as such, have had negative effects on learning of rarely occurring situations due to statistical effects. This may have been the cause of the poor ability to predict changes in rarely occurring situations, i.e. when they were most needed. The problems might indeed depend on more specific details of the method used. (4) Prediction fitness scoring may have wrongly reflected the ‘true’ quality of prediction, or (5) the selection routine may have made incorrect/insensitive selections thus losing potentially successful weight matrix configurations or (6) the weight adaptation routine may have been too coarse and thus unable to ‘fine tune’ on the correct part of solution space even though it may have been found. Because of problems with prediction learning, simulation abilities also suffered. A behavioral repertoire with some occasional correctness was however displayed when engaged in internal simulation and this was due to the *de facto* acquired ability to make predictions, however limited it was. But, as stated before, the observations made do not allow for the conclusion that the experiment was successful, since the abilities were not at all developed to a reliable level.

As was said in the introduction to Chapter 3, the experiments presented in this thesis were only a limited exploration of internal simulation. Only the matter of possible realization of that activity was investigated and that investigation was further limited to the specific case of training an SRN-architecture robot controller using a genetic algorithm. The architecture was kept as minimal as possible, implementing internal simulation simply as a copying of predicted sensor states back to the input layer and feeding the activation forward through the network in the regular fashion. Internal simulation thus automatically produced an overt motor response at each timestep. Internal simulation in this regime was not used to in any sense guide behavior through covert activity, as was implied in the theoretical account. The simplifications, however, allowed for observed behavior to be used as evidence in the analysis of internal simulation abilities. Successfully developed ‘true’ internal simulation abilities could possibly be integrated with the moment-to-moment control of behavior in future investigations. As shown by Gross *et al*. (1999), sensor prediction abilities can be used in neural network control of behavior with positive results. However, as this investigation indicated, the problems of learning to predict future sensor states may require substantial efforts in order to be solved.

For future investigations it might be necessary to consider some of the potential causes of problems identified in this work. Due limited time and space, only some speculations can be offered here: There is possibly a need for more complex network architectures with better abilities to develop internal dynamics congruent with different aspects of the environment. Perhaps several modules are needed, each one congruent with some specific aspect, much like the modular architecture used by Tani and Nolfi (1999) (cf. Section 2.4.3). Learning algorithms better suited for training on prediction may be necessary. It seems necessary to refine the development of an ability to handle a wide range of sensor situations, with less dependence on their statistical distribution, i.e. to handle both the general cases and the exemptions. The problems of finding good solutions may partially be solved by using more sensitive search methods, such as Evolution Strategies (cf., e.g., Rechenberg, 1973 or Bäck, 1996). They may be advantageous since (a) they work on real-valued genotype representations (as opposed to bitstring representations), thus possibly decreasing the risk of losing good solutions, and (b) the reinforcement learning strategy is itself subject to adaptation (as opposed to constant mutation rate and selection pressure), thus possibly increasing fine-tuning abilities once viable solutions appear in the population. Another alternative could be to combine global reinforcement learning with local reinforcement learning, e.g., using both a Genetic Algorithm and a backpropagation algorithm. They could thus be used for the development of suitable initial weight configurations and lifetime fine tuning of these weights respectively. In the light of the success when using abstract behavioral goals, it also seems worth investigating the possibility (and necessity) of stating abstract goals for sensor prediction. A problem is, of course, how one might go about doing this – How does one express an abstract goal for prediction? And is it necessary, since the goal vector at all times may be explicitly supplied?

4.4 Overall Conclusions

Unfortunately, the hypothesis of a non-representational inner world could not be supported with the empirical evidence from this investigation. The results were not conclusive and it is therefor not possible to say anything definite about the possibility of developing internal simulation abilities in RNN controllers. At the moment the simulation hypothesis stands unsupported from AI research data, while it is still attractive, by virtue of the minimal theoretical assumptions made and its support from neuroscientific data (cf. Section 2.5). Evidence from empirical AI investigations might later lend support to the claim that an artificial ‘inner world’, making internal simulation of behavior and perception possible, need not necessarily consist of a representational model. Although the experiments presented in this thesis have not been as successful as was initially hoped for, they might still serve as a useful starting point for later, more detailed investigations. The experimental strategy, although not clearly and strongly supported by the presented data, could still be of some contribution. In particular, the three questions asked in Section 3.1 might serve as an experimental framework for later experiments with other, perhaps less minimal, neural robot control architectures.

As in all fields of research, and particularly in new ones like adaptive neuro-robotics, exploration is essential. Many matters, both practical and theoretical, still remain unexplored in this field. Since there is no better test of theories than empirical data, all experiments may be of potential value in building better theories. But like all scientific explorations they start somewhere in their technical, but more importantly, their theoretical and philosophical assumptions. Too many theoretical assumptions with little or no empirical support may send the exploration off in a wrong direction. The cautious researcher assumes only as little as necessary. Occam’s razor may serve well in the development of theories and explanations, helping to identify those theories which have us assume unnecessary entities and thus only hinder a clearer understanding of the mechanisms which underlie observed phenomena. It is dearly needed in reaching an understanding of the mechanisms underlying behavior, due to the ease with which some entities creep into accounts simply because their existence seems self-evident.

References

Balkenius, C. (1994). *Biological Learning and Artificial Intelligence*. Lund University Cognitive Studies, 30.

Bäck, T. (1996). *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford University Press. New York.

Beer, R. D. (1990). *Intelligence as Adaptive Behavior: An Experiment in Computational Neuroethology*. Academic Press, Inc.

Brooks, R. A. (1991a). *Intelligence without Reason*. In Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (pp. 569-595). Morgan Kaufmann. San Mateo, CA.

Brooks, R. A. (1991b). Intelligence without Representation. In J. Haugeland (ed.) (1997). *Mind Design II*. MIT Press. Cambridge, Massachusetts.

Churchland, P. S. (1986). *Neurophilosophy*. MIT Press.

Churchland, P. M. (1989). *A Neurocomputational Perspective*. MIT Press.

Churchland, P. M. (1995). *The Engine of Reason, The Seat of the Soul*. MIT Press.

Craik, K. J. W. (1943). *The Nature of Explanation*. Cambridge University Press. Cambridge, UK.

Dennet, D. C. (1978). *Brainstorms: Philosophical Essays on Mind and Psychology*. MIT Press. Cambridge, MA.

Dorffner, G. (1997). Radical Connectionism – a neural bottom-up approach to AI. In G. Dorffner (ed.). *Neural Networks and New AI*. ITP.

Elman, J., (1990). Finding structure in time. *Cognitive Science*, 14, (pp. 179-211).

Fikes, R. E. And Nilsson, N. J. (1971). STRIPS: a new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2 (3-4), (pp. 189-208).

Floreano, D., and Mondada, F., (1996). Evolution of Homing Navigation in a Real Mobile Robot. *IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics*, 26 (3), (pp. 396-407).

Franklin, S. P. (1995). *Artificial Minds*. MIT Press.

Gallistel, C. R., (1990). *The Organization of Learning*. Cambridge: MIT Press.

Goodenough, J., McGuire, B. and Wallace, R. (1993). *Perspectives on animal behavior*. John Wiley & Sons, Inc.

Gross, H.-M., Heinze, A., Seiler, T., Stephan, V. (1999). Generative Character of Perception: A Neural Architecture for Sensorimotor Anticipation. *Neural Networks*, 12, (pp. 1101-1129).

Harvey, I. (2000). Robotics: Philosophy of Mind using a Screwdriver. In T. Gomi (ed.). *Evolutionary Robotics: From Intelligent Robots to Artificial Life, Vol. III* (pp. 207-230). AAI Books. Ontario, Canada.

Hesslow, G. (1994). Will Neuroscience Explain Consciousness?. *Journal of Theoretical Biology*, 171, (pp. 29-39).

Hesslow, G. (1996). Hjärnan och själen: kan fysiologin förklara den inre världen? In L. Widén (ed.). *En bok om hjärnan*. Tiden. Stockholm. (in Swedish)

Jeannerod, M. (1994). The representing brain: Neural correlates of motor intention and imagery. *Behavioral and Brain Sciences*, 17, (pp.187-245).

Kosslyn, S. M., Behrmann, M., and Jeannerod, M. (1995). The cognitive neuroscience of mental imagery. *Neurophysiologia*, vol. 33 (11), (pp.1335-1344).

Inder, R. (1996). Planning and Problem Solving. In M. A. Boden (ed.) (1996). *Artificial Intelligence*. Academic Press. San Diego, California.

Lee, D. N., and Thompson, J. A. I. (1982). Vision in action: The control of locomotion. In D. Ingle, M. A. Goodale, and R. J. W. Mansfield (Eds.), *Analysis of visual behavior* (pp.411-436). MIT Press. Cambridge, MA.

Maturana, H. R., and Varela, F. J. (1987). *The Tree of Knowledge*. Shambhala Press. Boston, MA.

McFarland, D. J. (1989). Goals, No-Goals and Own Goals. In A. Montefiore, and D. Noble (eds.). *Goals, No-Goals and Own Goals*. Chapter 4, (pp. 39-57). University Press. Cambridge.

McFarland, D. J., and Bösser, T. (1993). *Intelligent Behavior in Animals and Robots*. MIT Press. Cambridge, Massachusetts.

Miall, R. C., Weir, D. J., Wolpert, D. M., and Stein, J. F. (1993). Is the Cerebellum a Smith Predictor? *Journal of Motor Behavior*, vol. 25 (3), (pp.203-216).

Miall, R. C. (1997). Sequences of sensory predictions. *Behavioral and Brain Sciences*, 20, (pp.258-259).

Miall, R. C., and Wolpert, D. M. (1996). Forward Models for Physiological Motor Control. *Neural Networks*, 9 (8), (pp.1265-1279).

Meeden, L., (1996). An incremental approach to developing intelligent neural network controllers for robots in *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 26 (3), (pp. 474-485).

Meeden, L., McGraw, G., Blank, D., (1993). Emergent Control and Planning in an Autonomous Vehicle in *Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society*.

Miglino, O., Lund, H. H., and Nolfi, S., (1995). Evolving Mobile Robots in Simulated and Real Environments. *Artificial Life*, 2 (4), 417.

Mitchell, M. (1996). *An Introduction to Genetic Algorithms*. MIT Press. Cambridge, MA.

Mondada, F., Franzi, E., and Ienne, P. (1993). Mobile robot miniaturization: A tool for investigating in control algorithms. In *Third International Symposium on Experimental Robotics*, Kyoto, Japan.

Morgan, C. L. (1894). *An Introduction to Comparative Psychology*. Scribner, New York.

Newell, A. & Simon, H. A. (1976). Computer Science as Empirical Inquiry: Symbols and Search. In J. Haugeland (ed.), (1997). *Mind Design II*. MIT Press. Cambridge, Massachusetts.

Nolfi, S., (1997). Using Emergent Modularity to Develop Control Systems for Mobile Robots. *Adaptive Behavior*, 5 (3-4), 343.

Nolfi, S., and Floreano, D. (2000). *Evolutionary Robotics. The Biology, Intelligence, and Technology of Self-organizing Machines*. MIT Press. Cambridge, MA.

Palmer, Stephen E. (1978). Fundamental Aspects of Cognitive Representation. In E. Rosch, and B. B. Lloyd (eds.). *Cognition and Categorization* (pp. 259-303). Lawrence Erlbaum. Hillsdale, NJ.

Plunkett, K. & Elman, J. L., (1997). *Exercises in Rethinking Innateness: A Handbook for Connectionist Simulations*. MIT Press. Cambridge, MA.

Rechenberg, I. (1973). *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Fromman-Holzboog. Stuttgart.

Rey, G. (1997). *Contemporary Philosophy of Mind*. Blackwell Publishers.

Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1986). Learning internal representations by error propagation. In D. Rumelhart, and J. McClelland (eds.). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 1* (pp. 316-362). MIT Press.

Russell, S. J., and Norvig, P. (1995). *Artificial Intelligence: A Modern Approach*. Prentice-Hall.

Sharkey, N. E., and Heemskerk, J. H. (1997). The Neural Mind and the Robot. In A. J. Browne (ed.). *Neural Perspectives on Cognition and Adaptive Robotics*. IOP Press. Bristol, UK.

Sharkey and Ziemke (1998). A consideration of the biological and psychological foundations of autonomous agents. *Connection Science*, vol. 10 (3-4), (pp. 361-391).

Skinner, B. F. (1976). *About Behaviorism*. Random House.

Skinner, B. F. (1987). *Upon Further Reflection*. Prentice-Hall.

Staddon, J. E. R. (1993). *Behaviorism*. Duckworth.

Tani, J. and Nolfi, S., (1999). Learning to perceive the world as articulated: an approach for hierarchical learning in sensory-motor systems. *Neural Networks*, 12, (pp. 1131-1141).

Thach, W. T. (1998). What is the role of the cerebellum in motor learning and cognition? In *Trends in Cognitive Sciences*, vol. 2 (9), (pp. 331-337).

Varela, F., Thompson, E. and Rosch, E. (1991). *The Embodied Mind – Cognitive Science and Human Experience*. MIT Press.

Wolpert, D. M., Miall, R. C., and Kawato, M. (1998). Internal models in the cerebellum. In *Trends in Cognitive Sciences*, vol. 2 (9), (pp. 338-347).

Ziemke, T. (1997). The 'Environmental Puppeteer' Revisited: A Connectionist Perspective on 'Autonomy'. In *Proceedings of the 6th European Workshop on Learning Robots (EWLR-6)*, (pp.100-110). Brighton, UK.

Ziemke, T. (1999a). Rethinking Grounding. In: A. Riegler, M. Peschl, and A. von Stein (eds.). *Understanding Representation in the Cognitive Sciences* (pp. 177-190). Plenum Press. New York.

Ziemke, T. (1999b). Remembering how to behave: Recurrent neural networks for adaptive robot behavior. In L. Medsker and L. C. Jain (eds.). *Recurrent Neural Networks: Design and Applications* (pp. 355-389). CRC Press. Boca Raton.

Ziemke T. (2000). *Situated Neuro-Robotics and Interactive Cognition*. Ph.D. thesis. Department of Computer Science, University of Sheffield, UK.