Behavioral Category Acquisition

Ulaş Türkmen

University of Osnabrück
Institute for Cognitive Science

Erstgutachter: Prof. Kai-Uwe Kühnberger
Zweitgutachter: Roul S. John, MSc.

Abstract

An overview and selective history of the fundamental theory underlying traditional artificial intelligence, namely cognitivism, and an alternative to this approach are presented in this thesis. Major criticisms against cognitivism are explicated in detail, and the main principles of an alternative approach practiced in the last two decades, dubbed New AI, are presented. Among these principles are autonomy, situatedness and embodiment, which are then demonstrated in two embodied models. One of these models is an empirical one of the performance of hens in a visual discrimination task, and the other is a model of categorization, where both models use an exemplar-based categorization mechanism. The thesis concludes with an argument for studying the emergence of linguistic behavior and the acquisition and use of symbols, in order to build successful models of human-level intelligence. It is further argued that embodiment and situatedness are crucial in this endeavor.

Keywords: Categorization, embodied AI, New AI, behavior-based AI, robotics, symbol grounding, situatedness, autonomy, exemplar-based learning

Contents

Chapter 1

Introduction

Cognitive Science is more than a constellation of disciplines. Traditionally seen, it has possessed a very specific research paradigm, one which provided a coherence among the very different disciplines that are supposed to collaborate for one goal. The goal is to get an understanding of the fundamental processes of cognition, and the paradigm is called cognitivism. In this Chapter, a short overview of cognitivism, of how it was actualized in studies in Artificial Intelligence and a review of some of the prominent criticisms directed against cognitivism, especially in the context of Artificial Intelligence, will be given.

1.1 A Short History of Computational Psychology

After the Second World War, a number of technologically sophisticated psychologists and polymaths who worked for the development of better weapons and a better coupling of weapon systems and humans applied their knowledge of technical systems to the study of the mind and behaviour. What these scientists realized was that one could talk about the systems they were working on using intentional vocabulary. They afterwards took the next step and started talking about intentional beings using vocabulary belonging originally to the technical domain: “The new psychology sought to describe human beings using vocabulary that could be metaphorically associated with technologically realizable mathematics” (Agre, 1997a, p.1). One can discern three major strands of such scientific endeavour, although the scientists taking part were in close communication all the time, through conferences, meetings and projects, and they shared many ideas and visions. These strands are cybernetics, information theory, and the digital computer.

During the forming years of cognitive science between 1943 and 1953, the most important forebears of computational psychology were the cybernetics group, most importantly Norbert Wiener. While working on servomechanisms for better control of anti-aircraft artillery, Wiener realized the parallels between self-correcting artificial systems and living beings (Gardner, 1985, p.20). In their seminal article "Behaviour, Purpose, and Teleology", Wiener and his colleagues claimed that it is legitimate to speak of machines that exhibit feedback as "striving towards goals" (Rosenblueth et al., 1943). In this article, the authors described goal-oriented behaviour as movement controlled by negative feedback. One important feature of their work was that they contrasted behaviourism, which they took to study the input-output relationships of a system, with functionalism, which studied rather the internal characteristics of the system it was examining. Although they did not have any problem in principle with a functionalist methodology, the cyberneticians opted for behaviourism in order to be able to apply their methods to both living beings and machines.

Although the cyberneticians fascinated many people, their effect on the psychological community was not immediate, because they were not psychologists themselves. Rather, the effect of cybernetics penetrated into psychology through interpreters, again chiefly psychologists who worked for the design of better devices and better human-machine coupling (see Edwards, 1996, p.209). It was information theory developed by Claude Shannon that had a "clear and precise" influence on psychology (Simon, 1996, p.195). Shannon, an electrical engineer at MIT, "saw that the principles of logic can be used to describe the two states of electromechanical relay switches" (Gardner, 1985, p.25) while he was at the Bell laboratories for a summer job. He later worked at the same laboratories on a speech encipherment system, which was intimately tied up with his seminal contribution, "A Mathematical Theory of Communication" (Shannon, 1948). His theory "concerned itself mostly with measuring information and its transmission over various channels with various capacities and levels of noise" (Harnish, 2002, p.74). Shannon's theory dealt with the transmission aspect of information, rather than the semantic aspect: "Frequently the messages have meaning: that is, they refer to or are correlated according to some system with certain physical or conceptual entities. These semantic aspects of communication are irrelevant to the engineering problem" (Weaver and Shannon, 1949, p.26). Another important aspect of his work was that it allowed one to consider information to be independent of any kind of embodiment or context. He achieved this generality by defining "information" in technically specific and quantitative terms.

The third crucial contribution of wartime research to a computational psychology was the digital computer. Military funded efforts into automatizations of the computations done by humans (called "computers") were going in parallel in the United States and in England. In the USA, analog computers were being used to compute ballistic tables for antiaircraft weapons and artillery, but they weren't fast enough. A faster means of the automation of this process was needed, and this was the *raison d'être* of the ENIAC project (Goldstine, 1972, p.135). John von Neumann was involved in this project towards the end of the war, and he took part in the design of the first stored-program computer, the EDVAC, the first computer in the modern sense according to some sources[1].

[1] E.g. Edwards (1996), but McCorduck (1979) mentions Germany's Konrad Zuse as the

Von Neumann had earlier met Alan Turing, a mathematician working as a part of English efforts to crack the German ciphering of messages. Turing had proved the answer to the third part of Hilbert's *Entscheidungsproblem* to be negative (Turing, 1936)[2]. Just as important as the result of this proof for mathematics was for cognitive science the method he used to prove it. Turing conceived an imaginary machine like a typewriter, called the Turing machine after him. This imaginary machine with an infinite memory tape and a head which could move the tape or change the binary state it was in, was the most general information processing machine, in that it could execute any *effective procedure*, i.e. "a set of rules which tells us, from moment to moment, precisely how to behave." (Minsky, 1967, p.167). The only requirement was that this effective algorithm be expressed in binary code.

At the beginning, Turing's work on the Turing machine had influence in AI only through the work of McCulloch and Pitts, whose neural network was mathematically equivalent to a Turing machine (McCorduck, 1979, p.74)[3]. It was Turing's later ideas on the possibility of a machine that could communicate with a human through a text-only terminal and deceive him about its being a human, that were much more influential (Turing, 1963). Through the so-called *Turing Test*, Turing singled away written communication as the ultimate medium of communication, ignoring all other aspects of human communication like gestures, facial mimics, visual contact, a common context etc. This also was a perfect example of *disembodiment* which AI later on turned into a scientific ideology.

The efforts of the scientists from these three fields laid the foundations for cognitive science, and it was the students of the great names of cybernetics, information theory and digital computer research that initiated cognitive science as an independent research field. Examples of such continuity can be best seen in the works of two groups, with equally strong psychological claims. The first of these is Miller and his colleagues, who gave the first examples of information processing psychology. Miller's work is the first example of cognitive psychology and is a good example of the culmination of a certain discourse and relationship with the practical subject matter in a research program. The second example is the work of Newell and Simon, who are among the progenitors of AI. They transformed a technical field into a space of legitimation for psychological theories. In the following, the main aspects of the principal work by these two groups will be reviewed.

George A. Miller was in the middle of all the exciting technological advances during wartime research, having worked at the PAL (Psycho-Acoustic Laboratory), a World War II institution for experimental psychology. There he was exposed to the progress in cybernetics and information theory, and he later

designer of the first computer.

[2] The third question of Hilbert's *Entscheidungsproblem* is "Is mathematics decidable?", i.e. is there an algorithm to determine the truth of any mathematical expression? The other two questions were whether mathematics is complete and whether it is consistent.

[3] See McCorduck, 1979, p.57 for information on an unpublished paper by Turing which contained many ideas later pursued by Newell and Simon.

spent a summer studying mathematics and digital computation theory with John von Neumann. After moving to MIT, Miller worked on a project involving a huge computer system for air defence (the SAGE project) which kept him in close contact with the most advanced computer technology of his time. His psychological work on acoustics and interaction with the state-of-the-art in cybernetics and computational technology culminated in his famous paper "The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information" (Miller, 1956). Miller's paper was an information theoretic look at the human memory capacity, and is one of the first examples of information-processing psychology. He took the human subject to be a communication channel, and the aim was to find out the "channel capacity of the observer: . . . the greatest amount of information that he can give us about the stimulus on the basis of an absolute judgment" (Miller, 1967, p.38). One of his conclusions was that "the span of absolute judgment and the span of immediate memory impose severe limitations on the amount of information that we are able to receive, process, and remember" (Miller, 1967, p.41). The ideas in this paper can be traced back to his work at the PAL, where the psychologists saw that the performance of humans at the outer limits of their capacities could reveal truths about their performance in ordinary situations. Another influential work which he co-authored was "Plans and the Structure of Behaviour' (Miller et al., 1960). In this book, the authors gave an account of intentional behaviour in terms of goals, plans, knowledge, strategies, and tactics to achieve these goals. One of the main features of their work was that they defined the "cognitive position", and claimed that the people who held the cognitive position "are pretty sure that any correlations between stimulation and response must be mediated by an organized representation of the environment, a system of concepts and relations within which the organism is located. A human being . . . builds up an internal representation, a model of the universe, a schema, a simulacrum, a cognitive map, an image" (Miller et al., 1960, p.7). The unit of analysis they used to explain such representations and models was called a TOTE unit, short for Test-Operate-Test-Exit. These hierarchical units aimed to replace the reflex arc of behaviourism, and were in the same vein with computer algorithms: "a Plan is to an organism essentially the same as a program for a computer"[4] (Miller et al., 1960, p.16). Nevertheless, TOTE units were formulated using the same feedback principle as in cybernetics, and the announced goal of the authors was "to discover whether the cybernetic ideas have any relevance for psychology" (Miller et al., 1960, p.3).

The change from cybernetic models to symbolic models was, along with the coupling of artificial intelligence with cognitive psychology, one of the two most important motives that gave birth to cognitive science. Cyberneticians (and the neurobiologists working in the tradition of McCullough) had a vision of reproducing the functions of the nervous system, leaving it nevertheless inside the body. The target of their studies was the brain as a machine, with input and

[4] Because Miller et al. (1960) use the word *Plan* in a very specific sense, they always spell it with a capital *p*.

output channels defined, and the behaviour specified through combinations of low-level, basic units like reflex arcs. Complex behaviour would then emerge through the interaction with the environment. Cybernetics studied the *embodied mind*, whereas symbolic AI opted for the disembodied, formal, abstract mind: "Instead of modeling brains in computer hardware - the central goal of cybernetics - AI sought to mimic minds in software" (Edwards, 1996, p.239)[5].

The Dartmouth conference is agreed to be the birth date of artificial intelligence as an organized field. Pioneers in the field of AI came together in the summer of 1956 to share their ideas. Among these scientists, two were already presenting the results of their efforts on a program that could prove algebra theorems from Russel and Whitehead's *Principia Mathematica*. These two scientists were Allen Newell and Herbert Simon, who had met at the Systems Research Laboratory at RAND in 1952 and started working on their ideas about simulating human thought with digital computers. They were both impressed by the fact that computers were "more than simply number crunchers and could, in fact, manipulate all manners of symbols" (Gardner, 1985, p.145)[6]. Simon, who had earlier worked on organizational systems, had realized the power of formal methods based on symbols, such as logic, and mentioned the role of such methods in his work on organizations. For him, the role of formal logic in starting off computational psychology was by demonstrating that "manipulating symbols is as concrete as sawing pine boards in a carpentry shop" (Simon, 1996, p.193). However, he had also realized the main drawback of a formalist approach; not every kind of human thinking is logical, and one needs more than deductive logic to study processes such as metaphorical thinking: "Exploiting this new idea in psychology requires enlarging symbol manipulation to embrace much more than deductive logic" (Simon, 1996, p.193). Newell had studied with the renowned mathematician Polya when he was an undergraduate. Polya, who is also the author of a book named *How to Solve It* (1957) on mathematical thinking, introduced Newell to the idea of *heuristics*, simple rules of thumb one uses when solving problems. Heuristics were the major components Newell and Simon used to constrain the search space, and they were also what allowed their systems to cross the border Simon had recognized: the border between formalized symbol manipulation and human behavior which is not always logical, and not always prone to being modelled with precise formalizations. Heuristics were also one aspect of their principle that "artificial intelligence was to borrow from psychology, and psychology from artificial intelligence" (Simon, 1996, p.202). Accordingly, systematic decision procedures of any sort were excluded to simulate as accurately as possible the processes employed by humans.

Newell and Simon collaborated with J. C. Shaw to develop an automatic

[5] The reasons for this shift are documented to be purely intellectual in histories by people from inside the field, such as McCorduck (1979) and Crevier (1996). Some different views can be found in Edwards (1996).

[6] This recognition is stated lucidly by Newell: "The digital-computer field defined computers as machines that manipulated numbers. The great thing was, adherents said, that everything could be encoded into numbers, even instructions. In contrast, the scientists in AI saw computers as machines that manipulated symbols. The great thing was, they said, that everything could be encoded into symbols, even numbers." (Newell, 1983, p.196)

theorem prover, called the Newell-Simon-Shaw Logic Theorist. Shaw and Newell devised the high-level language called IPL-I, a list processing language, which would allow them to write more complex programs than possible using the machine language. The first program they wrote using IPL-II, the successor to IPL-I, was the Logic Theorist, which could prove theorems from *Principia Mathematica*. Newell and Simon found out through thinking-aloud sessions with subjects doing the Moore-Anderson logic task that human subjects were using means-ends analysis as the primary problem-solving tool, and this led to the development of the General Problem Solver (GPS). GPS was a means-ends analysis system, and the computational processes were independent of the particular topic on which it was reasoning. They then published *Human Problem Solving* (1972), which presented their theory of human information processing. The same ideas are presented more succinctly in a paper presented on occasion of the 1975 ACM Turing award, *Computer Science as Empirical Inquiry*. The definition of a *physical-symbol system* found in this paper is at the very heart of the information processing paradigm, and of classical cognitive science:

> A physical-symbol system consists of a set of entities, called symbols, which are physical patterns that can occur as components of another type of entity called an expression (or symbol structure). Thus, a symbol structure is composed of a number of instances (or tokens) of symbols related in some physical way (such as one token being next to another). At any instant of time the system will contain a collection of these symbol structures. . . . A physical-symbol system is a machine that produces through time an evolving collection of symbol structures.(Newell and Simon, 1976, p.109).

The accompanying psychological claim is then the *Physical-Symbol Systems Hypothesis*:

> 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 . . . This is an empirical hypothesis. (Newell and Simon, 1976, p.111). [7]

[7] Harnad (1990) gives a more detailed definition of a physical-symbol system as follows: A symbol system is:

1. a set of arbitrary "physical tokens" – scratches on paper, holes on a tape, events in a digital computer, etc. that are
2. manipulated on the basis of "explicit rules" that are
3. likewise physical tokens and strings of tokens. The rule-governed symbol-token manipulation is based
4. purely on the shape of the symbol tokens (not their "meaning"), i.e., it is purely syntactic, and consists of

The symbols which are manipulated have to be connected to the objects in the outer world if the workings of the machine that manipulates them are to carry any significance for humans, and this is accomplished with *designation*: "An expression designates an object if, given the expression, the system can either affect the object itself or behave in ways depending on the object"(Newell and Simon, 1976, p.110). A popular name given to AI based on these premises by Haugeland (1985) is Good Old Fashioned AI (GOFAI, for short)[8]. Haugeland (1985) points out that there are two claims essential to all GOFAI systems embodying a psychological theory:

1. Our ability to deal with things intelligently is due to our capacity to think about them reasonably (including subconscious thinking).

2. Our capacity to think about things reasonably amounts to a faculty for internal "automatic" symbol manipulation.

1.2 Cognitivism and Its Critique

The two efforts whose outlines were given (Miller's work on *Plans* and Newell and Simon's pioneering work on cognitive modelling) are exemplary cases of cognitivism. As Gardner remarks, one of the main principles of cognitive science, as it was conceived, is a reliance on representations: "Cognitive science is predicated on the belief that it is legitimate – in fact, necessary – to posit a separate level of analysis which can be called the 'level of representation' " (Gardner, 1985, p.38). The symbols used by the cognitive machinery, and the results of the computations made by the machinery, when considered by an outside observer, have meaning, but the physical symbol system does not manipulate these symbols according to their meaning. "Just as is the case in modern logic, it is the *form* of the symbol (or the proposition of which the symbol is a part) and not its meaning that is the basis of its rule-based transformation" (Anderson, 2003, p.95). This is the formalist aspect of cognitivism. In a symbol-manipulating machine like the computer, then, the syntax has to mirror the ascribed semantics, in order for the machine to function properly. The cognitivist claim is then that "this parallelism [between syntax and ascribed semantics] shows us how intelligence and intentionality are physically and mechanically possible" (Varela

5. "rulefully combining" and recombining symbol tokens. There are primitive atomic symbol tokens and

6. composite symbol-token strings. The entire system and all its parts – the atomic tokens, the composite tokens, the syntactic manipulations both actual and possible and the rules – are all

7. "semantically interpretable:" The syntax can be systematically assigned a meaning e.g., as standing for objects, as describing states of affairs.

[8] In the rest of this text, AI work that relies on the physical symbol-system, and that can be characterized by the definition of a GOFAI system will be referred to as either traditional or symbolic AI.

et al., 1993, p.41). Another feature, which follows from representations and formalism is the commitment to explicitly specifiable rules of thought.

One important characteristic of cognitivism is that it foresees computational theories of mind and intelligence. These theories are formulated at the symbolic level, which corresponds to the software level when implemented in a modern computer. This software, however, is independent of the computing machinery on which it is implemented. The central processing units (CPUs) of modern computers are made of semiconductors, but this does not restrict a program running on such a computer from running on another computing system that can carry on the same computations, even though on a different substrate. A radical example is using the whole population of China to build a computer, where individual persons are acting like the gates in a processor made from semiconducting material (Block, 1991).

Cognitivism and AI have attracted a decent amount of criticism, regarding both prevalent technical practices and basic philosophical assumptions. Here a short account of three criticisms will be given. The first of these is the early and fairly controversial one by Hubert L. Dreyfus in his book *What Computers Can't Do*[9], which stems from a report on AI written for RAND. Dreyfus is a philosopher, and his criticism is based on philosophical problems he sees in AI. The authors of the interactivist critique of AI, Agre & Chapman, however, are AI researchers themselves, and aim at a productive reassessment of AI. They have principled arguments against traditional AI, but share some of its methods and concerns. The third critique is the rather practical one by Rodney Brooks, through practical robotic work that contrasts starkly with some of the central methods and assumptions of traditional AI.

1.2.1 Hubert L. Dreyfus and *What Computers Can't Do*

Dreyfus, after reviewing the work done in AI in what can be called its first decade, the years between 1957 and 1967, points to what he sees to be the major problems in machine translation, heuristic search (principally, Simon and Newell's GPS), and pattern recognition. One of the problems he points out with cognitive simulation concerns the practice of using spoken protocols as sources for heuristics in building heuristic-based AI systems. The spoken accounts given by the subject, which are a result of conscious activity, involve search after a certain feature has been recognized. An example for such a feature is a chess player seeing that the rook of the opponent is not defended. The subject starts with this observation and iterates through the possible actions he could undertake. Zeroing in on this crucial feature was unconscious, and if this also involves search, the subject should have been able to give an account of it, or the rest of the search process should also have been unconscious. Dreyfus argues that games like chess involve two stages on the part of the player: zeroing in on an area, and then counting out explicit alternatives. The success of early

[9] Dreyfus wrote his book in 1972, and an updated version was printed in 1993 under the name *What Computers Still Can't Do*. The updated version is used here as the resource.

game-playing programs is then due to work on "those games or parts of games in which heuristically guided counting out is feasible" (Dreyfus, 1993, p.107). Dreyfus also points out that problem solving involves a certain insight, in that the subject perceives what is essential to the solving of the problem: "one breaks away from the surface structure and sees the basic problem –what Wertheimer calls the 'deeper structure'– which enables one to organize the steps necessary for a solution" (Dreyfus, 1993, p.114). One example for such a difference between surface structure and deeper structure is the order of symbols in logic problems: in the expression $P \vee Q$ the order of the symbols is not important, whereas in the expression $P \Rightarrow Q$ it is. In Newell and Simon's programs, this insight is introduced by the programmers in that they choose the most effective formulation of the problem domain, a formulation that effectively includes this level of insight. When using the simulations by Newell & Simon as a support for their theory, Miller et al. (1960) ignore this. According to their theory, the human should first understand a problem, which corresponds to gaining an understanding of the above explained deeper structure. They assume that it is the machine that realizes the first step, using heuristics. Dreyfus concludes that "[o]nly those with faith such as that of Miller et al. could have dismissed the fact that Simon's 'planning method', with its predigesting of the material, poses the problem for computer simulation rather than provides the solution" (Dreyfus, 1993, p.117).

Dreyfus maintains that the cognitivist idea that the human mind functions like a general-purpose symbol-manipulating device relies on the following assumptions about the essence of natural intelligence (Dreyfus, 1993, p.156):

1. A biological assumption that on some level of operation the brain processes information in discrete operations.

2. A psychological assumption that the mind can be viewed as a device operating on bits of information according to formal rules.

3. An epistemological assumption that all knowledge can be formalized, that is, that whatever can be understood can be expressed in terms of logical expressions.

4. An ontological assumption that everything essential to the production of intelligent behavior must in principle be analyzable as a set of situation-free determinate elements.

Dreyfus' treatment of the biological assumption is rather short, because the matter is, as he rightly observes, an empirical one, and our current (as of now as well as of 1972) knowledge of the human neural machinery is insufficient to inquire into such an hypothesis. The psychological assumption, however, is tightly connected with a certain idea of what *explanation* in cognitive science means. As Dreyfus points out, cognitivism is heir to a philosophical tradition that takes as explanation of behavior a set of instructions which can be carried out with as little interpretation as possible. The definition of a behavior in such explicit steps, in addition to constituting an explanation, reveals how the agent actually *produces* that behavior. The problem Dreyfus points out with such a conception of explanation is the level at which it operates. At the physical level, one can not talk e.g. of the sides of a square, because at this level, one has only a certain pattern of energy impinging on the retina. The claim of cognitive science is the existence of a level of explanation between the physical and the phenomenological, and the explanatory rules are then to function at this level. But, Dreyfus points out, when one speaks of these rules and the predicates they are based on (such as a corner, a certain color etc.) they are either at the physical level, or already at the phenomenological, i.e. at the place to which how they arrive has to be scientifically explained.

The epistemological assumption has a subtle, but crucial difference from the psychological one: "those who make the psychological assumption suppose that the rules used in the formalization of behavior are *the very same rules* which produce the behavior, while those who make the epistemological assumption only affirm that non-arbitrary behavior can be formalized according to some rules, and that these rules, whatever they are, can then be used by a computer to reproduce the behavior." (Dreyfus, 1993, p.190). The assumption, therefore, has two parts: (a) that nonarbitrary behavior can be formalized according to some rules, and (b) these rules can then be used by a computer to produce behavior. Against (a), Dreyfus points out that there is an empirical side to the question whether this is so (and it is still open), but also an a priori argument is purported by cognitivists. This argument takes *behavior* not necessarily to be meaningful, and claims that because human beings are physical entities, they should principally be amenable to a law-based simulation on a digital computer, like any other physical entity. This, however, vacuates the cognitivist claim, and runs counter to any hope of establishing a symbolic level of explanation that enables a cognitive scientific practice. If a digital computer is to simulate the human mind at a representational level, it should process the same kind of information, which are symbols.

To argue against (b), Dreyfus contrasts the relationship between a theory of competence and a theory of performance in generative grammar to the relationship between the formal universe of science and the situational world of human beings. As a scientific theory, a theory of competence is a formal theory that goes after timeless and universal rules, but such a theory cannot explain the *use* of a language, because that would require a theory of all human knowledge (which takes us to the ontological assumption). A theory of the use of language, however, if it were to be formulated in the same way like a theory of competence, would have to "deal with phenomena which belong to the situational world of human beings as if these phenomena belonged to the objective formal universe of science" (Dreyfus, 1993, p.201).

The only way left to verify the epistemological assumption would be to hold that the world can be exhaustively analyzed in terms of context-free data or atomic facts. Dreyfus maintains that this argument, the ontological assumption, is "the deepest assumption underlying work in AI and the whole philosophical tradition" (Dreyfus, 1993, p.205)[10]. This is the atomistic, rationalist tradition. Descartes assumed that "all understanding consisted of forming and manipulating appropriate representations, that these representations could be analyzed into primitive elements (*naturas simplices*), and that all phenomena could be understood as complex combinations of these simple elements"; Hobbes claimed that "Reason . . . is nothing but reckoning"; Leibniz "dreamed of reducing reasoning to an algebra of thought" (McCorduck, 1979, p.33). This tradition culminated in Ludwig Wittgenstein's *Tractatus Logico-Philosophicus*, where he provided a concise statement of this syntactic and representational view of the relationship of the mind to the world. In another article on the re-emergence of the connectivist paradigm, Dreyfus and Dreyfus (1988) conclude that

> "AI can be thought of as the attempt to find the primitive elements and logical relations in the subject (man or computer) which mirror the primitive objects and their relations which make up the world. Newell and Simon's physical symbol system hypothesis in effect turns the Wittgensteinian vision –which is itself the culmination of the classical rationalist tradition – into an empirical claim, and bases a research program on it." (p.18).

All this is no new revelation, of course, and this continuity does not pose any problems for symbolic AI as a scientific project. However, Dreyfus and Dreyfus (1988) observe that the atomistic tradition had already undertaken similar projects, although not in the natural sciences but in philosophy, and they did not end particularly happy. The first of these projects is that of Wittgenstein's, who, after writing the *Tractatus*, "spent years doing what he called phenomenology – looking in vain for the atomic facts and basic objects his theory required" (Dreyfus and Dreyfus, 1988, p.26). The second project is Husserl's phenomenology. Both of these projects came under harsh scrutiny, at the time of the first steps of AI, by no other than Wittgenstein himself, and a student of Husserl's, Heidegger. Both of the rationalist projects were based on a precise view of understanding and meaning: "The branch of the philosophical tradition that descends from Socrates through Plato, Descartes, Leibniz, and Kant to conventional AI takes it for granted . . . that understanding a domain consists in having a *theory* of that domain" and such a theory "formulates the relationships among objective, context-free elements . . . in terms of abstract principles" (Dreyfus and Dreyfus, 1988, p.25). Such an approach has been succesful in the natural sciences, and this success has been generalized to all kinds of knowledge. What AI assumed was that natural intelligence should also be based on context-free representations. An important condition for using context-free elements is representing context, i.e. the subset of information received from the environment

[10] Dreyfus gives a definition completely different from this one right on the following page. There the ontological assumption is said to be the assumption that "everything essential to intelligent behavior must in principle be understandable in terms of a set of determinate independent elements" (Dreyfus, 1993, p.206). Clearly, this second formulation hinges on the nature of intelligence, whereas the first one hinges on the nature of reality and how we analyse it. In my opinion, the rest of the text justitifes the first formulation.

that has a potential effect on the content of the representation, explicitly – this is necessary for disambiguating potentially ambiguous terms. If the system processes nothing but formal representations, the situation has to be formally represented too, in terms of symbols of varying degrees of complexity. What an artificial system then has to do is to recognize a context in this situation. The main problem is that the recognition of this context requires another context, because the situation will potentially include a huge number of potential contexts. One encounters an infinite regress, which can be resolved only by positing an ultimate context. It appears that there actually is such an ultimate context for us human beings: common sense. The problem now pertains to the nature of common sense: whether it consists of a set of facts that can actually be amassed using representations (frames, scripts, schemas etc.) of the kind used in computers, or whether it is futile to work on formulating common sense, because it's not even *knowledge* as we know it. According to Dreyfus and Dreyfus (1988), the crucial question is "Can there be a theory of the everyday world as rationalist philosophers have always held? Or is the common sense background rather a combination of skills, practices, discriminations etc., which are not intentional states, and so, *a fortiori*, do not have any representational content to be explicated in terms of elements and rules?" (p.29). The authors are of the second opinion, following the two critiques of rationalist tradition: "As Heidegger and Wittgenstein pointed out,what commonsense *understanding* amounts to might well be *everyday know-how*. By "know-how" we do not mean procedural rules but knowing what to do in a vast number of special cases"(Dreyfus and Dreyfus, 1988, p.33). The way Dreyfus and Dreyfus (1988) paint the picture, we have clear cases of philosophy doing progress, and philosophy culminating in technology: rationalist philosophy made way for symbolic AI, which ignored criticism stemming from the ranks of philosophy, whereas Heidegerrian phenomenalism and the Wittgenstein of *Philosophical Investigations* supplied a criticism of rationalism. The thought-provoking question at this point is whether AI will also experience such a change of methodology.

1.2.2 Philip E. Agre

Despite the philosophical power of Dreyfus' arguments against a viable cognitivist AI, it is sometimes difficult to find in AI programs and the documentation organized around them what one would call a theory of e.g. action or autonomy, which one could evaluate in the light of his arguments. The reason for this is that rather than forming a coherent body of clearly stated hypothesis, AI has embodied its foundational ideas in the programs and texts produced by its practitioners. Agre, who studied AI at MIT, arrived at an original critical perspective, in that he discerned these ideas and how one can dissect them. Agre's critique consists of two parts. The first is a criticism of the theoretical basis of AI. Agre arrives at this criticism by tracing the development of particular ideas and making explicit the connection between these ideas and the impasses to which they have led. The second criticism is about the practical organization of AI work and the relations of these schemes with the theoretical assumptions and justifications.

Agre (1995) points out that AI ideas have their genealogical roots in philosophical ideas and that AI research programs attempt to work out and develop the philosophical systems they inherit[11]. The impasses that are then observed in the technical work of (traditional) AI, such as temporal intractability and inscalability to real life, are the results of these philosophical ideas; the reasons for the persistence of these ideas, and the insistence on regarding the problems caused by them to be of technical nature rather than resulting from internal tensions in the underlying theoretical framework are, as will be summarized below, of a rather social kind. Agre (1997b) claims that the main theoretical groundwork of the cognitivist movement was the philosophy of Descartes, although there were other philosophers that had argued for a mechanistic explanation of the mind, like Hobbes. The reason for the identification with a Cartesian dualism was that later mechanists such as Hobbes and Locke prescribed a certain physical model, while the division of the body and soul prescribed by Descartes provided a freedom as to what kind of a physical realization of human intelligence one pursued:

> Although nobody has mechanized Descartes' specific theory, the stored-program digital computer, along with the theoretical basis of formal language theory and problem-solving search and the philosophical basis of functionalism, provided the pioneers of AI with a vocabulary through which rule-based accounts of cognitive rationality could be rendered mechanical while also being meaningfully treated as mental phenomena, as opposed to physical ones (Agre, 1997b, p.142).

According to Descartes, the mind is sequestered from the body although it interacts with it, and its privileged object of thought is mathematics. An important idea of the cognitivists was that "the mind does not simply contemplate mathematics ... the mind is *itself* mathematical, and the mathematics of mind is precisely a technical specification for the causally explicable operation of the brain" (Agre, 1997a, p.3). The first realization of this aim of mechanization of the mind was the work of Newell and Simon, who were basing their work on search in a space of possible solutions. The practical result was that the more complex the environment became, the bigger grew the search space, which was contained under the name of an *explosion.* Agre (1995) remarks that "The metaphors speak of a struggle of containment between explosion and control. Such a struggle, indeed, seems inherent in any theory for which action is said to result from formal reason conducted by a finite being" (p.15).

Agre and Chapman (1990) distinguish between what they call the *plan-as-program* view[12], and their alternative understanding of a plan, which they call the *plan-as-communication* view. According to the plan-as-program view, the

[11] He also draws an unusual, but in my opinion obviously true, conclusion: "In short, AI is philosophy underneath" (Agre, 1995, p.5).

[12] For examples of such planning systems, see e.g. Fikes and Nilsson (1971), Sacerdoti (1977) and Wilkins (1988). For survey of recent work see Long and Fox (2003).

use of plans is like the execution of a program. Carrying out a plan means walking over the primitive commands included in the plan in a syntactic and mechanical fashion. The generation of the plan and its execution are done by different modules, and once it is produced, the execution of a plan is an unproblematic matter. No or very little additional reasoning is necessary, and sensing the environment is needed solely for the monitoring of the conditions necessary for the correct execution of the plan. Additionally, the plan executor is domain-independent, because all the necessary knowledge about the domain is included in the plan. Agre and Chapman (1990) point to four principle problems that haunt the plan-as-program view:

1. It poses computationally intractable problems.

2. It is inadequate for a world characterized by unpredictable events. If a plan does not anticipate a possible deviation in the environment, its rigid structure will avoid the robot reacting to the respective contingency or making use of the opportunity.

3. It requires that plans be too detailed[13].

4. It fails to address the problem of relating the plan text to the concrete situation. Plans are usually formulated in terms of symbols referring to objects in the environment, and the executing module has to find the connection between the symbols and the objects. This, however, requires in many cases domain knowledge.

In addition to giving a view of the intellectual fundamentals of AI, Agre has observed and criticized the internal workings of the field, i.e. how AI systems are conceived, constructed, discussed and how the researchers react to (especially continually resurfacing) technical problems. This critique is utterly relevant for a deeper understanding of cognitive science, because it opens a perspective on the processes which enable researchers to ignore criticism, and put off practical symptoms of theoretical difficulties.[14] According to Agre, there is a certain way that AI systems are built. Such words as plan or knowledge, although they have precise meanings in particular systems, are vague in an overall sense, in that they enable the researcher to make a wide range of domains commensurable to each other. This is the case with the meaning of *Plan* in Miller et al. (1960): "absolutely any structure or purposivity in anybody's behavior ... can be interpreted as the result of planning" (Agre, 1997b, p.147), and nevertheless, this book is the field's "original textbook in the rhetoric of planning". The construction of a model then works inside out using such basic terms. One takes into consideration a behavioral pattern that exhibits regularity, and tries to arrive

[13] One can read this as saying "you don't need to prove your plans". McDermott (1987) makes the same point: "think of the last time you made a plan ... Chances are you could easily cite ten plausible circumstances under which the plan would *not* work, but you went ahead and adopted it anyway."

[14] For an interesting and more pragmatic look at the practical problems of AI research and possible solutions, see McDermott (1981).

at a set of basic actions which can form the basis of a hierarchical structure. The hierarchical character of such systems was already proposed by Miller et al. (1960). Once a formal hierarchy has been provided, such as TOTE units that come together to make other TOTE units, one has a sufficient repertoire to assemble all the other actions. Then, these basic units are used as terms in formalizing the behaviour in a certain way.

The vagueness of such terms has a number of effects on the AI practice. First of all, although this provides an unbounded generality, in that everything can be seen as a plan under such a vague definition, it precludes any alternative way of seeing things. When plan means any structured and hierarchical process, it is impossible to think of any structured human activity outside the technical language based on plans. Pertinent to this is the second point, the reaction of the AI community to systems that claim to have a different and alternative worldview. This is a result of the engineering aspect of AI, and confines the discussion of different ideas to actually building a system that works. This insistence on practical consistency has installed a deep technical language and led to the evaluation of all proposals "within a tacit system of discursive rules that virtually rules out alternatives from the start" (Agre, 1997b, p.151). Therefore, even when you claim having developed a different perspective, the reaction will be one of unsurprised readiness that reinterprets everything in the already existing language:

> AI's elastic use of language ensures that nothing will seem genuinely new, even if it actually is, while AI's intricate and largely unconscious cultural system ensures that all innovations, no matter how radical the intentions that motivated them, will turn out to be enmeshed with traditional assumptions and practices (Agre, 1997b, p.151).

The third problem is the effect of reducing the meaning of a word to a much simpler technical term when one assimilates the word in a formalization. For example, for *action* this would be reducing it to "a repertoire of possible 'actions' assembled from a discrete, finite vocabulary of 'expressive elements' or 'primitives' " (Agre, 1995, p.13). Once such a concept takes its place in the formalization, the original meaning of this concept and its potential implications as possible resources for ideas are lost for AI. This leads to the result, Agre observes, that "formalization becomes a highly organized form of social forgetting – and not only of the semantics of words but of their historicity as well" (Agre, 1995, p.14). As a result, the history of ideas that find their applications in AI systems are not regarded by AI researchers as worth inquiring.

Another important characteristic of cognitivist talk is the tendency to conflate "representations with what they represent" (Agre, 1997b, p.9). This is obvious in the case of Miller et al. (1960): the book switches between the retrospective description of a behavior by an observer and the purported *Plan* that the organism executes to produce this behavior. What the authors did, according to Agre (1997a), was actually to bring together two ideas: the idea that structured action could be predetermined by mental processing, which originates from Lashley (1951), and the notion that the mental structures could be hierarchical, which came from the work of Newell and Simon on GPS (Agre, 1997a, p.143). This theory of plans led in AI to the above explained *plan-as-program* view. In addition to the above mentioned technical difficulties, there is one peculiarity about the status of plans in AI. Agre (1997a) claims that Miller et al. (1960) contains actually two notions of Plan: one is *the Plan*, that is constructed as one goes through a process, and then there are *Plans* that are pulled out of a repository of Plans and executed as a whole. These two notions are both aimed to be an account of the origins of action, but they are conflated, and hard to distinguish from each other. The aim of Miller et al. (1960)'s theory is to give an answer to how human activity can respond to the limitless changes that take place in the environment and at the same time demonstrate an overall coherence and routine. Agre (1997a) claims that the two notions of Plan aim to answer one of the two contrasting elements in this theory each: "the incremental assembly of the Plan accounted for flexibility in the face of contingencies and the execution of preconstructed Plans accounted for routine organization. Neither theory accounts for both" (p.150). The crucial conclusion has to do with the implications of this inconsistency for the use of plans in AI: the plans-as-program view cannot be a definite doctrine, but a discursive formation, whose practical logic is hidden in the programming work, and in an historical perspective.

1.2.3 Rodney Brooks

Rodney Brooks, an AI researcher at MIT, has argued strongly against the disembodied and mentalist standpoint of traditional AI and cognitivism. In addition to questioning the main tenets of cognitivism and traditional AI, he carried out a research program aimed at creating robots that could operate in realistic dynamic environments. The details of this research program, and similar paradigms of embodied modelling and AI will be dealt with in the next chapter. Here, an overview of Brooks' critique of cognitivist AI will be given.

Brooks criticizes primarily the *abstracting* approach of cognitivist AI. According to Brooks (1991b), AI has abstracted away the subjects it found unsuitable for study, and traditional AI "typically succeeds by defining the parts of the problem that are unsolved as not AI". The subjects that are abstracted away are motor skills and perception. This way, the only possible subjects of study left for AI are world models and problems based on symbolic representations. Information on the environment is to be supplied by the sensory apparatus, and cognitivist AI has accepted the study of how perceptual data is processed to form symbolic representations not to be in its own field of study. However, perception and motor skills are the hard problems actually solved by natural systems. One of the arguments Brooks brings forward for this point is the time it took for the evolution of mobile organisms, and the relatively shorter time it has taken vertebrates and human beings to evolve afterwards. The abstracting method of AI researchers is rationalized with the argument that it is the scien-

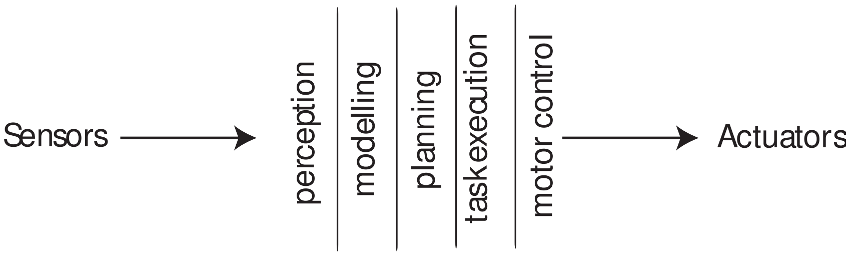

Figure 1.1: The SMPA architecture (adapted from Brooks (1986)).

tific attitude to do abstractions, and this abstraction serves for the computer to experience the same *Merkwelt*[15] as the human being. However, Brooks argues, the *Merkwelt* programmed into the computer is perceived by introspection, and it does not necessarily correspond to the *Merkwelt* that we actually experience. Moreover, each natural and artificial creature with its own sensory apparatus will have its own *Merkwelt*.

Traditional AI has manifested its tendency to draw a clear line between cognition on the one side, the field it has picked for study, and perception and action on the other, in what Brooks calls the SMPA architecture (see Figure 1.1). In this kind of architecture, there is a central system which connects to perceptual modules supplying input to it and action modules which receive commands from it. Because the perceptual modules deliver symbolic inputs and the motor modules receive again symbolic representations, the central system is a symbolic information processor. Brooks (1990) remarks that the perceptual module is expected to deliver "a description of the world in terms of typed, named individuals and their relationships". This assumes a knowable objective truth, which AI implementers try to achieve using modal or non-monotonic logic. This leads to increasingly more complex and cumbersome, and therefore biologically increasingly less plausible models.

Brooks (1991a) gives a history of how the SMPA architecture became accepted and widespread. He argues that during the years following the Dartmouth conference, the disembodied and abstracting method of AI was legitimated through certain demonstrations which were falsely interpreted. In the case of vision, it was expected on the AI side that vision research would one day be able to deliver the representations derived from images that would then enable the agent to construct a model. This belief was supported by systems that could match visual images to pre-stored representations. One example is Roberts (1963), which took a grey level image of the world, and extracted a cartoon-like line drawing. Brooks (1991a) points out that in this case "the lighting was carefully controlled, the blocks were well painted, and the background was chosen with care". Although Roberts' program was working on a very

[15] *Merkwelt*, a concept put forward by von Uexküll (1909), refers to the perceptual world of a living being. See Ziemke and Sharkey (2001) for a survey of the use of this concept in AI and its relevance for embodied AI and Artificial Life.

strictly controlled set of inputs, the result was that people believed that it was possible, at least in the future, to extract a world model from camera images. Another well-known example is Shakey, one of the first embodied AI robots (Nilsson, 1984). Shakey operated in specially prepared rooms, and navigated from room to room, following goals such as pushing an object from one place to another. A planning system named STRIPS was used to construct plans, and this planner used information stored in a symbolic world model. This model was maintained using sensors such as a black and white television camera and bump sensors. Shakey was very successful, because it was among the first demonstrations of integrated mobility, perception, representation, planning, execution and error recovery. However, according to Brooks (1991a), Shakey was carefully engineered to make use of the features of the environment. The surfaces and objects in the environment had uniform and contrasting colors, the space was carefully lighted, and there were relatively few blocks and wedges, in order to avoid partial obscuration. The dark rubber baseboards and the lighter colored floor made clear boundaries, which meant that "very simple and robust vision of trihedral corners between two walls and the floor could be used for relocalizing the robot in order to correct for drift in the robot's odometric measurements" (Brooks, 1991a, p.143). Shakey had an effect similar to Roberts' vision system: although it was a carefully engineered system that could operate in a certain environment and due to this did not provide any chances of being extended to natural and dynamic environments, it bred the idea that one day, through the improvement of the sensing apparatus, one would arrive at systems that can deliver reliable representations of the outside world.

1.2.4 Two problems of AI

Among the problems symbolic AI has ran into, two are well-known and offer insights into the critics summarized above. These are the symbol grounding problem and the frame problem. While the frame problem is a concrete technical problem for which AI researchers have been looking for solutions, the symbol grounding problem is rather theoretical. Both problems, however, are relevant to the discussion, and are summarized here.

1.2.4.1 The Symbol Grounding Problem

The cognitivist assumption was put to question by Searle in his famous chinese room argument (Searle, 1980). He replaced the central processor of a computer with a human being, which receives questions written in Chinese symbols that are totally incomprehensible to her. The human inside the room has a book in her native language that explains her algorithmically how she should respond to the symbols she receives from the outside. The answers she thus composes are then passed on to other people outside the room. These people can understand Chinese and, having seen the symbols that were fed into the room, they interpret the symbols that come out as responses which are meaningful. Searle's argument appeals essentially to the obvious intuition that the human being understands no bit of Chinese just by carrying out the rules. Therefore, a computer cannot understand anything just by running a program, either. Cognition cannot be pure symbol manipulation.[16]

If the meanings of the symbols a symbol system is using are extrinsic to the system, i.e. are a result of external interpretation, rather than being intrinsic, this system cannot be a consistent model of cognition. The meanings of the symbols used in such a system are parasitic to them, and are a result only of our interpretation. This is the symbol grounding problem, which has been defined by (Harnad, 1990, p.335) as follows: "How can the semantic interpretation of a formal symbol system be made intrinsic to the system, rather than just parasitic on the meanings in our heads? How can the meanings of the meaningless symbol tokens, manipulated solely on the basis of their (arbitrary) shapes, be grounded in anything but other meaningless symbols?". In a later paper, Harnad (2003) gives the explicit definition of grounding to be the capacity of symbols "to pick out their referents".

Harnad (1990) points out that the cognitivist solution to the symbol grounding problem, having perceptual modules that ground the symbols to the world outside, is not viable, becuase this "radically underestimates the difficulty of picking out the objects, events and states of affairs in the world that symbols refer to, i.e. it trivializes the symbol grounding problem". Another important aspect to the symbol grounding problem is that in a system which uses grounded symbols, the way the symbols are processed is dependent not only on the arbitrary shape of the symbols, but also, maybe even exclusively, on the content of the percepts on which the symbol is grounded: "In an intrinsically dedicated symbol system there are more constraints on the symbol tokens than merely syntactic ones. Symbols are manipulated not only on the basis of the arbitrary shape of their tokens, but also on the basis of the decidedly nonarbitrary "shape" of the iconic and categorical representation connected to the grounded elementary symbols out of which the higher-order symbols are composed" (Harnad, 1990)[17].

[16] After being subject to criticisms about the radically abstracting nature of the Chinese room argument, Searle has presented it in a more straightforward form:

- *Axiom 1*: Syntax is not sufficient for semantics
- *Axiom 2*: Minds have contents; specifically, they have intentional or semantic contents.
- *Axiom 3*: Computer programs are entirely defined by their formal, or syntactical, structure.
- *Conclusion*: Instantiating a program by itself is never sufficient for having a mind.

[17] Iconic and categorical representations are components in Harnad (1990)'s model of grounding. Iconic representations are "internal analog transforms of the projections of distal objects on our sensory sensory surfaces", and categorical representations are "invariant features of the sensory projection that will reliably distinguish a sensory member of a category from any nonmembers with which it could be confused".

1.2.4.2 The Frame Problem

The frame problem has been identified by McCarthy and Hayes (1969). What it means has caused some difference of opinion between the philosophers and AI scientists, with the AI researchers blaming the philosophers for not understanding what the frame problem actually is[18]. According to Hayes (1987), the frame problem appears when a logical artificial reasoner is made to reason in a changing world, where there are events that have effects on entities, instead of a static world like the worlds of game playing and theorem proving programs. The usual symbolic AI way to model a changing world is to introduce entities to capture the state of the world at a certain time, and to use these entities as temporal indices to distinguish what is true at one time from what is true at another instant. Any properties or relations whose value can change are then treated as relations between their static values and these time indices. Events and actions are then functions from instants to instants, and the effect of any event or action is described in the end state that it produces. Here crops the frame problem: one can specify the consequences of a change, but there is no sure way of specifying the non-consequences. Many properties are defined as being relative to the time instant, because they can change, but if they are not changing, this should be deducible: "whenever something *might* change from one moment to another, we have to find some way of stating that it *doesn't* change whenever *anything* changes. And this seems silly, because almost all changes in fact change very little of the world" (Hayes, 1987, p.125). The frame problem therefore is, stated from a symbolic perspective that looks for the best representational system to model the world, "the problem of finding a representational form permitting a changing, complex world to be efficiently and adequately represented" (Janlert, 1987, p.7).

Although the frame problem at the first glance may look like a technical one, it has been pointed out that it is actually an important philosophical problem, and that it actually is paradigmatic of symbolic AI, in that it is one of those problems that have a characteristic property: "They seem to involve . . . computations that are, in one or other respect, sensitive to the whole belief system" (Fodor, 1983, p.114). Harnad provides a very different view of the problem, however. Computational systems built by AI researchers are underdetermined: just as one instance of a collision of two billiard balls is underdetermined in that there are many theoretical interpretations of it, so can there be many interpretations of a toy domain. Nevertheless, when we are projecting our interpretation onto such a system, we are overinterpreting it, typically by projecting such terms as knowledge, thought and meaning. Consequently, "a 'frame' problem arises every time we run up against evidence that we have exceeded the limits of that underdetermined toy; evidence that we are overinterpreting it – and have been all along" (Harnad, 1993). The problem seems to be with ungrounded symbols that do not have any intrinsic connections with what they are representing. Pylyshyn too realizes this: "It may be that we need

[18] See the contributions in Pylyshyn (1987). It is also interesting how little agreement there is between the philosophers that try to identify the problem.

to get away from logical calculi entirely and use some analog medium of representation for reasoning about change – a medium whose properties ensure that the representation remains faithful to the represented world through a natural harmony of causal correspondence" (Pylyshyn, 1996, p.xiii).

Chapter 2

New AI and Robotics

The creation of intelligent artificial beings has been a dream of human beings since the creation of the first automatons. In this lies of course the difficult question, *what is intelligence?* Cartesian dualism gave a clear answer to this question: whatever distinguishes us humans from animals. Although animals have sensation, they lack thought. Symbolic AI accepted this idea and took disembodied intelligence exhibited e.g. while playing chess and proving mathematical theorems to be prototypical cases of intelligent behavior. This position formed the core of cognitivism, which relied on symbolic representations, formalism and rule-based transformation as the primary tools with which to create artificial intelligence. The problems created by this position and some philosophical arguments against it have been summarized in the first chapter. In this chapter, an alternative approach which was born as a reaction to symbolic AI will be explained.

For symbolic AI, intelligence resides in structures of knowledge in the head (or hard disk, or some other kind of storage medium) of the agent. These structures contain objective knowledge in terms of certain atomic fundamentals, and are evoked when the agent needs them. What structures should take effect and how they will be combined to create new structures is decided by a rational deliberation mechanism. This deliberation mechanism receives the goals and the current state of the agent and objects, events and situations in the environment as symbolic inputs. Intelligence is internal to the agent, just like the world on which this mechanism operates: the world is copied into the brain, and thinking takes place on this copy, instead of the reality outside.

New AI is an umbrella term that covers AI methodologies which attempt to locate intelligence in the interaction between the agent and the environment. Intelligence is intelligent behavior, and this behavior does not necessarily have to be the product of symbolic knowledge and structured, recursive plans in the head of the agent. One important result of the stress on this interaction is the necessity to accept change as given. Intelligent agents are not living in a static environment that does not change until they do something: natural intelligent agents are parts of a dynamic world in which change is the rule rather than the exception. Intelligent agents have to use this dynamics to create intelligent behavior, possibly making use of the features of the environment, external tools and possibilities. Even structures which could let the agent transfer knowledge onto the environment can be a part of these external possibilities (Clark, 1997). An agent is much better off tracking the world through its sensors than modelling it, because the source of intelligence for an agent is the external dynamics rather than internal symbolic structures, and it is costly to model dynamic as compared to static structures. This also removes the problem of accurately projecting the world onto a symbolic representation scheme. In Brooks' words, the world is its own best model.

Due to the fact that the tools offered by symbolic AI for the analysis and design of adaptive dynamic agents were not adequate[1], New AI has turned to other fields for inspiration and tools. One of the main fields from which New AI draws inspiration is biology. In general, New AI shares a definition of intelligence with biology. This definition is based on the ability of an agent to survive, and in order to survive, to adapt to the environment: "The behavior of a system is intelligent to the extent that it maximizes the chances for self-preservation of that system in a particular environment" (Steels, 1994, p.23). Adapting to the environment involves changing the behavior, which is the proof of intelligence. Another source of inspiration is continental philosophy, represented by the likes of Heidegger and Merleau-Ponty, who criticize the tradition of rationalist philosophy and point to the role of the body and the cultural environment in intelligent human behavior.

In New AI, one can distinguish two major directions in the interpretation and utilization of these ideas from different sources. The first of these is a rather liberal AI way of interpreting concepts. This involves using them as sources of inspiration, instead of sticking to a strict and rigid quantitative description. This is *what behavior-based AI* does: behaviors are defined according to what is observed by the observer, rather than data collected from specific experimental control variables. The other is the psychological and biological way of interpretation which involves a more meticulous and quantitative description of concepts and their relationships. In this case one can talk of *biorobotics*. In what follows, the important notions of embodiment, situatedness and autonomy, which are common to both methodologies, will be discussed, and then a short summary of the two different methodologies in New AI will be given.

2.0.1 Autonomy and Agency

Before going on with the discussion, it is necessary to give a definition of what an *autonomous agent* is. According to Russell and Norvig (2003), "An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators" (p.32). Therefore, *automaticity* is attributed to an agent: it is a locus of control, and can take actions

[1] The frame problem can be seen as a symptom of the hostility of traditional methods to the modelling of agents that act in dynamic worlds.

as a result of its decisions. Automaticity involves only a control structure that is not external to the agent – the decisions that the agent has to make should not be made by an external source of intelligence, such as a human being with a remote control.

An autonomous agent possesses, in addition to automaticity, *autonomy*. According to Smithers (1995), "[a]n agent is autonomous if it is able to cope with all the consequences of its actions to which it is subjected while remaining viable as a task-achieving agent in the world it operates in" (p.123). Autonomy is the reliance of an agent on its own experience instead of prior knowledge of the designer. It is not an all-or-none feature: AI has up till now produced partially autonomous agents but has been unable to come up with completely autonomous ones. Autonomy has two aspects, which are related to each other. The first of these is independence from the viewpoint of the designer. Conventionally, the designer partitions the environment an agent is to operate in into categories that she thinks are useful for the agent. This partitioning relies on the concepts the designer utilizes, and does not necessarily correspond to any of the different categorization schemes which the agent could generate, and which could be more advantageous for it. Therefore, an autonomous agent should learn as much as possible of its own categories. The second aspect of autonomy is that an autonomous agent should "learn what it can to compensate for partial or incorrect prior knowledge" (Russell and Norvig, 2003, p.37).[2] This involves the agent's learning from the results of its own actions. Another important point is that an autonomous agent has to be persistent in time in order to learn: for example, a program that is removed from memory once it has completed its interaction with the user can not be counted as autonomous, because it can not evaluate the interaction and use what it has learned in another course of interaction. An autonomous agent therefore has the following properties:

- Sensing: The agent has transducers which enable it to react to certain impulses from the environment.
- Acting: Actuators enable the agent to move in the environment and act on objects.
- Goal directedness: An agent has its own agenda which affects its actions and provides a context.
- Independence from the viewpoint of the designer
- Learning, i.e. improvement of performance through interaction with the environment

2.0.2 Embodiment and Situatedness

Embodiment and situatedness are two concepts that are strongly emphasized by the behavior-based methodology. Embodiment refers to the agent that is being

[2] My favorite informal definition of autonomy is that the only way to make an autonomous agent believe that there is an apple in front of it is to put an apple in front of it.

modelled having a physical body[3]. It also refers to the idea that "intelligence cannot merely exist in the form of an abstract algorithm but requires a physical instantiation, a body" (Pfeifer and Scheier, 1999, p.43).

According to Brooks (1991a), there are two reasons why it is crucial for intelligent systems to have a body. The first is that "only an embodied intelligent agent is fully validated as one that can deal with the real world" (Brooks, 1991a, p.15)[4]. When a simulated environment is used, one faces the additional task of proving that the environment used was realistic enough, and that it did not exclude the necessary features of the real world. If an agent that has actual sensors and actuators is used, all the issues of developing an agent for the real world have to be faced. One question that is also relevant here is *why use robots?* The symbolic approach to AI emphasized operations in an abstract mental space, and therefore focused on systems operating in simulated environments. New AI, however, principally promotes the use of physically embodied agents as experimental devices and physical spaces as operating environments for these agents. The discussion about the advantages of simulated and physical environments is still unresolved, but one argument is worth stressing here. In a simulated environment, the dynamics of the agent-environment couple is modelled as a part of the building the simulation.

The second reason why having a body[5] is crucial for an intelligent agent is that "only through a physical grounding can any internal symbolic or other system find a place to bottom out, and give 'meaning' to the processing going on within the system" (Brooks, 1991a, p.15). This is the *physical grounding hypothesis*, which is as important for embodied AI as the fundamental theorem of calculus is for calculus. In the first chapter, the symbol grounding problem was explained: the representations used by a system without the means to ground its representations in its environment derive their meaning from the meanings attributed to them by the designer. The general idea is that, apart from being a mere theoretical objection to a practically functioning way of building intelligent systems, this also has practical implications. A symbolic approach neglecting the importance of grounding will run into certain problems, because "grounding provides the all-important constraints on representation and inference with which the purely symbolic approach has such trouble" (Anderson, 2003).

Situatedness refers to the agent acting in a world that surrounds it, and there being a constant interaction between the agent and the world. Obviously, it is difficult to make a clear-cut distinction between embodiment and situatedness:

[3] A relatively restricted view of embodiment is taken here. For comparisons of different notions of embodiment, see Ziemke (2001) and Chrisley and Ziemke (2002). The concept of embodiment here refers to what Chrisley and Ziemke (2002) distinguish as *physical embodiment*, which is the requirement that "the realizing physical system be a coherent, integral system, that to some degree persists over time".

[4] Steels (1995a) has a very strong view on the subject: "When we are building robots we are clearly no longer simulating intelligence or making computational models of intelligence, we are building artificial intelligence" (p.93).

[5] It is also worth noting that *having a body* is a remnant of the mentalist system of metaphors; one should read it as *being a body*.

these are closely related concepts that are not mutually exclusive. Anderson (2003) points out one difference between the two concepts: “it is the centrality of the physical grounding project that differentiates research in *embodied* cognition from research in *situated* cognition, although it is obvious that these two research programs are complementary and closely related”. Symbolic AI systems are structured as problem-solving systems: a problem with the initial state is given to the agent or a central mechanism in the agent, and then a solution is produced. This procedure is repeated until a goal state is reached. This is not the case for real agents that act in real worlds. The world is a constant source of sensory impulses, and the coupling between the agent and the world is not severed once the agent believes it has enough information for engaging in a certain action, or that it has found the solution for achieving a goal. For a situated agent, it is also unnecessary to build an internal world model: the world is always there, and it can be referred to again and again as a space for acquiring information from and looking up the results of possible operations in.

2.1 Behavior-Based AI

Behavior-based AI draws inspiration from biology and aspires to build (usually embodied) agents that exhibit intelligent, goal-oriented behavior. It shares some principles with biology, and analyzes intelligent beings at the behavioral level, in contrast to the knowledge level(Newell, 1982), as in symbolic AI, or the physical (i.e. implemantational) level, as in connectionist AI[6]. A behavior is defined as “a regularity observed in the interaction dynamics between the characteristics and processes of a system and the characteristics and processes of an environment” (Steels (1994)). In the following, a number of significant behavior-based approaches will be reviewed. Although it is not strictly behavior-based, the concrete-situated approach is reviewed in this section, because it has a lot to offer for behavior-based AI.

2.1.1 Subsumption Architecture

In the first chapter, a review of Brooks’ criticism has been given. His main point was that AI abstracted away from fundamental aspects of natural intelligence such as perception and action, and declared only processes suitable to being modelled with symbolic processes as proper for AI research. This resulted in the SMPA architecture, where sensing is seen to be delivering perfect representations of the environment in the form of symbolic representations and the motor module is seen to be receiving commands from the central directive. When a robot builder wants to design a system that is to solve a certain problem, he decomposes the problem into a series of *functional units*, as shown in Figure 1.1. Each of these units solves a certain problem and passes on information to the next unit. Brooks (1986) takes a different route and decomposes the problem

[6] For an explanation of this well-known division of cognitive scientific research into three different levels of analysis see Marr (1982).

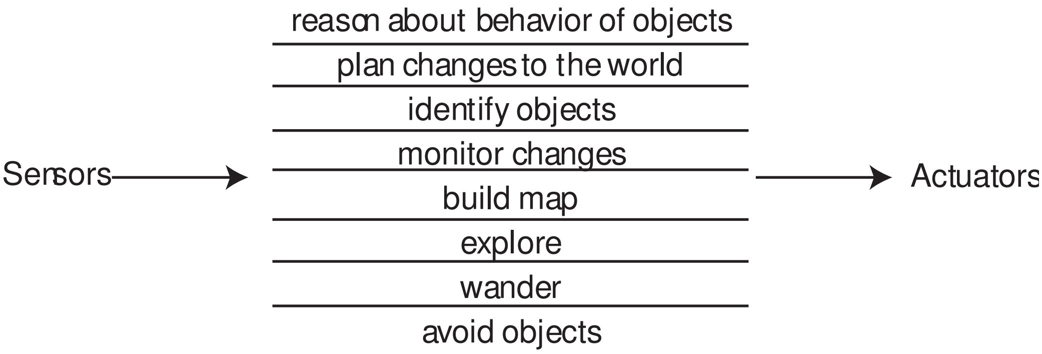

Figure 2.1: Task-level decomposition (adapted from Brooks (1986)).

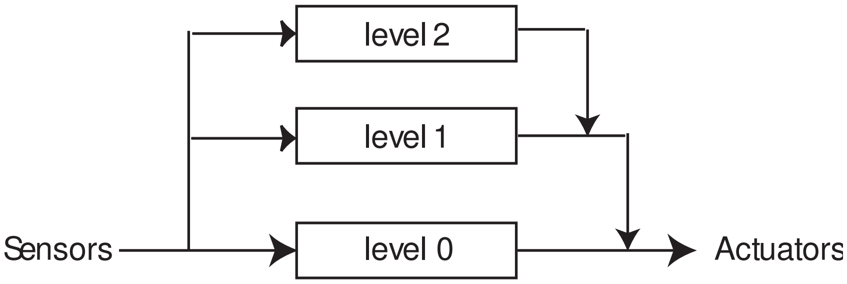

Figure 2.2: The subsumption architecture (adapted from Brooks (1986)).

into *task achieving behaviors*: "Rather than slice the problem on the basis of internal workings of the solution we slice the problem on the basis of external manifestations of the robot control system" (see Figure 2.1).

Brooks has championed an architecture called the *subsumption architecture* to implement task-level decomposition. This architecture is based on behaviors implemented as levels on top of each other. The most basic behavior, in the case of Brooks (1986) avoiding objects, is the *level 0*, and once this level is complete and achieves the designated behavior, it is not changed in later additions to the system. The next level, *level 1*, which in the mentioned study is wandering, is built on top of level 0. It can examine data from the lower level, inject data into the internal interfaces of level 0 suppressing the normal data flow, or prevent a signal from the behavior from reaching the actuators, i.e. inhibit it. Once a level is programmed and is functional, it is not changed any more, and further functionality is built on top of it. The behaviors are implemented as combinations of finite state machines. The communication links are static, and there is no way to access global data. As a result, it is impossible to have a central control unit and a universally accessible world model. The subsumption architecture is illustrated in Figure 2.2.

The subsumption architecture and Brooks' "mobots" (moving robots) are supported by the physical grounding hypothesis. Physical grounding is possible only if the agent has a physical body that carries sensors, and if it is situated in a physical environment. Brooks ties situatedness to the idea of a robot continuously referring to the world instead of an internal model, an idea he sloganizes as "The world is its own best model" (Brooks, 1991a). Functioning in the real world is not enough, however: this world should not be a simplified one, where certain properties are generalized so as to provide cues about a universal property such as the location of the robot. With a simplified world it is very easy, even with the best of intentions, to build a submodule (a level) that relies on a simplified property in the environment, like a sharp contrast in the colors of the ground and the walls (Brooks, 1991b). Once one module relies on this property, the functioning of the whole system depends on this one single property. After moving on to the real world, one realizes that the system has to be rebuilt. In the case of Brooks' own robots, the real world corresponds to the unmodified worlds found around their laboratory, with the people working in the environment allowed to walk around, and environmental conditions such as lighting allowed to change.

The subsumption architecture has been implemented and tested on a number of robots and in a number of tasks, such as navigation (Brooks, 1986), chasing moving objects (Connell, 1987)[7], wandering around in office areas and collecting empty soda cans (Connell, 1989), walking (Brooks, 1989), map building and use (Mataric, 1989) etc. (see Brooks (1990) for a short summary of the robots built until 1990). The general framework has been extended to a humanoid robot named Cog to examine human-level intelligence (Brooks et al., 1999; Brooks and Stein, 1994). The robot consist of an immobile torso with 21 degrees of freedom. Lower level control is done with the subsumption architecture, but the higher level computations also include symbolic processes, where symbolic binding is restricted to within individual processors. Among the phenomena Brooks aims to model are social interaction and development. This causes an even heavier reliance on embodiment and situatedness. The humanoid robot is to learn human-level interaction by interacting with humans, which requires that the robot have a realistic humanoid body (Brooks et al., 1999).

2.1.2 The Concrete-Situated Approach

In the first chapter, the arguments of Philip E. Agre and David Chapman against some prevalent notions of cognitivist AI, especially the role of planning, were presented. Agre, in individual efforts as well as in collaboration with David Chapman, has proposed and to some extent realized a different approach to intelligent agency. Their approach, stated in most detail in Agre (1997a), draws its inspirations from continental philosophy and ethnomethodology. There are

[7] The robots Tom and Jerry explained in this article also served to demonstrate how little raw computation is necessary for the subsumption architecture: the whole program fitted on a single 256 gate programmable array logic chip.

two aspects to this approach, which one could call *the concrete-situated approach.* The first is that it aims to develop, in contrast to a theory of cognition or a theory of thinking, a theory of activity, which should answer the question *what determines an agent's actions?* The second is that it is interactionist. The concrete-situated approach has two interconstraining parts: a theory of cognitive machinery and a theory of the dynamics of activity.

Interactionism, as Agre (1997a) defines it, is an alternative set of metaphors that are to replace the set of metaphors that form the scientific program of mentalism. The interactionist methodology is concerned with the relationship between the machinery and dynamics: its fundamental question is "what kinds of machinery get into what kinds of dynamics in what kinds of environments?" (Agre, 1997a, p.61). The most important principle of the interactionist methodology is *machinery parsimony*: "postulate the simplest machinery that is consistent with the known dynamics" (Agre, 1997a, p.62). This means that instead of looking for novel machinery, we should look for novel dynamic effects.

Against the problem-solving approach of the established AI practice, Agre stresses the role of routines in human life. Routines are "the frequently repeated and phenomenologically automatic rituals of which most of daily life is made" (Agre, 1985). A routine can consist of a series of actions, without the order of the actions being determined at the beginning, but rather each consecutive action being chosen as the result of a new process of decision. The actions comprising a routine are not dictated by the routine; they are simply the individual's chosen actions in particular situations (Agre, 1997a, p.109). Routines are not static structures, they evolve and change, and the main reason for change in a routine is a change in the relationship between the individual and the environment. A routine is a dynamic. The difference between the nouns "routine" and "dynamic" is that a routine involves a particular individual whereas a given dynamic might occur in the lives of many individuals (Agre, 1997a, p.108).

In the *plan-as-program* methodology, the planner passes a completed plan to the executor, which then gets executed step by step. Agre (1997a) proposes to abolish this distinction between planner and executor and conceive of the agent as thinking through the actions it undertakes at every moment. This does not correspond to *radical re-planning*, i.e. letting the planning module run at every step. Instead, once a train of reasoning is carried out it becomes part of a network based on a combinational logic circuit [8]. At every moment, inputs to the system are updated, and reasoning consists of the changed inputs propagating through the circuit. This way, the structure observed in the behavior of an agent is the result of similar decisions made in similar situations, instead of a fixed schema of action.

[8] Agre chooses digital circuitry as an alternative to computational metaphors, because "it is the fundamental stuff out of which digital computers's processors are made: fast, simple and continually sensitive to the states of both the agent and the environment" (Agre, 1997a, p.261).

2.1.2.1 Running Arguments

The Running Arguments (RA) system built by Agre is a demonstration of how routines can evolve and be the grounds for prevailing structure in activity. This system operates in a blocksworld simulation, where it takes commands from the user regarding how it should manipulate blocks with labels. The system is based on rules formulated in the Life language. Each rule consists of a list of conditions that can be **on** or **off** at any moment[9] and a proposition which is assigned **on** once the conditions are satisfied. A cartoon example is as follows:

```
R1: (if (sees the-shepherd the-wolf)
        (rings the-shepherd warning-bells))
```

This rule would read "As long as the shepherd sees the wolf, the shepherd rings warning bells" (Agre, 1997a, p. 127). As the "as long as" at the beginning of the phrase shows, this rule is **on** only when the conditions are satisfied and turns **off** when they are not satisfied any more.

Another important component of the system is a *dependency maintenance unit*. This unit is connected to a symbolic module called the *reasoner* which, through some mechanism, produces "thoughts" that the agent decides to believe or disbelieve. These thoughts consist of a conclusion and a set of reasons. When such a thought is formulated by the reasoner and handed over to the dependency system, it integrates this thought (from this point in the system on called a *justification*) into the dependency network[10]. When a situation is presented to the whole system in the form of propositions, the dependency system finds a configuration of **on** and **off** assignments to the conclusions that settles the network, if that deduction has already been made and the proposition has been integrated into the network. Therefore, a deduction that has been made once does not have to be repeated, but the system nevertheless thinks through each decision it makes.

In the case of RA, the central reasoning unit is a *running arguments* system. In a running arguments system, a decision arises through a discussion between different components of the machinery. At any moment, a proposal can be made about what has to be done and why. In case of a disagreement, the argumentation is carried on, each component bringing reasons why its proposal is better than the others. This corresponds to a style of rule-language programming, where the rules are formulated in the Life language.

In a number of experiments in the blocksworld environment, the RA system exhibited the emergence of routines, but also some obvious deficiencies (see Agre (1997a), Chapter 10 for a summary of the experiments). When the system first runs, the central reasoning unit does most of the work, and many rules have to be fired and arguments have to be evaluated in order for actions to be taken. Once a decision has been taken in a certain situation, in similar situations that follow the central reasoning system does not have to be run again,

[9] Agre (1997a) actually refers to positive and negative output values as **in** and **out**. For reasons of clarity they are referred to as **on** and **off** here.

[10] It is this network that is modelled on a combinational logic circuit by Agre

because previous argument structures have been integrated into a dependency network. The dependencies are helpful, because the simulated hand usually has to do the same things, and such dependencies, by accelerating operation, cause the emergence of routines. There is one condition for such transfer and consequent speeding up to occur: the objects in the environment which the simulated hand is manipulating should carry the same names: if in one case a block that has to be put on another is named **BLOCKA** and in the other **BLOCKB**, the reasoning system has to carry out the same argumentation once for each case, and the result is integrated into the dependency network separately in both cases. The reason for this apparently trivial deficiency is the inability of dependency networks to implement independent variables, which would allow the complete system to treat two differently named blocks the same way. One other important deficiency of the RA system is that it operates on an internal world model. The problems with world models have been explained in the first chapter; RA is not free of them.

2.1.2.2 Pengi

A game-playing program developed by Agre & Chapman offers substantive improvements on the RA system (Agre and Chapman (1990), Agre and Chapman (1987)). This program plays the game *Pengo*, where the aim of the user is to navigate a penguin in a dynamic environment populated with ice cubes and bees. If the penguin comes too close to a bee it dies and the game is over, but the bees can be killed by kicking ice cubes at them. Ice cubes slide vertically or horizontally when they are kicked if there aren't any obstacles on the way. The player wins the game when all the bees are dead. The Pengo environment provides a number of obvious improvements to that of blocksworld: "things move, the geometry is more complicated, the arrangement of objects in space is more meaningful, and the individual tasks relate in a clear way to an overall goal" (Agre, 1997a, p.264). Pengi, the program that plays Pengo, is based on the same running arguments system explained above. The whole system is coded into a combinational logic circuit, and at each moment a new decision on what has to be done next is made. Although Pengi does not employ complex plans, it exhibits goal-oriented behavior, and additionally it makes use of opportunities that arise and avoids gracefully any threats that show up.

Pengi consists of a central system, a visual system and a motor system. The visual system employs visual routines that operate on the data structures of the game[11]. The motor system is very simple in that only two actions are possible: moving up, down, left or right and kicking. In order to decide what to do, Pengi visualizes what has to be done, instead of using a world model. This is done using *visual markers.*

Visual markers are also an aspect of a new theory of representation proposed by Agre and Chapman (1990). In a conventional program, representations like

[11] See (Agre, 1997a, p.275) for a detailed description of the fairly complicated design of Pengi's visual system and the use of visual routines.

BEE35 or bee_a would refer to the same two bees irrespective of in which conditions the representations were individuated. These representations would maintain their references even if the bees were out of the screen or out of the sight of the agent. Such universal representations, however, are not common in real life, as Agre (1997a) points out. For example, each time we sit at a dinner table that's decorated with utensils, we do the same things with the utensils, i.e. eat our meal. We do not individuate each and every set of utensils separately each time we want to use them. Agre and Chapman (1990) call their alternative idea of representation *indexical-functional*, or *deictic* representation. Instead of a semantic relationship posited by traditional representations, the authors promote a causal relationship between the agent and the indexically and functionally individuated entities in the world. Being indexically individuated refers to being defined in terms of the relationship to the agent, whereas being functionally individuated refers to being defined in relation to the ongoing projects of the agent. Deictic representation that Pengi makes use of are of the form *thee-bee-I-am-chasing*. Such a variable refers to different bees at different times, but enables Pengi to make use of the contingencies currently available. Visual routines are used to register aspects of the entities in the environment, such as *the-bee-I-am-chasing-is-running-away*, by visually marking the entities in the environment with markers that correspond to such deictic representations.[12]

Pengi "plays a pretty decent game of Pengo . . . it wins from time to time and usually puts up a good fight" (Agre, 1997a, p.265). It engages in activity that shows variability: it will carry on different actions after each other to arrive at a goal, and this without planning. The reason for this variability is not structures that are kept inside the head and combined in a disembodied abstract space to produce more complex structures, but rather the dynamics of the interaction between Pengi and the game environment: "Pengi has its own kind of generativity, an infinity of dynamic possibilities rather than an infinity of structural combinations" (Agre, 1997a, p.23). Despite this variability, there is one important problem with Pengi: it does not learn. It learns neither the circuit with which it begins execution, nor does it improve itself relying on its own experience. Although e.g. Agre and Shrager (1990) offer a first view on the learning and evolution of routines, there aren't any adequate proposals on how such a network could be learned.

One important feature of Pengi is the reasons for which it would take an action. In a system that employs the plan-as-program view, an agent that follows a plan to kill a bee would take the action of hitting an ice cube because the program (or the plan) counter points to that action. Pengi, however, takes an action because it makes sense: it would kick an ice cube if it were possible to kill a bee with it. This way, Pengi makes use of contingencies, and does not have to re-plan whenever there is a change that makes it impossible to carry out the plan. What enables this is that each time Pengi has to make a decision, the

[12] For a recent work on the comparison of the performance of a system operating in the blocks world when it uses deictic representations and propositional representations, see Finney et al. (2002a) and Finney et al. (2002b). For an argument for the role of deictic representations and motor routines, seeBallard et al. (1997).

decision network runs again, producing a fresh evaluation of the environment. This is called *improvisation* by Agre (1997a). The use of deictic representations also enables Pengi to engage in the same kind of interaction with each cube and bee, without having to individuate and plan for each and every one of them.

That Pengi can get around in the Pengo environment is of course no proof that plans are unnecessary or are not used at all by human beings. The point is rather that their role is more complicated than plans-as-programs. Plans are just one kind of resource, just like any other resource human beings use, e.g. tools, external memory devices etc. The way plans are used by humans is similar to how instructions given in natural language are used, i.e. as linguistic entities that have to be interpreted in their own right before they can be used. Chapman (1991) is an interesting study into the use of instructions as additional resources in the context of computer games.

2.1.3 Steels' Language Games

One eminent scientist that does research on embodied agents is Luc Steels. Steels carried out extensive research on behavior-based AI. He focused at the beginning on low-level studies on intelligence (e.g. see Steels (1997b)). The robots developed by Steels operated in *robot ecosystems*, physical environments cluttered with multiple robots and objects. Behaviors are implemented as networks of processes, where a process can increase or decrease a quantity (that is a control variable, such as the speed of a motor) as a function of the evolution of other quantities (Steels, 1994). What is stressed by Steels in these early studies is the emergence of behaviors from the dynamics of interaction between the process networks and the environment (Steels, 1991).

Steels aims to move beyond the basic capacities such as obstacle avoidance and navigation he and many other researchers have studied in embodied AI. According to Steels and Vogt (1997), in order to move from agents that can solve such low level tasks towards agents that could be said to exhibit "cognition", robots have to be equipped with at least basic communication abilities. This communication must however again be autonomously developed by the agents themselves, in the spirit of the embodied AI bottom-up approach, and not designed or programmed in by a human engineer. The communicated concepts and the means of communication must be grounded in the sensory-motor experiences of the robot (Steels, 1997a). This way, robots could be used to study the origins of language and meaning in the self-organization and co-evolution of autonomous agents (Steels, 1996c). Steels and his collaborators carried out a number of experiments with robotic and software agents to study the emergence of reference and meaning, a lexicon, syntax and phonology. Here a brief overview of the first two aspects will be given.

In the studies of perceptually grounded meaning creation, meaning is defined as "a conceptualization or categorization of reality which is relevant from the viewpoint of the agent" (Steels, 1996b). The hypotheses tested is that the origins of meaning can be found in construction and selection processes embedded in discrimination tasks. The agent attempts to discriminate one object or situation from others using just low-level sensory processing. Each individual agent is able to construct its own visual features by segmenting the input space of its different sensory channels. The attempts to perform a discrimination based on the current feature repertoire and the adaptation of the repertoire is called a discrimination game. In one such game, if the discrimination based on one or more distinctive feature sets fails, the agent will construct new feature detectors. Feature detectors are refined in the process and form discrimination trees. As a result, "the system arrives quite rapidly at a set of possible features for discriminating objects. Most interestingly, the system remains adaptive when new objects are added or when new sensory channels become available" (Steels, 1996b, p.14).

Lexicon formation is based on language games, which involve a dialogue between two agents that interact in a common situation. A word is a sequence of letters drawn from a finite shared alphabet, and an utterance is a set of words. Agents have the capability of creating new words (as random combinations of letters from the alphabet) and associating these new words with sets of features they are meant to denote (Steels, 1996a). One agent communicates words to the other and the other agent tries to guess the set of features of the commonly perceived situation that the other agent might refer to with a word. It might find out that its initial guess was wrong when the same word is later used in another situation that does not contain the same features. Then another feature set would have to be assumed as the meaning of this word. On the other hand, a word can successfully be used by both agents if the assumed meaning (i.e. feature set) fits to all situations they both encounter. By using the common situation for feedback throughout a history of interactions, a set of common words and meanings emerge in both agents. As a result of the experiments, "it was shown that self-organization is an effective mechanism for achieving coherence and many properties of natural languages, in particular synonymy, ambiguity and multiple-word sentences, occur as a side effect of the proposed lexicon formation process“ Steels (1996a).

The two agents start out with no repertoire of perceptual distinctions and no lexicon. After a number of discrimination and language games, they have acquired

1. a perceptual system for categorizing sensory experiences and identifying distinctive feature sets and

2. a lexicon that associates features or feature sets with words and vice-versa (Steels and Vogt, 1997).

In a sense, the agents thus can be said to autonomously acquire grounded means of communication.

There is one important problem with Steels' approach to symbol grounding, however. Namely that the agents interact solely for the purpose of playing these pre-programmed games. Thus, the categories they acquire serve no other purpose for the agents than to utter and compare them with other such categories. In this sense, "[t]he system has no concept of . . . what to use the produced labels for, i.e. it is not embedded in any context that would allow/require it to make any meaningful use of these labels“ (Ziemke, 1999). The case is very different with categories and symbols used by natural agents like us: Our categories develop primarily due to and to serve our needs, they are related to our purposes in successfully and autonomously interacting with the world. A category is “ours“ when it has a function in our cognitive machinery that goes far beyond just using it in communication. The first step to symbol grounding should be learning categories that serve the agent in its interaction with the environment. And communication itself is used to successfully solve a task together with other agents, a task beyond just finding a common sets of words. The inventor of the idea of “language games” himself, Ludwig Wittgenstein, pointed out that we do not just learn words like “chair” and “table”, but rather get involved in behaviors like sitting on chairs, putting things on a table etc., and learn to use words in such contexts in which they play a role. Luc Steels' approach resembles rather the simple model of Augustinus explicitly criticized by Wittgenstein, where words are labels attached to features of a commonly perceived scene that one agent points to instruct the other (Wittgenstein, 1953, p.4).

Steels has also carried out experiments based on language games with robots. These experiments are reported in Section 3.3.4.2, due to their relevance in the context of the work explained in Chapter 3.

2.2 Biorobotics

Although they share fundamental principles and methods, the biorobotic methodology contrasts with behavior-based AI in its emphasis for comparison with natural agents and the level at which the models are formulated. In this section, some important examples of biorobotics will be presented and the main characteristics of the methodology will be discussed.

In one study, Prof. Dr. Holk Cruse and his team at the University of Bielefeld studied the coordination of the six legs of the locust Carausius morosus and the emergence of different types of gait patterns depending on the structure of the surface the insects walked on and their moving speed. This model was then implemented in the form of a six-legged mobile robot by Prof. F. Pfeiffer and his team at the Technical University Munich. The robot model showed similar gait patterns under similar conditions (surface structure walked on, walking speed). Moreover, the experiences with the robot model led to a significant improvement: It was realized that the interaction with the environment could be used to simplify the computations needed within the agent. This insight into an important aspect of the behavior of the locust was made possible by the availability of an embodied model (Dean et al., 1999; Schmitz et al., 2001).

Another prominent example of biorobotics is the robot model of the navigation behavior of the Saharan ant *Cataglyphis Fortis.* This ant is able to find its way back to its nest over a long distance even though pheromones cannot be used to mark the track because of the heat and the desert environment lacks cues which could be used as landmarks by an ant. The proposal of Prof. Dr. Rüdiger Wehner (University of Zürich, Switzerland) that the ant can make use of polarized light to find its way back was tested with a mobile robot model implemented by Dr. Dimitrios Lambrinos of the AI Lab at the same university. A robot model with a polarized light sensor was programmed with the proposed orienting mechanism and was tested in the original Saharan habitat of the ant. In the course of developing this real-world implementation, the researchers were able to understand the role of certain neuronal mechanisms possessed by the ant for the processing of the polarized light data, which seemed redundant before. As it turned out, the robot had to use similar mechanisms to deal with the noisy real-world data. This had not been understood until an embodied model tested in the real-world environment had been used (Lambrinos et al., 1999).

Webb (1994) presents a study of a robotic model of cricket phonotaxis. The phonotaxis behavior consists of the female crickets' finding a conspecific male by moving towards the sound the male produces. The model by Webb (1994) brings together the sensory and motoric aspects of the problem, and does this without recourse to the traditional notions of representation and construction of a world model. The embodied model makes use of simple filters and direct connections between the sensory channels and motors, and local variables whose role is comparable to "the function of the gears connecting the motors to the wheels" (Webb, 1994, p.53), rather than to those of representations in a symbolic system. The robot was tested in experimental conditions comparable with those used with crickets, and was able to demonstrate a number of key effects that occur also in experiments with crickets, such as phonotaxis, recognition of the ideal syllable rate, and choosing one sound source despite the existence of multiple sources. Webb (1994) concludes that "the cricket's response can plausibly be explained by a combination of slow auditory neural response (effective low pass filtering) and temporal summation in motor neuron response (effective high pass filtering)" (p. 51). See Figure 2.3 for a diagram of the cricket auditory system and the circuitry implemented in the robot control system.

Webb points to a number of facts about the model that are important. The first of these concerns the nature of robots as models of biological phenomena. She points out that in the cognitive scientific studies of robotics, the connection to biology consists of *adopted vocabulary*. However, there are also not many biological perceptual systems that are understood good enough to be implemented on robots. The methodology championed by Webb is based on the idea that "the *process* of attempting to implement physical models of biological systems can potentially contribute to our understanding of how perceptual systems work" (Webb, 1994, p.45)[13]. Webb also argues strongly for the value of robotic models in biology. First of all, the sensory and motor apparatus of the robot is as a matter of fact less precise than those of the insect, which means that the performance of the robot is not a result of superior machinery: the robot constitutes a *subset* of the capabilities of the cricket, rather than an *abstraction* of them.

[13] This is in contrast to Beer's approach, which will be explained below.

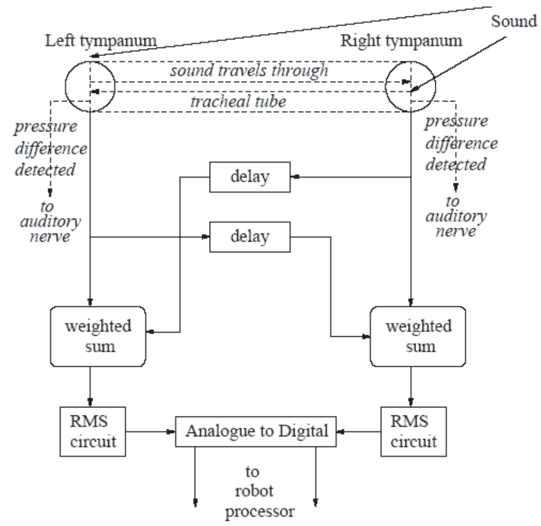

Figure 2.3: The cricket auditory system (dotted lines and italic text) and the circuitry that copies its function. RMS (Root Mean Square) circuits measure the amplitude of the signals. Adapted from Webb and Harrison (2002).

The second point concerns the value of robot models against simulation[14]. She makes three pertaining points:

1. A model will be trivial as far as it does not attempt to bring together sensory and perceptual aspects, because taxis is not such a complicated pattern of behavior.

2. "It would require a great deal of effort to build a computer model that reflected the real situation well enough to make strong claims that the mechanism actually works" (Webb, 1994, p.53).

3. In a simulation, the correct functioning of a model may be result of idealized conditions: the condition of the model object being a subset of the real object would not hold.

Webb articulates her ideas about robotic modelling in biology in a later paper. Webb (2001) addresses the nature of models in biology, and the dimensions which should be used for classifying and evaluating embodied models[15]. She argues that for the inclusion of a work under the rubric "biorobotic modelling", the following criteria should be satisfied:

[14] Webb actually attempted to build a computer model, and is speaking in the light of this experience.

[15] For a general review of biorobotics from the same author see Webb (2000).

- It must be robotic, i.e. "[t]he system should be physically instantiated and have unmediated contact with the external environment" (Webb, 2001, p.1037).
- It must be biological, i.e. "[o]ne aim in building the system should be to address a biological hypothesis or demonstrate understanding of a biological system" (Webb, 2001, p.1037).

This excludes disembodied work, where the environment and the animal are simulated on a computer, and behavior-based approaches in AI, where there is no attempt made to compare the performance of artificial agents to that of natural agents. Nevertheless, one important aspect is shared with the behavior-based approaches: "It is the engineering requirement of making something that actually works that creates much of the hypothesis testing power of robotic models of biological systems" (Webb, 2001, p.1046). The aim of biorobotics is to build systems that function, and this makes up the justifying power of biorobotics. Relevant here is the problem of accuracy and technical means: the current technology for sensors and actuators does not provide devices which allow the precise reproduction of natural behavior. Webb (2001) argues that instead of working on one precise model that can not be built due to technological constraints, more will be learned by building multiple relatively imprecise models that address different aspects of a phenomenon.

Another important parallel between biorobotics and behavior-based AI is the role of abstraction in the two methodologies. In behavior-based AI, simplification of a problem involves accounting for certain abilities that are exhibited by the natural agent through other means – such as using a speech processing system known to be unrealistic – instead of assuming ideal conditions, such as perfect environmental conditions and sensory devices. Certain kinds of idealization are impossible anyway if one follows an embodied approach: physical phenomena such as friction and momentum effects are outside any influence of the researcher. Webb (2001) points out that this is also the case in biorobotics: "What does distinguish abstraction in biorobotics from simulations is that it usually occurs by leaving out details, by substitution, or by simplifying the representation, rather than by *idealizing* the objects or functions to be performed" (p.1047).

2.2.1 Randal D. Beer

Randal D. Beer has also proposed an alternative view of AI, a view which has many parallels with the work of Brooks and Agre. What Beer's work shares with these efforts is the idea that "the appropriate patterns of behavior emerge from the dynamic interaction between an intelligent agent and its environment. The ability of its internal control mechanisms to somehow mirror the structure of its external environment is irrelevant" (Beer, 1990, p.14). According to Beer, intelligence is adaptive behavior, and "[a]ll that is required for adaptive behavior is a structural congruence between the dynamics of an intelligent agent's internal mechanisms and the dynamics of its external environment" (Beer, 1990, p.14).

Giving up the principal methodology of symbolic AI used to model intelligent agents, namely the symbolic modelling of expert performance in restricted domains, opens up two problems: what should one model if one is to abandon the abstracting method of symbolic AI, and with which tools and at which level, if not with explicit world models and representation?

Beer offers a new methodology he calls *computational neuroethology*, which is “the computer modelling of the neural control of behavior in simpler whole animals” (Beer, 1990, p.17), as a solution to these two problems. The working assumptions of computational neuroethology are as follows (Beer, 1990, p.17):

1. The ability to flexibly cope with the real world is a defining characteristic of intelligent behavior, and more fundamental than conscious deliberation.

2. Adaptive behavior derives from a structural congruency between the dynamics of an intelligent agent's internal mechanisms and the dynamics of its external environment.

3. Modeling the neural control of behavior in simpler whole animals will provide insights into the nature of the dynamics required for adaptive behavior.

Therefore, one should model complete agents instead of specific capacities, and the models should be at the neural level. Beer chooses as a subject for his model an insect that's similar to the American cockroach, *Periplaneta americana*, and names this simulated animal the *Periplaneta computatrix*, the computer cockroach.

The *Periplaneta computatrix* was first realized in a computer simulation, which was “only complex enough to support the behaviors of interest” (Beer, 1990, p.48). The control mechanism was based on a neural network implementation in the simulated model. Later on, a similar mechanism was implemented on physical machinery. A model of walking in insects for six-legged robots with 2 and 3 degrees of freedom was developed by Beer and his colleagues at the Case Western Reserve University. One of the aims of their research was to implement and test different hypotheses about the nature of walking in insects and the emergence of different gaits[16]. Although the mechanism in the simulated and the embodied models is very similar, the controller for the embodied model is not so much biologically inspired, not relying on biological mechanisms such as inhibition and stimulation. Here the mechanism for the embodied model will be explained: the essence of the two systems is the same.

Beer claims that using a centralized system to control the legs of a hexapod robot results in heavy computational load, which would be unrealistic for the simple nervous system of a cockroach. Therefore they have built a distributed system which relies on separate processors for each leg and simple communication channels between these processors. A leg can at any time be either in swing movement or positioned on the ground, which is called stance. The dynamics is centered around two kinematic parameters, the anterior extreme

[16] A gait refers to a pattern of leg movement.

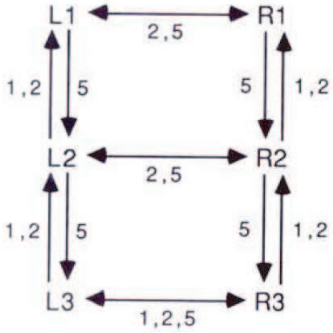

Figure 2.4: Influence of different mechanisms on the legs (adapted from Espenschied et al. (1993)).

position (AEP) and the posterior extreme position (PEP). These parameters control the switch from swing to stance and vice versa: when a leg reaches the PEP, it switches to swing, and when it reaches AEP, it switches to stance (see Espenschied et al. (1996) and Espenschied et al. (1993) for detailed explanations of the mechanisms). The leg processors influence each others' PEP and AEP values: Figure 2.4 shows the connections between the processors and the directions of the influences of each leg on the others. Numbers 1, 2 and 5 refer to different mechanisms with which leg processors influence each other. The arrows show the directions of these influences, where the arrow originates from the sender of an influence and ends at the receiver. The mechanisms functions as follows:

- Mechanism 1 shifts the PEP of the receiving leg backward.
- Mechanism 2 shifts the PEP of the receiving leg forward.
- Mechanism 5 shifts the PEP of the receiving leg forward. While mechanisms 1 and 2 are step functions, mechanism 5 is a ramp function.

These simple mechanisms and connections were enough for the robot to exhibit three different insect-like gait patterns: "As the speed is varied, a continuous range of statically stable insect like gaits are produced . . . The gaits range from the wave gait to the tripod gate." (Espenschied et al., 1993, p.460).

Beer and his colleagues have also carried out lesioning studies of the same model. When a cockroach loses a leg or a neural connection is severed, it can still exhibit coherent gait patterns: this is not the case in centrally controlled systems, where the loss of one unit or connection can effect the functioning of the whole system. The hexapod robot exhibits *graceful degradation*, in that the severing of a connection does not cause catastrophic failure: "The robot walks effectively throughout the range of speeds despite the removal of any single inter-leg influence. By *effective* locomotion, we mean that the robot walks in a gait that is primarily statically stable but may experience occasional brief periods of static instability" (Espenschied et al., 1993, p.461).

The designers of this hexapod robot state that they "tend to err by including more biology than may at first appear to be strictly necessary. The reason for this is straightforward: it almost always pays off" (Beer et al., 1997, p.33). The first source of inspiration for their work comes from biological studies, and the results of experimental studies on animal behavior and biology are integrated into the design and testing of a robot. This is one of the reasons why they choose insects as the object of their model: invertebrates display complex goal-oriented behavior, and the study of their nervous system is for most cases more feasible than it is for vertebrates. Nevertheless, there are many cases where the biological knowledge is not enough and the designer has to fill in the gaps: "Unfortunately, much of the artificial insect's nervous system is rather *ad hoc.* While some portions of it are based directly upon neurobiological data, many other parts were hand-designed" (Beer, 1990, p.167).

2.2.2 Some remarks on Connectionism

A question frequently stated asks for the difference between what is called connectionist AI, which appears also under the names of neural networks and parallel distributed processing (Smolensky, 1988; Rumelhart and McClelland, 1986) and New AI. Although connectionist modelling is claimed to be the alternative in cognitive modelling to classical symbolic models, the two methodologies have in essence a lot in common. Among these commonalities, the most important are the use of micro-worlds, and the existence of these neural nets in a biological vacuum (Cliff, 1991). What is meant with a biological vacuum is a disconnectedness from the biological environment which defines a living being, the independence from the physical bases a biological organism would be situated in. This vacuum leads to the neural nets facing the same problem as the symbolic systems, namely the symbol grounding problem. Computational neuroethology promises to study intelligent beings as complete organisms and coupled to their environment. Therefore, it appears to be the more promising approach at the biological level, when compared to connectionism.

Chapter 3

TINAH Experiments

In the second chapter, outlines of biorobotics and behavior-based AI have been given. These research programs promote using concepts from biology and ethology in the design of robotic systems. The major difference between them is in the emphasis they put on the accordance of the borrowed concepts with experimental methods and evaluation. In this chapter, a research program that has been proposed as a combination and possible cognitive scientific mixture of these two approaches, comparative cognitive robotics (from here on CCR), will be explained. The author has also taken part in a a project that applied the framework of CCR to a concrete study of animal learning. The main outlines and results of this project will also be presented.

3.1 Comparative Cognitive Robotics

Though very successful, the approaches of biorobotics and computational neuroethology also have some important limitations. One problem is the level at which they are operating. These two methodologies both model behavior at the neurobiological level. However, as we saw in the section on computational neuroethology, our knowledge of the biology of animals is very limited, even for very simple animals such as cockroaches: Beer had to fill in gaps in the neurobiology, albeit with neurologically plausible assumptions, although he chose cockroaches for their simplicity in the first place.

Due to the nature of biological data, the models are only applicable to one species or a small group of species. In the examples given in the second chapter, these species and behaviors are navigation in the desert ant and gaiting behavior in a certain species of insect. Moreover, only aspects of sensory-motor coordination are studied, which imposes severe restrictions on the methodology from a cognitive scientific point of view. This kind of research reveals little about human cognition, even if we were to gather a huge amount of such data.

The main focus of research in biorobotics is on fixed behavioral patterns which show little or no adaptation and learning (e.g. the mechanism to coordinate the six legs in a stick insect is rather fixed, and only certain parameters have to be updated in the orienting behavior of *Cataglyphis*). The explanatory level is also a neural one. Because explanations on this level are not available for more complex phenomena, especially phenomena that involves autonomous learning and adaptation, one has to stick to a narrow range of rigid phenomena in a limited number of species. Hence, what is needed is an approach which allows studying behavioral mechanisms that can be found in many different species. These mechanisms should go beyond sensory-motor coordination, be described on a level of explanation other than the neuronal level and relate to the central phenomena of adaptation and learning.

Comparative Cognitive Robotics (John (1998); John and Werner (2004a); John and Werner (2004b)) is a research framework that "aims to understand better how living beings learn by imitating these with animats and robots, in this process orients itself to the empirical findings of the psychology of learning and memory, and concentrates on such phenomena whose universality has been proven by comparative psychology"(John, 1998, p.88). CCR aims to retain the strengths of biorobotics stemming from its empirical methodology, while getting beyond its weaknesses by basing the research on a more cognitively-oriented foundation. As collaboration partner in the construction of autonomous agents, CCR takes comparative psychology instead of biology. Instead of mere sensory-motor coordination, theories of learning and adaptation are the target of modelling. Comparative psychology aims at finding common mechanisms of learning and adaptation in a wide variety of species which can be as different as monkeys, pigeons, dolphins, rats, and even bees and humans. This research gives insight into possible evolutionary steps in the development of complex cognitive abilities as well as into mechanisms which seem to be common to many – if not all – species (cf. Domjan (1998); McLaren et al. (1994); McPhail (1987); Roitblat and von Fersen (1992); Shanks (1995)). Focusing on comparative psychology rather than biology provides a horizon beyond behaviors that are limited to a specific species and have little to offer on learning and adaptivity. This also permits independence from the neurological level of explanation. The focus is rather on psychological, yet "subsymbolic" models of behavioral control.

In implementing empirical models in robotic agents and testing them in the same environments with the natural agents, a most parsimonious approach is taken: the model at the start should build in as little designer knowledge as possible. If the robot model and the animal show similar adaptive behavior under similar conditions, the model is successful and can be extended to cover more phenomena. Otherwise, the model is modified in again a parsimonious way, and is subjected to the same testing conditions. Another important point is that open questions in the design process should be turned into empirical questions and answers for these questions should be sought in the same experimental framework with the natural agents. The environments for the robot and animal experiments are ideally standard learning environments (e.g. experiments done in a Skinner box). Such environments allow the examination of the details of adaptive behavior in animals under experimentally controlled conditions, and at the same time verification of the robot's behavior under similar conditions.

To summarize, the main features of comparative cognitive robotics are as follows (for a detailed account see John (1998)):

1. As a collaboration partner in constructing autonomous agents, comparative psychology of learning and adaptation is chosen.

2. Focus is on those phenomena of learning and adaptation that can be found in a wide variety of species.

3. Empirical research with the model animal and the construction of the robot model work hand in hand. Work on the robot model inspires new experiments and new theories, and the empirical findings are used to update the robot model.

4. The model animal and the robot model are tested in the same or similar environments commonly used in comparative psychology, with the same or similar means of analysis. The match between those measurements is taken as an indication of the success of the model.

The rest of this chapter will report on a project in the framework of CCR that was carried out at the University of Osnabrück. The project, named EROSAL (Empirical RObot Study of Animal Learning) had the aim of studying theories on categorization, using as the resource for empirical data two groups of experiments carried out by Christian Werner of the C. and O. Vogt Brain Research Institute at the Heinrich-Heine-University of Düsseldorf. The methods and results of these experiments (referred to as the Düsseldorf experiments in the rest of the text) will be explained in the following section. Afterwards, the EROSAL project will be expounded.

3.2 The Düsseldorf Experiments

The experiments that we aimed to replicate in our project with a robot model studied the discrimination capability of hens (genus *Gallus gallus fd*) in two different tasks (Werner and Rehkämper, 2001, 1999; Werner et al., 2004, 2003; Werner, 1999)[1]. These simultaneous discrimination tasks involved two different sets of stimuli. The experimental environment was a quadratic Skinner box with three Plexiglas pecking discs. The visual stimuli were projected onto the pecking discs using a slide projector. The tasks involved two stimuli, so one of the pecking discs (the middle one) was obstructed with black cardboard. Positive reinforcement in the form of food was given to the hens using a food hopper. There were two light sources inside the conditioning chamber: a white bulb (the house light) and a red one (the feedback light). The house light was turned off to signal negative reinforcement, whereas the feedback light was turned on to signal the presence of food after a correct discrimination by the

[1] Birds are traditionally used in experiments involving visual stimuli, due to their superior visual capabilities. In experiments involving olfactory and gustative stimuli, in contrast, rats and mice are traditionally used as experimental animals.

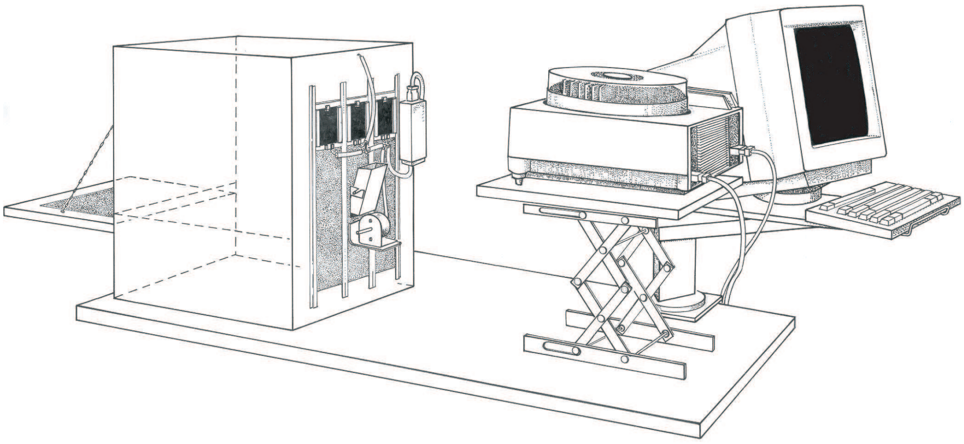

Figure 3.1: The experimental environment in which the chickens and the robot were tested.

hen. During the experiments, the hens were neither food nor water deprived. Figure 3.1 shows the experimental setup from outside, and Figure 3.2 shows the hen and the apparatus inside the box.

3.2.1 The first group of experiments

The set of stimuli used in the first group of experiments consisted of opaque, black geometrical figures of equal area. These stimuli varied along the dimensions given in Table 3.1. All possible combinations of these variations were used, which led to 54 different stimuli. Figure 3.3 shows some sample stimuli from these experiments.

The figures are integral compounds, because each of the dimensions used contributed to the form of the figures. Integral compounds are stimuli "in which the perception of each element is influenced by each other element and in which the elements resist individual analysis by the subject" (Riley, 1984, p.335). It was known that hens preferred rounded objects; therefore, in this first set, the chickens learned to discriminate between stimuli with sharp corners and those with round corners. The aim of the first group of experiments was to "investigate whether chickens exposed to a discrimination and categorization problem were able to adapt their pecking behavior to a single dimension only, or whether they used additional cues which were uncorrelated with reinforcement" (Werner and Rehkämper, 1999, p.30).

The stimuli were distributed elaborately to avoid any unwanted effects. The distinction the hens had to learn was between sharp and round corners. Pecks on the sharp cornered shapes were rewarded with food, whereas pecks on the

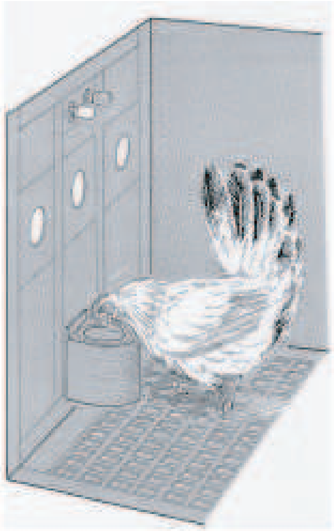

Figure 3.2: The hen and the apparatus inside the box.

Name of dimension	*Possible feature values in this dimension*		
Corners	Rounded	Sharp	
Basic figure	square (1:1)	rectangle (1:1.6)	rectangle (1:2.8)
Sloped side	horizontal	vertical	all
Slope	10^o	25^o	45^o

Table 3.1: The variation of the stimuli used in the first set of experiments with hens.

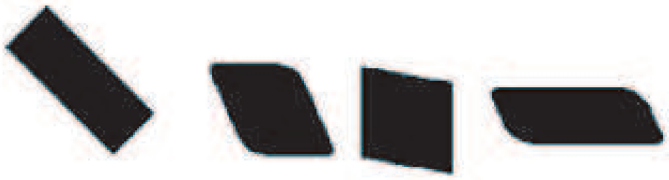

Figure 3.3: Samples from the first set of visual stimuli used in the chicken experiments.

round-cornered shapes were punished by turning the lights out, which is aversive for hens. The stimuli were divided into 27 pairs, with each pair consisting of a figure with rounded corners and another with sharp corners. These 27 pairs differed in all four dimensions. The stimuli were further divided into three sets consisting of nine of these pairs. This division was done such that in each group, the values of the additional features of the stimuli with sharp corners were presented an equal number of times.

Eighteen hens were used to carry out the experiments, of which one died during the study. Most of these hens had already taken part in an earlier experiments, albeit with different kinds of stimuli. The ones that did not have experimental experience went through an auto-shaping procedure that ran as follows:

Phase 1: Getting acquainted with the Skinner box

- Food was put around the food hopper to attract the hen's pecking.
- Light on the pecking discs (without patterns) was switched on permanently.
- The food hopper was opened for 10 seconds, followed by 20 seconds of pause with a closed food hopper, then it was opened again, etc.

Phase 2: Autoshaping

- 15 seconds of pause was followed by the food light switched on for 10 seconds.
- If the hens pecked during this interval, the food hopper was opened immediately. Otherwise, it was opened after the interval each time for 5 seconds.
- Afterwards the pause started again.

Phase 3: The time parameters were adapted step by step to the ones used in the experiment.

3.2.1.1 Experiment 1: Successive training

In the first experiment, the first group of nine pairs from the three groups of stimulus pairs was used. Each stimulus pair in this set was presented until the subject reached a certain criterion of performance. The learning criterion for a hen's successfully finishing the experiment was either making at least 80% correct choices in each of five consecutive sessions or making at least 90% correct choices in three out of five consecutive sessions.[2] The order of the presented pairs was different for each hen (pseudo-randomized and balanced), and the presentation of the reinforced stimuli (i.e. stimuli with sharp corners) on the left or the right pecking disc was balanced.

[2] A *session* refers to the presentation of all nine stimuli, and the presentation of one stimulus pair is referred to as a *trial*. See Werner and Rehkämper (1999) for a more detailed explanation of the experimental procedure.

The experiment was stopped when the last of the seven fastest learners among the hens reached the learning criterion, which took four months. There was a large variability in the learning ability and performance of the hens. Among the eighteen hens, seven learned to discriminate all nine pairs, one hen eight pairs, one hen seven, one hen four, one hen two, and two hens were unable to exhibit any discrimination behavior. The quickest learner needed six sessions in order to learn to discriminate all nine pairs, whereas the slowest among the seven fastest learners needed 32 sessions. There were two interesting effects that were observed. One was the speeding up of learning through consecutive trials. Learning became faster during the sessions, from around eleven sessions which were needed to learn the first stimulus pairs to about seven for the last: this can be taken as an indication of generalization. Another effect was drops below chance performance in the first session following pair exchanges. If the hens were not paying attention to the integral dimensions, it would be expected that they start the discrimination task at chance level. That they perform worse could be due to their picking an irrelevant cue. The aim of the second experiment was to test whether the hens could attend selectively to a single dimension.

3.2.1.2 Experiment 2: Simultaneous training

In this experiment, the fastest learning seven hens from the first experiment were used, in order to speed up the experimental procedure. In order to avoid the hens switching from one disc to the other during the presentation of a pair, they were given positive reinforcement if they pecked three times on the correct slide. In the training phase, each pair from the first stimulus set was presented three times, and in case of an incorrect choice correction trials were inserted, which presented the same stimulus pair once more. In contrast to the first experiment, the pairs were presented in randomized order. At the end of the training phase, the hens had learned to peck on the stimuli with sharp corners (see Figure 3.4). The training phase was followed by a testing phase, in which it was tested whether the hens could generalize their pecking behavior to new stimuli. For each session, the nine stimuli of the first group of nine were presented three times, plus one pair from the second group of stimuli, again three times, mixed (pseudo-randomized, balanced) among the familiar stimuli. Nine sessions per test were carried out. In each session, for the new stimuli, pecking had no effect for the first presentation, but only for the second and third presentations. The inclusion of new stimuli was limited to one new pair per session. There were two reasons for this. One was avoiding the confrontation with a lot of new stimulus material and a possible drastic decrease of performance. The other was to assure that natural variations of performance between days do not have drastic effects on the measurements. The hens discriminated the new stimulus pairs as well as the ones they had already been trained with. Around six pecks were made on sharp-cornered figures, and two on rounded ones. The number of pecks on different kinds of stimuli did not differ for familiar and unfamiliar pairs. One important result was that there was no indication of an effect of set size.

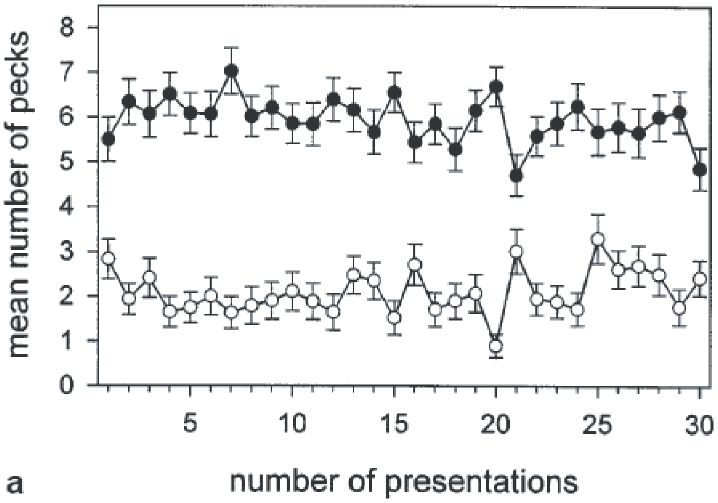

Figure 3.4: Mean number of pecks and SE for sharp-cornered (filled) and rounded (unfilled) figures in the second half of the training period (adapted from Werner and Rehkämper (1999)).

3.2.1.3 Experiment 3

The aim of the third experiment was to analyze the performance of the hens in relation to the other dimensions. In this experiment, all the stimulus pairs from the first and the second sets of stimuli were presented, and in the test phase the hens were tested for their performance with the stimuli from the third set. Similar to the second experiment, one pair from the third set was presented randomly placed between nine pairs from set one and nine from set two. The discrimination performance of the hens was very similar to that in the second experiment: in the training phase, the number of pecks on the cornered stimuli was around six, whereas the number of pecks on the rounded ones was around two. In the testing phase, the hens demonstrated the same discrimination behavior with the unfamiliar stimuli from the third set.

Although these results show that the hens have learned to discriminate according to the significant dimension, in this case whether the corners were sharp or rounded, a number of findings cast doubt on the classical view on classification. Statistical analysis revealed that for each individual hen, at least one other, "irrelevant" dimension significantly influenced the pecking behavior, although the stimulus material and the experiment was constructed such that only the distinction of round-cornered vs. sharp-cornered stimuli could serve as a predictor for reward. This was true also on the level of stimulus pairs. Not all stimulus pairs were discriminated equally well, although each time, one sharp-cornered and one round-cornered stimulus was paired. In addition, there was an overall decrease of discrimination performance over the sessions, depending

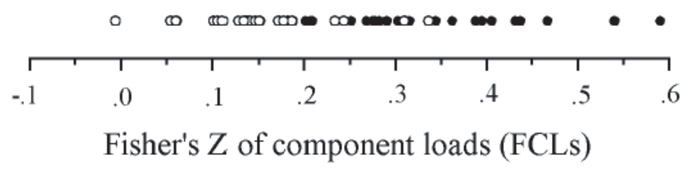

Figure 3.5: The results of the PCA. Filled circles are exemplars of the reinforced category, whereas empty circles are members of the non-reinforced category.

on how many stimulus pairs had been trained.

In order to test whether the configural effect of the stimuli was a result of learning the configural cues as category-relevant information or a result of the memory retrieval processes, a new experiment was carried out, in which the same stimuli were formed into new pairs (Werner and Rehkämper, 2001). Already on the first presentation of the new pairings, the hens were able to discriminate the two types of stimuli equally well, which is a strong indication that not the pairs but the individual exemplars were learned beforehand. In order to test the authors' ideas about the representation of category information by the hens, the number of pecks by the hens on the different pecking discs was taken as data and statistically evaluated. A principal component analysis (PCA) was done to find out which components influenced the pecking behavior, and to which extent. This analysis revealed that the corner dimension (i.e. whether the corner was sharp or rounded) was responsible for 54% of the variance, which made it the most important component. The other components could be used to order all stimuli according to the degree to which they lead to pecking or non-pecking (component loading, transformed by Fisher's Z transformation). On a one-dimensional axis, stimuli which would elicit pecking behavior were grouped more to the right, stimuli which would elicit non-pecking were grouped more to the left. The results are shown in Figure 3.5.

These theories would in this case predict different distributions of the components. Next, the predictions made by different theories will be explained, and compared to the actual results.

3.2.1.4 Feature-based theories

Theories that take a feature-based perspective partition stimuli into features that correspond to natural language terms. Category membership is an all-or-none quality, and is defined by the possession by a stimulus of a set of features or combinations of them (Pearce, 1994). Each exemplar of a given category is equally representative of the category. The expected distribution of the exemplars would then be one of random variation, and the mean of the exemplars in the relevant dimension would correspond to the membership criterion. The predictions of the feature theory are presented in Figure 3.6.

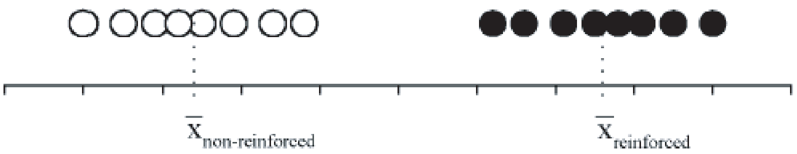
$\overline{X}_{non\text{-}reinforced}$
$\overline{X}_{reinforced}$

Figure 3.6: The prediction of a feature-based approach how instances would be separated.

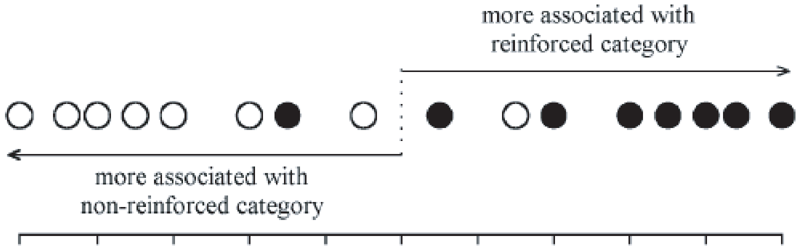

Figure 3.7: The prediction of a prototype-based approach how instances would be separated.

3.2.1.5 Prototype-based theories

These theories assume that a prototype is formed for each category, which is the most representative exemplar of that category.What determines category membership is similarity to this prototype (Posner and Keele, 1968; Rosch, 1975). A prototype-based account in this experiment would produce prototypes that are symmetrical along the significant dimension, because the dimensional structure of the reinforced and non-reinforced stimuli are the same except for the significant dimension. The center would be the point of lowest typicality for both components. See Figure 3.7 for the predictions of a prototype-based theory.

3.2.1.6 Exemplar-based theories

[3]n exemplar-based learning mechanisms, stimuli are considered to be complete configurations Medin and Schaffer (1978); Nosofsky (1984). In an exemplar-based account, no assumptions are made about the importance of any dimensions for category membership. The distribution of the exemplars depends only on their association with the selective responses that correspond to the categories (in this case, approach vs. avoid). It is possible that the exemplars are mixed with each other. See Figure 3.8 for a depiction of the predictions of an exemplar-based account. A comparison of Figure 3.5 and Figure 3.8 shows that

[3] I

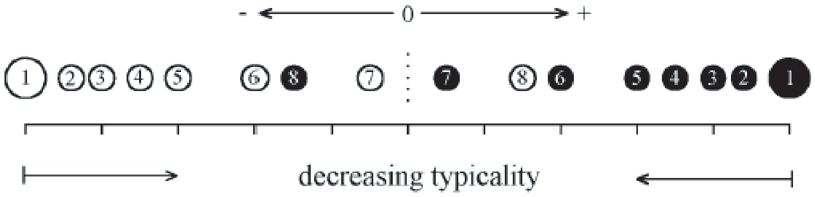

Figure 3.8: The prediction of a feature-based approach how instances would be separated.

among the predictions of the categorization theories, the exemplar-based account gets the closest to the experimental findings. The one-component model of categorization according to exemplar theory matches the results of discrimination of categories on the basis of the raw data on pecking rates....One can conclude that exemplar theory is more suitable for explaining chickens' discrimination of well-defined categories of multi-dimensional geometrical figures than either feature or prototype theory.

The authors conclude, regarding the first group of experiments, as follows: "we would interpret our results as indicating that chickens represent stimulus arrays as a whole... and that they might adapt their behavior to elements or dimensions of these representations secondarily" (Werner and Rehkämper, 2001, p.37).

3.2.2 The second group of Düsseldorf experiments

The first group of experiments explained above used only integral compound stimuli. In contrast to integral stimuli, one can speak of separable compound stimuli, which are defined as stimuli that can be decomposed. Decomposing means "that the subject performs operations on some property of the stimulus without reference to other properties" (Riley, 1984, p.335). In the literature of category acquisition, one can read of two different kinds of processing that correspond to these two kinds of stimuli: analytic vs. holistic processing. Analytic processing is assumed to take place with separable compound stimuli, wheres holistic processing is assumed to be the primary method of interacting with compound stimuli. Although an experiment that uses integral stimuli allows one to evaluate different theories of categorization, it is impossible to examine whether the claim for two kinds of processing is true. The aim of the second group of experiments was to test whether the chickens processed stimuli differently, depending on whether they were separable or integral (Werner et al., 2003).

Analytic processing Theories that are based on the analytical processing of stimulus claim that when a compound stimulus is presented to the animal, the stimulus is decomposed into its components (in the case of the stimuli used in

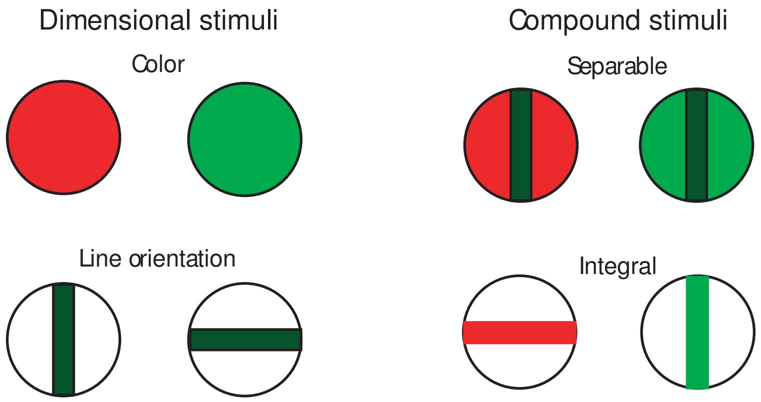

Figure 3.9: The stimuli used in the second group of experiments. Left: Dimensional stimuli consisting of one dimension, color (red or green) or line orientation (horizontal or vertical). Right: Compound stimuli. Top right: Separable compound stimuli. Bottom right: Integral compound stimuli.

this experiment, dimensions such as sloped side, slope, basic figure and corners) and then the stimulus as a set of such components and the reinforcement are the sources of learning (Cook et al., 1992; Leith and Maki, 1977).

Holistic processing Theories that assume holistic processing claim that a compound stimulus is accepted to the learning system as a complete configuration, without being analyzed into smaller parts. The stimulus as a complete pattern enters the learning process. In this kind of processing, one does not have to assume features that are extracted (Grant and MacDonald, 1986; Cox and D'Amato, 1982).[4]

The second group of Düsseldorf experiments used both integral and compound stimuli, in order to be able to find out whether hens process them differently. The stimuli dimensions were line orientation (horizontal or vertical) and color (red or green). The separable stimuli consisted of black lines on a colored background, whereas the integral stimuli consisted of colored lines (red or green and horizontal or vertical). The stimuli can be seen in Figure 3.9.

3.2.2.1 Experiment 1

If it is the case that chickens process integral and compound stimuli differently, it is expected that animals have more difficulty attending to the dimensions of an integral stimulus than attending the dimensions of a compound stimulus. In this experiment, it was tested whether chickens that learned to discriminate between

[4] See Sloman (1996) for a review of the literature of the same distinction in humans and an argument for the existence of two systems of reasoning in humans, one analytic and the other holistic.

color and line orientation transferred differently to integral and compound stimuli made up from the same dimensions. The experiment was carried out with nine hens which were experimentally naive except for one hen that had already taken part in the first group of experiments. The stimulus material was accepted to be different enough not to effect performance in this experiment. The hens were, as in the first group of experiments, neither food nor water deprived. The experimental setup was exactly the same as in Werner and Rehkämper (1999), explained above. For a more detailed explanation of the stimulus material and the experimental procedure see Werner et al. (2003)[5]. The separable compound stimuli were a black line on a colored background (designated as SCLO, which stands for Separable Compound Line Orientation; top right in Figure 3.9). The integral compound stimuli consisted of a colored line, either red and horizontal or green and vertical, on a transparent background (designated as ICLO, which stands for Integral Compound Line Orientation; bottom right in Figure 3.9). In order to acclimate the hens to the chamber, they went through an autoshaping procedure as in the first group of experiments.

Hens were first trained to discriminate between the dimensional color and line orientation stimuli. Four of the hens were reinforced for choosing red and horizontal, and five were reinforced for green and vertical. Each hen was trained for twenty trials daily, where ten trials were allocated for color stimuli and another ten for the line orientation stimuli. The criterion for ending the training phase was reaching 90% correct discrimination in one session for both color and line orientation stimuli. Some hens had difficulty learning to discriminate the line orientation stimuli – these hens were trained only with line orientation stimuli until reaching the 90% criterion before carrying on with the other phases of the experiment.

The transfer phase followed the training phase. In the transfer phase, integral and separable compounds were exchanged for dimensional stimuli. In the training phase, hens that had been reinforced for two particular elements were reinforced for choosing the compounds, either separable or compound, of the same elements. For example, a hen that was reinforced for choosing red and horizontal dimensional stimuli was reinforced in the transfer phase for choosing horizontal black line on a red background (separable compound) or horizontal red line (integral compound).

The hens learned to discriminate color faster than line orientation in the training phase. In the transfer phase, performance was better for separable stimuli than integral stimuli. These results are in accord with the hypothesis that dimensions making up a separable compound are easier to attend to than those of an integral compound. However, the authors argue that this could be due to a difference in the processing of similarity, e.g. more efficient processing of color compared to line orientation. If this is the case, color stimuli would be more similar to SCLO stimuli, because the overlapping color patch is much bigger.

[5] Werner et al. (2003) presents, in addition to the experiments, a mathematical model. This model will not be explained here.

3.2.2.2 Experiment 2

In this experiment, the hens were tested for the reversal of one dimension, while the other was reinforced as before. In the test phase, transfer to non-reversed compound stimuli was tested. The aim here is to test for a separation of dimensions associated with processing of SCLO compared to ICLO. This experiment included a new set of stimuli where the lines were two times thicker (designated SCTLO and ICTLO), in order to increase the area covered by color in ICLO (colored lines on transparent background) and decrease it in SCLO (black line on colored background).

In the training phase, three hens from the first experiment were first trained to criterion with the color reversed compared to the first experiment, while the reinforced line orientation remained the same. Afterwards, the transfer to compounds was tested, with the reinforcement contingency as in Experiment 1, i.e. not reversed. Each type of compound stimulus (two combination rules and two types of line thickness) was presented randomly for ten trials, until the hens reached criterion. This was followed by another dimensional training phase, this time with the line orientation reversed. A second transfer training followed this, whose reinforcement contingencies again remained unchanged. The reversal of line orientation and color dimensions, and the subsequent testing for transfer, was done three times in total.

The results revealed that for the stimuli with thicker lines, there were significant fewer errors for ICTLO compared to ICLO. This is in accord with the assumption that the size of the color patch has a significant role in a similarity decision. The reversal of color in the dimensional training phase led to significantly more errors for SCLO than for ICLO. This result is important, because it contradicts the predictions of an analytic theory of processing. If the hens processed the separable stimuli analytically, they should have made fewer errors with separable stimuli, because the learning process leading to an adequate response would simply have to consider the feature that is still reinforced after the reversal, and dismiss the other features (see Leith and Maki (1977)). A holistic processing account, in contrast, would not have any difficulty accounting for this result. After the reversal of line orientation, more errors were found for ICLO than for SCLO. Additionally, the number of errors after each reversal showed a continuous decrease. The results can be seen in Figure 3.10.

3.2.3 Evaluation of the Düsseldorf Experiments

The first group of experiments, which used only integral stimuli, provided proof for doubting the feature view of categorization. Statistical analysis revealed that, although the reinforcement was controlled strictly by just one dimension, hens were attending to irrelevant dimensions too. Furthermore, the results of the principal component analysis showed a distribution that could be best explained by an exemplar view of categorization. The second group of experiments provided proof for the claim that one does not need to assume different kinds of processing for separable and compound stimuli. Traditionally, separable com-

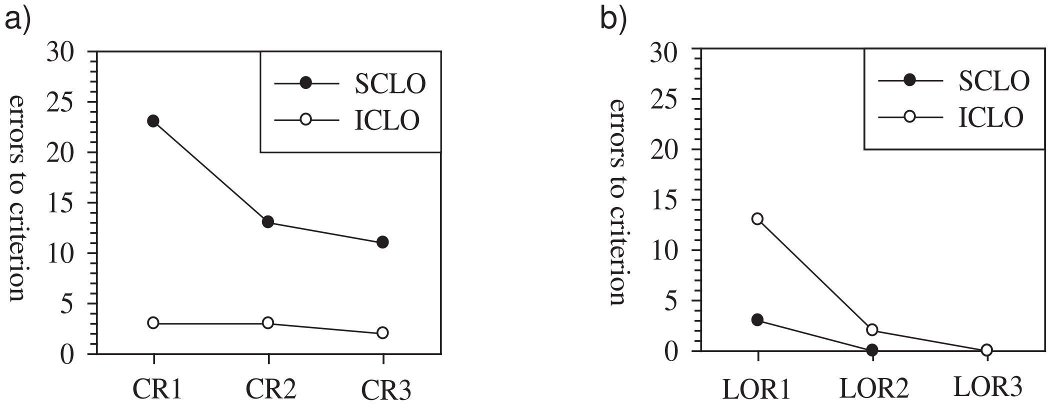

Figure 3.10: The results of the reversal experiments. LEFT Errors to criterion for successive color reversal trials. RIGHT Errors to criterion for successive line orientation reversal trials.

pounds are accepted to be processed analytically, and integral compounds are accepted to be processed holistically. In the second experiment, the combination rule of the compound stimuli (integral vs. compound) was not sufficient to predict transfer after reversal. The dimension that was reversed interacted with the combination rule to determine the salient dimension in the compound. These results point to a *holistic* processing of both kind of stimulus, integral and compound, with an *exemplar-based* categorization system.

3.3 The EROSAL Project

The CCR framework was pursued in a one year project at the University of Osnabrück in the Winter and Summer semesters 2002/2003 (Deiwiks et al., 2003). The project was called EROSAL, which stands for Empirical Robot Study of Animal Learning, and was for the partial fulfillment of the Cognitive Science Master's Degree at the same university. The students involved in the project were, in addition to the author, Christa Deiwiks, Katerina Gergou, Leonhard Läer, Rüdiger Land, Sascha Lange and Jan Plate. As the name suggests, our aim was to do an empirical study of animal learning using a robotic model. The study was to be in the framework of CCR, which proposes that the data produced by comparative psychological studies be taken as resources for the construction of models. We therefore had to pick a study of animal learning. We decided to work with Christian Werner and replicate the above explained experiments.

There were a number of reasons for our choosing to work with Christian

Werner. The most important among these was that the studies were on discrimination learning and categorization, which are central subjects in the field of cognitive science. The criteria for a good robot model put forward by the framework of CCR, listed on p. 45, were met; primarily, the experiment did not depend on an expensive and complex set-up, and we were able to meet the demands of building the robot using relatively cheap resources. Last but not least, we could rely on a collaboration between Christian W. Werner and our project group. This gave us the possibility to get plenty of information on the informal background of the experiment and about chicken behavior in general. It also helped a lot in constructing the actual model and setting up the experiments.

3.3.1 The Main Idea

As we have seen in the Düsseldorf experiments, it is not necessary to assume two different kinds of processing in order to account for the effect of the composition rule of a stimulus. An exemplar-based account is sufficient to explain the experimental findings of these experiments. The aim of our project was to build a robot model of the hens that would be able to carry out the same experiments. The learning mechanism we wanted to implement assumed that there exist similarity relations between dimensional and compound stimuli, which led to transfer of learning from dimensional training to compound training. We also believed that one of the effects that accounted for the results of the experiments with the hens was the existence of differences in efficiency of processing between dimensional stimuli, meaning that color is processed more efficiently than line orientation.

3.3.2 The Robot and How It Learned

In order to construct an embodied model of the discrimination behavior of the hens, we have built a robot that would operate in the same environment in which the hen experiments were carried out. The robot was named TINAH, an acronym that stands for *This Is Not A Hen.* The robot was built using Lego Mindstorms^{TM}, which provides a versatile set of tools for building custom robots, although limited in complexity and rigidity. In addition to the usual Lego building blocks, the Mindstorms^{TM}kit provides various sensors and electric motors, and an RCX (*Robot Control X*), a programmable brick with a microprocessor. The RCX can connect to the sensors, such as touch sensors and rotation counters, and drive the motors. The size of any program that can run on a RCX is limited by the size of the on-board memory of 512 Kilobytes. This is enough for simple programs, but the learning algorithm that we have implemented, which will be explained below, requires a much bigger memory. Therefore, we have opted for controlling the robot through the tower, an infra-red transceiver that connects to a computer via the serial port and communicates with the RCX. This way, we could control the robot by issuing commands from the computer. We have built the robot using two RCX bricks in order to provide for future contingencies, but only one of these bricks was

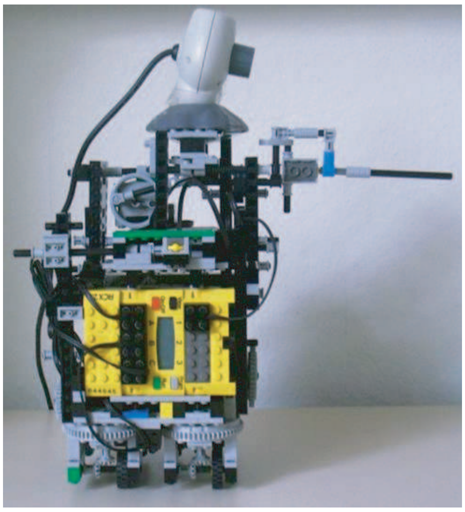

Figure 3.11: The final version of TINAH, built using Lego Mindstorms$^{\mathrm{TM}}$.

used to control the robot. The robot is based on a synchro-drive, which can move in all four directions without changing the orientation. The final form of TINAH can be seen in Figure 3.11. In the figure, the webcam, which was used to take pictures of the pecking discs, and the pecking mechanism are visible.

3.3.2.1 An Exemplar-Based Learning Algorithm

The learning algorithm that we implemented was an exemplar-based one. Exemplar-based learning is a subset of lazy learning, which is a general term for such machine learning methods as case-based, instance-based, and memory-based learning.[6] Lazy learning systems are characterized by the following three aspects (Aha, 1998):

- They do not process their inputs until given information requests.
- They respond to requests by combining information from the stored data.
- They dismiss any temporary intermediate results created during problem solving.

In contrast to lazy learning, eager learning signifies approaches which build data structures when new input data is received, discard the new data and base any responses on this data structure, which involves an abstraction from the input data.[7] One can make a clear binary distinction among the lazy learning

[6] Instance-based, memory-based and exemplar-based methodologies in effect signify the same thing, hence a distinction will not be made here.

[7] For a comparison of rule- and exemplar-based learning mechanisms see Clark (1990).

methodologies. On the one hand, there are systems that base any new categorizations and decisions on a certain subset of stored processes of decision making that have already taken place, chosen according to their similarity to the current case. Case-based learning, which is a widely applied methodology in the field of expert systems, refers to such systems (Riesbeck and Schank, 1989). In exemplar-based systems, however, when a decision has to be made, all the exemplars in the memory are effective to the degree that they are similar to the current case.

One of the most important aspects of an exemplar-based learning system is the level of representation of the exemplars and the similarity measure, which is closely related to the form of representation. In case-based reasoning, the representation of the input data to the learning system already provides the necessary separation into features[8]. Consequently, the representation of stimulus material is decisive, and a high level representation serves to blur the distinction between a rule-based and exemplar-based (or similarity-based) system (Hahn and Chater, 1998). Another, and more important, consequence of such a separation is that the agent is supplied with the categories and correlations that are significant, at least from the perspective of the designer. This is a breach of the autonomy of the agent, because it feeds to the agent what it actually should learn itself. What is or can be useful is provided to the agent from the perspective of the designer when such a strict *a priori* partitioning of the stimulus material is done. Also, it is not clear whether the partitioning by the designer is actually the most efficient one. One more important point has to do with the parsimony of the model. Avoiding the extraction of patterns by the designer "also avoids the problem of fixing certain categories of features and non-features, which leads to a more parsimonious theory" (Deiwiks et al., 2003, p.17).

The learning mechanism employed in TINAH used exemplars that consisted of the input data from the sensors, the action undertaken, and the reinforcement that ensued. Therefore, an exemplar can be represented as $a = [s_t, a_t, r]$. There was only one source for sensory data, which was the camera. A webcam was used to gather images inside the box. These images were then stored in raw form in exemplars. Raw form corresponds here to an image file that is encoded in a certain color space. The available color spaces were RGB, XYZ and L*a*b*, which are all designed by the CIE (*Commision Internationale de l'Eclairage*). Among these, the L*a*b* space was constructed primarily to mirror the color perception of humans as found in psychophysical experiments. Although there is no data regarding color perception in hens, we picked the L*a*b* color space for color coding, because it is the only space relevant to some kind of natural data.[9]

[8] "In CBR, instances are typically represented using more rich symbol descriptions, and the methods used to retrieve similar instances are correspondingly more elaborate' (Mitchell, 1997).

[9] See van Dam et al. (1997) for a detailed treatment of color spaces and Section 3.4.2 of Deiwiks et al. (2003) for detailed information on the processing of image information in the TINAH robot.

The similarity measure we employed was based on pixel-wise comparison. The distance of two images I and I', where the size of the images is N pixels per M pixels, is the average Euclidean distance per pixel:

$$|I, I'| = \frac{\sum_x^N \sum_y^M |I(x,y), I'(x,y)|}{N \cdot M} \tag{3.1}$$

In this equation, $I(x,y)$ corresponds to the pixel with the cartesian coordinates x, y in image I. The similarity of two exemplars is then the inverse exponential of this distance:

$$d(I, I_i) = e^{-|I,I_i|} \tag{3.2}$$

This is an application of Shepard's universal law of generalization (Shepard, 1987), which claims that in a psychological space, generalization gradients fall exponentially with increasing distance. This psychological space should be scaled with an appropriate metric, which in our case is the Euclidean distance between images. [10] The psychological space is color in our case, and the distance metric is the Euclidean distance.

The exemplar-based learning algorithm developed for TINAH uses a set of behaviors $A = \{a_1, a_2, \ldots, a_n\}$ as the set from which to choose one for execution. In the case of TINAH, we had only two behaviors because the operation of TINAH was limited to moving inside the box and pecking. This produced the set of behaviors $A = \{\texttt{move},\texttt{peck}\}$. The response strength R_a of a behavior with respect to a database E of exemplars, which consists of n exemplars, is then calculated from the subset of exemplars that have the same behavior, as follows:

$$R_a = \sum_{i=1}^{n} d(I, I_i) \cdot r_i \tag{3.3}$$

Here, I corresponds to the current image data retrieved from the camera, and I_i corresponds to the image data of the ith exemplar. The response strength of each behavior is calculated in case a decision has to be made. Equation 3.3 makes clear that each exemplar in the memory takes effect in case of a decision. In order to calculate the execution possibility for each behavior in A, the response strengths are weighted:

$$P(a' = PECK) = \frac{R_a}{\sum_{a \in A} R_a} \tag{3.4}$$

This weighting process produces a probability distribution between 0 and 1, and the sum of the execution probabilities of all behaviors is unity.

The selection of which behavior to execute, once the execution probabilities are determined, is straightforward. A random number between 0 and 1 is drawn,

[10] Shepard (1987) also supports the validity of a Euclidean metric: "For unitary stimuli, such as colors differing in perceptually integrable attributes of lightness and saturation, the closest approximation to an invariant relation between generalization data and distances has uniformly been achieved in a space endowed with the familiar Euclidean metric" (p.1319).

and the PECK behavior is activated if its execution probability is less than the random number. The selection process is so simple because there are only two behaviors. A process for the selection of a behavior from among more than two behaviors will be explained in Chapter 4.

3.3.3 The Results of the Experiments

The robot hen was tested in a set of training experiments that used the same equipment and stimulus material as in the hen experiments depicted in Figure 3.1. Due to time restrictions and technical problems with the hardware, we have been able to carry out only two training phases with our robot. These experiments provided enough data to compare the model with the behavior of the hens, but not enough to carry out elaborate statistical analysis as was the case with the hens. The results of the training phases carried out with the robot can be seen in Figure 3.12. There are two important results that are worth pointing out. The first is that the robot learns. In the first training phase, the robot reached a correct pecking rate of 60%. In the second training phase, the correct pecking rate increased to 70%, which meant 10% within-session learning. This result shows a good accordance with the chicken data, which is 63% correct in the first session and 71% correct in the second session (Werner et al., 2004).

The second important result is that the robot learns color better than line orientation. In the first session, the model reached 70% correct trials for discriminating the correct color, compared to 50% for discrimination of line orientation. In the experiments with chickens, the subjects pecked 82% correct for color discrimination and 44% correct for line orientation. In the second session the rate of correct pecks by the robot increased to 80% for color and 60% for line orientation, whereas chickens improve their performance to 89% for color and 53% for line orientation. The performance for color discrimination is still better than for line orientation for both the robot and the chickens. In both trials, however, the performance of the robot in line orientation is better than the hens. The reason for this is probably the nearly perfect centering behavior that moves the robot so as to have the pecking disc at the center of the image taken by the camera. Ideally, this centering behavior should be acquired, like the pecking behavior, through an autoshaping procedure, similar to the one for the hens.

These results are naturally very simple, and are not reason for early enthusiasm. Nevertheless, we have reason to expect that further experiments to be carried out will reveal similarities with the hen data. This expectation is based on a test run of the similarity measure on sample data in the form of pictures taken with a camera. The similarity measure judged separable compound stimuli to be most similar to dimensional color stimuli: for example, a black vertical bar on a red background, taken as a whole, are most similar to a purely red stimulus. Taking the hypothetical case of a color reversal trial where the colors that are reinforced are switched, this would mean pecking on the color that was reinforced before. But as this color is not reinforced anymore, this reaction will be counted as an error, for the robot as well as the chickens. For example, pecking on a black vertical bar on red background will lead to an error, because

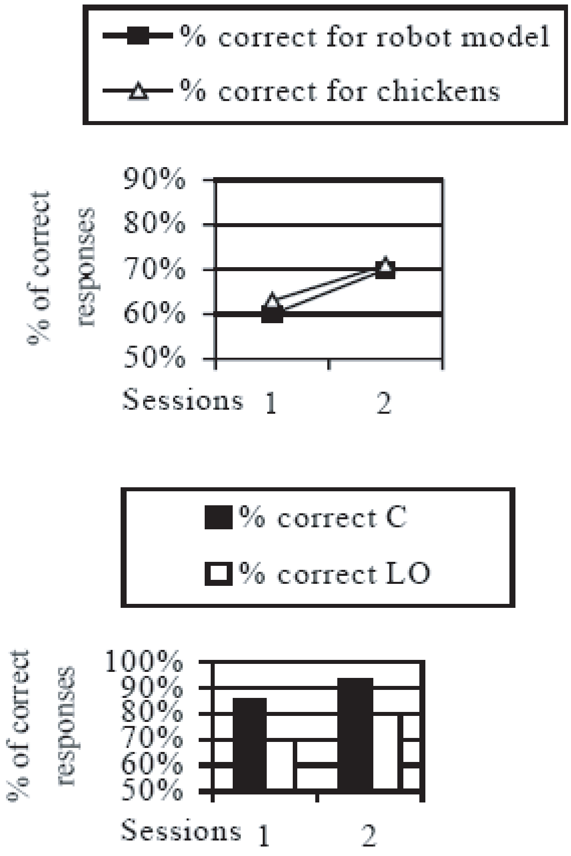

Figure 3.12: The results of the training phases (adapted from John and Werner (2004a)). Top: Rate of total correct pecks. Bottom: Rate of correct pecks for color and line orientation stimuli.

pecking on a purely red stimulus was reinforced before. Integral stimuli (colored lines on transparent background), on the other hand, were revealed to be most similar to dimensional line orientation stimuli: a green horizontal bar, for example, is most similar to a black horizontal bar. Repeating the same behavior for similar patterns will in this case not lead to an error, as the reinforced line orientation was not reversed in color reversal. Thus, after color reversal, the robot model will make substantially more errors for separable compound stimuli than for integral stimuli, just like the hens. The reverse was valid for the case of a line orientation reversal. These results are grounds enough for optimism for reproducing the empirical findings using only a simple type of holistic processing.

3.3.4 Evaluation

The aim of the TINAH project and the robot model was to examine the claim that assuming holistic processing was enough for accounting for the empirical data from the experiments with chickens, by building an embodied model that operated in the same environment with the subjects of the original experiment. This was in accord with the Comparative Cognitive Robotics framework, explained in Section 3.1. To this end, one can count the project fairly successful, keeping in mind the scarce data. The experiments are still in progress, and we believe that even if we have to make major changes in the model, the results will reveal valuable insights into the nature of visual processing in chickens.

Categorization without categories One of the most important characteristics of TINAH has to do with the nature of the processing done by the learning mechanism. The distinction between analytic and holistic processing has been explained in Section 3.2.2: in analytic processing, the stimulus is decomposed into features such as a horizontal line or a red circle, and these features than enter into the categorization mechanism as an input pattern. This kind of processing corresponds to a certain kind of model building in cognitive science and AI, in which the designer builds in all the data structures and processing mechanisms that correspond to the entities to be learned or the phenomena to be explained. The behavior of the designed agent is then evaluated in the light of these constructs, as if they were not designed in the first place by the programmer or engineer. Smith (1996) calls this an *inscription error*: "a tendency for a theorist or observer, first, to write or project . . . a set of ontological assumptions onto a computational system . . . and then, second, to read those assumptions or their consequences back off the system, *as if that constituted an independent discovery or theoretical result*" (p. 50, emphasis in the originial).This is not the case for TINAH: although no categories, or programming constructs that could correspond to categories were built in, the robot showed category acquisition. We called this *categorization without categories*. The computational system does not first search for certain distinguishing features in the camera input and then map them onto built-in categories; each pattern is represented on itself (without being matched to a node), and differential responding is a result of a comparison operation realized on the whole database. The categories that emerge are not a result of a correct formulation by the designer of the agent, but a result of the increasing number of exemplars and the effect of the reinforcement received following the actions performed. That is, the interaction with the environment is the main source of the categorization behavior. This guarantees two things. The first is the autonomy of the agent: as has been pointed out earlier, autonomy means the reliance of an agent on its own existence, and the dependence of categories solely on the situatedness of the agent leads to highly autonomous behavior. The second important advantage of *categorization without categories* pertains to parsimony. Computationally, our agent can be seen to be less parsimonious than a program that uses features, which would be more efficient and build smaller data structures. Because all the cases of pecking are stored, TINAH ends up building considerably large databases. When regarded from the point of view of how much designer knowledge is built in, however, our model is more parsimonious than a feature-based model.

3.3.4.1 Criticism of CCR

The CCR framework restricts embodied models to empirical data gathered from natural subjects. The behaviors relevant for the model are consequently those which actually produced this data. In the case of TINAH, the only important – one can even say scientifically extant in the context of this model – behaviors were pecking and moving away, or not pecking. The unity of the organism becomes a non-issue; instead of Brooks' functional circle mentioned earlier, we have a reduced environment similar to a microworld, e.g. to the blocksworld.[11]

The emphasis put on replicating empirical data also causes an unpleasant commitment to replication, which comes for a price. The robot used in this project had nothing in common with the hens which took part in the experiments: they were completely different beings, of which we claimed the one was a computational model of the other. In this case, expecting – wishing for – identical performance is far-fetched. However, fortunately for the engineer in the cognitive scientist, there are parameters that one can tweak, just as in every model. Playing with the parameters to achieve the exact same performance, one is displaying what Lehnert (1989) calls the TWITIT methodology: Tweak It Til It Thinks, that is, play with the parameters so long until you get the performance of your model so close to the real subjects as possible.

3.3.4.2 A similar model by Steels and Kaplan

After we have developed our model, we have discovered a study using a very similar methodology. Steels and Kaplan (2001) have opted for an exemplar-based system with holistic processing in their model of social learning of lan-

[11] The problem goes actually deeper, and has to do with the methodological principle of animal and cognitive psychology to deal with meaningless stimuli. This commitment to meaningless atomic stimuli is problematic, because for living beings all stimuli are meaningful. The scope of this discussion is outside this thesis, but for a broader treatment see Clancey (1997), especially Chapter 4.

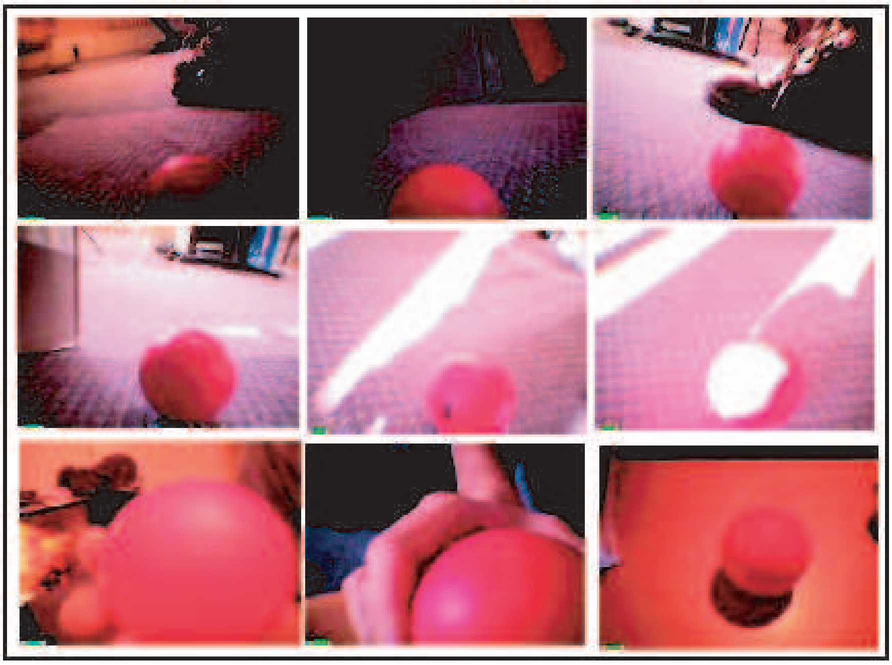

Figure 3.13: Sample images of a red ball from the experiments on social learning by Steels and Kaplan (2001).

guage. They also used a pixel-based similarity measure for category learning. While their model at first was based on the standard RGB color space used by computer cameras, they later switched to the L*a*b* space and report an improvement of the results. Using a similar pixel-based similarity measure as the one used in the EROSAL project, Steels and Kaplan try to develop a robot model of learning words for single objects (names). Like Steels' earlier studies, the experiments are again constructed as "games". The game the robot is engaged in this time is called "categorization game". The robot (in this case a Sony AIBO$^{\mathrm{TM}}$robot dog) is shown an object by a human experimenter and is expected to classify it by uttering the same name that was uttered by a human instructor before. If the robot successfully classifies the object, it gets positive reinforcement. In case of error, the experimenter gives negative feedback. An associative memory stores relations between object views and words. Note that the object views are not matched on a single category which is then associated with a word, as Steels' earlier models would assume. Rather, an input pattern is directly associated with an output pattern (here, a name), without an intervening abstraction step. This is what has earlier been called *categorization without categories*. As can be seen in Figure 3.13, the views which are associated with the same name ("ball") are actually quite different. Extracting their common, invariant features would be a very difficult, if not impossible task. And without pre-coded detectors for the “relevant” features and other constraints implemented by a human engineer, we could also not be sure that the “correct” defining features are associated with the name “ball” (e.g., features of the floor might be included in the definition). However, building in such pre-coded mechanisms would contradict the goal of behavior-based AI to construct robots that are autonomous, and it would also not lead to a realistic model of categorization and naming. The authors state that “[t]he different views of an object form an implicit category, based on the fact that they are named the same way”. At the end of the experiments, the use of appropriate words, when averaged through one whole experiment including the training trial, was 80%. There is one important drawback of their model, however: Steels and Kaplan (2001) used pre-coded names for objects within their model. While the category is constructed by the robot itself by assembling object views, the robot has ready-made detectors for all object names that might possibly occur in the experiments, and simply attaches the respective pre-coded label to the current object view.

Chapter 4

Behavioral Categorization

The TINAH robot (and the EROSAL project in general) stressed two points: the role of empirical data and testing the model in the same environment in which the subject performed. Behind these two principles, there is one more general principle: taking as the source of the definition of intelligence (or natural intelligence) and learning the psychologically quantified behavior of animals in strictly controlled environments. The only measured behavior of the hens in the experiments that led to the robot model was the pecking behavior. This meant that the robot model was also restricted to this one behavior, where the model refers not only to the behavior system, but also to the morphology of the robot.

TINAH demonstrated that one does not need categories inside the agent to be able to talk about categorization. Another group that emphasizes a behavioral description of categorization is the AI Lab at the University of Zurich. In the next section, some categorization experiments done at this laboratory will be explained. The setup and the general idea of these experiments were taken over in the Khepera experiments that are explained in the second section of this chapter.

4.1 The Sensory-Motor Coordination Experiments

Pfeifer and Scheier (1997) point out that the attribution of categorization behavior to a natural agent does not need to stem from categories inside the head of the agent. This means that "if an agent consistently displays one kind of behavior when it encounters one type of object but not when it encounters other objects, it is reasonable to talk about categories of the agent" (ibid.). They therefore implemented an autonomous agent that categorizes objects in the environment without building in any pre-programmed categories or nodes. Another feature of their system is its being inspired by sensory-motor coordination. Unfortunately, they don't provide a definition of sensory-motor coordination, except for a quote from Dewey, which claims that "it is the movement which is primary, and the sensation which is secondary" (Pfeifer and Scheier, 1997). The idea of sensory-motor coordination is also presented without any clear empirical support, which would justify the implemented computational mechanisms. Therefore, despite the name of their article (*Sensory-motor coordination: the metaphor and beyond*), the role of this principle does not get beyond a metaphor. Viewing classification as a sensory-motor coordination process is achieved by including the robot's own actions into the classification process (Lambrinos and Scheier, 1996).[1] Another feature of their work is that action selection and categorization are not seen as separate problems: categorization is seen to be bounded closely to action selection, because it is evaluated behaviorally.

Even as a metaphor, the idea of integrating action into categorization has an important computational advantage. Sensory-motor coordination offers a solution to the problem of *perceptual aliasing.* Perceptual aliasing refers to the association of similar sensory patterns to different actions (Whitehead and Ballard, 1991). In order to differentiate between such similar patterns, the agent can make use of its actuators to manipulate the situation and enable a more favorable sensory input pattern. This is called *active perception* (Bajcsy, 1988; Nolfi and Marocco, 2002). Sensory-motor coordination also facilitates the extraction of regularities from the sensory input. In many cases, regularities in the sensory pattern which could lead to correct mappings to output patterns are hidden and should be extracted by a transformation. One way to do this transformation is to recode the sensory pattern, whereas another way is to use sensory-motor coordination. [2]

The SMC experiments can be divided into two phases, with important differences between the underlying computational systems. The two phases will be explained here shortly, with emphasis on the learning mechanisms.

4.1.0.3 First phase of SMC experiments

The SMC experiments were carried out using a Khepera robot[3]. The first group of SMC experiments (referred to as SMC I from here on) involved the classification of pegs of two different sizes (Scheier and Pfeifer, 1995). See Figure 4.1 for a depiction of the experimental setup. A hook was attached to the back of the Khepera robot for picking up small objects. This hook was big enough for a group of the pegs, and too small for the rest of the pegs. The robot could therefore pick up only some of the pegs, by circling around them, and getting them inside the hook.

[1] Because the authors stress sensory-motor coordination as a primary feature of their robots, their experiments will be referred to as the SMC experiments from here on. However, what should be attributed to this principle is not seen to be more than the inclusion of action in the categorization process.

[2] For a more detailed treatment see Nolfi and Parisi (1999). The authors refer to the transformation of the sensory space and the perceptual aliasing problem as two different things. The two issues are related, in that the sensory-motor transformation of the sensory space is actually a solution to the perceptual aliasing problem.

[3] See Section 4.2 on page 74 for a description of the Khepera robot, and Figure 4.3 on page 74 for a picture of the robot.

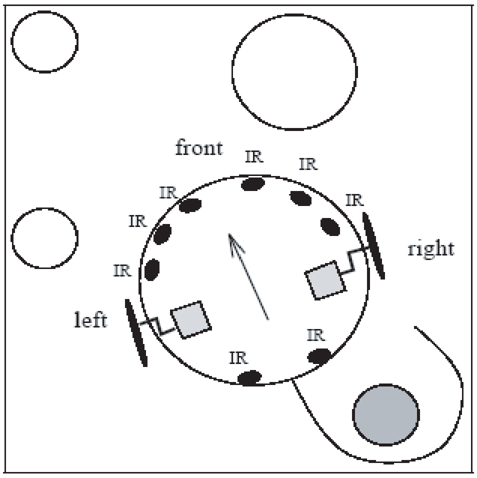

Figure 4.1: The experimental environment in the first SMC experiments (Scheier and Pfeifer, 1995).

Scheier and Pfeifer (1995) claim that the demonstration of a behavior does not mean that the behavior is actually represented inside the agent: in order to have wall-following behavior, one does not need a wall-following module. Therefore, the authors opt for a processing scheme called the Extended Braitenberg Architecture (EBA, from here on)[4]. In this architecture, one still has behaviors, called processes, but the difference is that they are all continuously active, and what changes is the effect they have on the actuators (see Figure 4.2 for a diagram of EBA). As shown in the diagram, each of the processes active take input from the sensors and produce values that are written to the actuators. The values from different processes are then summed to get the final activation value for the effectors. In the first experiment, the only effectors are the wheels, and the speed values of the two wheels are simply the sums from different processes:

$$s(t) = (s_l(t), s_r(t)) = \left(\sum_{i=1}^{N} o_i^l(t), \sum_{i=1}^{N} o_i^r(t) \right) \tag{4.1}$$

In Equation 4.1, $s_l(t)$ and $s_r(t)$ correspond to left and right wheel speeds at time t, respectively. Similarly, $o_i^l(t)$ and $o_i^r(t)$ correspond to the output of process i for left and right wheels at time t, respectively. The processes produce values for writing to the actuators. The process `avoid obstacle`, for example, determines the values it adds to the wheel speeds as follows:

$$o_{ao}^l(t) = \sum_{i=1}^{3} \phi_i IR_i(t) - \sum_{i=4}^{6} \phi_i IR_i(t) \tag{4.2}$$

[4] See Lambrinos and Scheier (1995) for details of EBA, and for a forerunner of it see Steels (1995b).

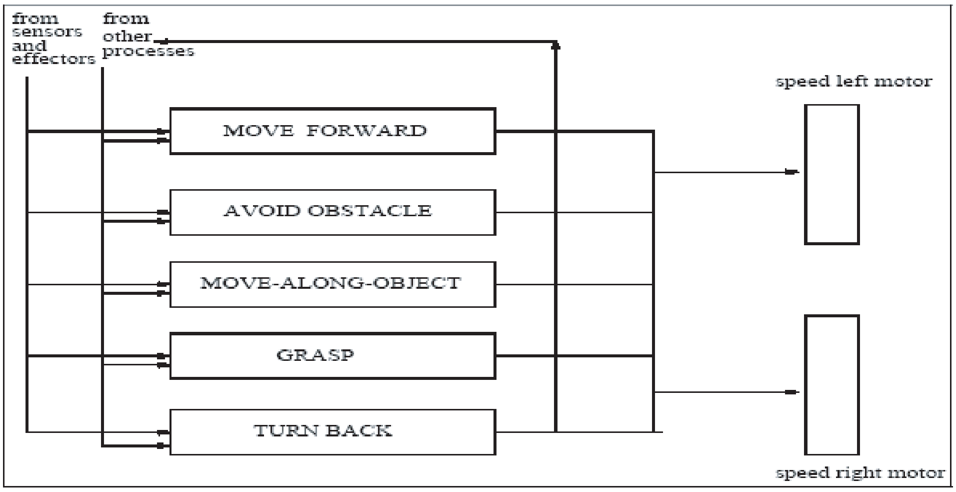

Figure 4.2: Schematic diagram of the extended Braitenberg architecture (adapted from Scheier and Pfeifer (1995)).

$$o^{r}_{ao}(t) = -\sum_{i=1}^{3} \phi_{6-i+1} IR_i(t) + \sum_{i=4}^{6} \phi_{6-i+1} IR_i(t) \tag{4.3}$$

In Equations 4.2 and 4.3, ϕ_i corresponds to a parameter determining the effect of the infrared sensor IR_i on the output of the process.

The only mapping that had to be tuned, i.e. changed by learning, was the effect of the `grasp` process on the actuators. In case a reinforcement was received, the angular velocity vector, which corresponded to the motor values, was associated with the quantity Q which determined how strongly the motor values were controlled by the `grasp` process. A positive reinforcement was determined by the existence of an object in the hook behind the robot, as sensed by a proximity sensor. A simple Hebbian network was used for learning. Scheier and Pfeifer (1995) claim that expecting the robot to learn by chance is unrealistic, because it would take too long, and a bias is therefore introduced into the system. This is done through the `move along object` process, which caused the robot to circle around an object, and in the meanwhile also increased Q. Circling around the pegs, with the execution of the `move along object` process, was a reflex action; what was to be learned was not to circle around bigger pegs, so as to pick up only the small pegs.

In the experiments carried out, the robot learned trying to grasp small pegs, and not grasping the bigger pegs. Because circling around the pegs was a reflex action, the robot kept circling around the bigger pegs, although it did not try grasping them. In order to avoid this, a reflex (called "heuristic" by Scheier and Pfeifer (1995)) was built in, which caused the robot to stop circling around the peg once learning had occurred. In the experiments, the robot encountered

50 small and 50 large pegs per trial over 10 trials. Over all trials, the robot grasped 100% of the small pegs and 10% of the small pegs.

4.1.0.4 Second phase of SMC experiments

The second group of SMC experiments (SMC II) involved a different setup with pegs of the same size (Scheier and Lambrinos, 1996; Pfeifer and Scheier, 1997). The two types of pegs were distinguished according to the existence of a texture on them, and the pegs with a texture were conductive, due to a metal band around them. A camera mounted on the robot was used as a visual sensor. The robot was also equipped with a gripper extension, which could be used to hold and lift the pegs, and measure their conductivity. In addition to the hardware extensions, a simulated fovea was implemented.

In SMC II, the Extended Braitenberg Architecture was preserved. What was changed was the object-related processes (i.e. `move along object`, `grasp`) which were replaced by a haptic and a visual system. Both of these systems consisted of a sensory map, an attention map and a feature map. The sensory maps supplied sensory information to the feature and attention maps. The attentional maps were coupled to the effectors, which made the robot orient itself to any interesting stimulus, interesting being defined with bright spots. Together with the effectors and the fovea the visual attention map formed a sensory-motor loop: it brought the robot to relevant places in the environment while at the same time keeping the eye focussed on the spot where the robot was heading to. The attention loop created by the haptic system made the robot focus on the object by turning towards it.

Feature maps, which distinguished certain features in the environment, were connected via modifiable feedback connections to the attention maps. One important feature is the connectivity of the haptic and visual feature maps. These maps were connected via reentrant connections, and the basic categorization mechanism involved the correlation of the signals of these feature maps. Learning was achieved by updating the weights of the connections between the feature maps, using a Hebbian learning scheme. The feature maps could inhibit or activate the attention maps, whereby the attentional sensory-motor loops formed by the attention maps, effectors and sensory maps could be either enhanced or broken down. The learning process was modulated by a value map, which was connected to the conductivity sensor of the gripper. Learning took place only when there was a conducting object present in the gripper. In essence, the result of learning was that relevant (i.e. conductive) objects enhanced activity in the attentional loop while it was broken in the case of uninteresting objects.

In the experiments carried out with the SMC II model, the robot learned not to inspect the non-conductive/non-textured pegs. This took the robot on average 12 encounters with non-conductive objects (Scheier and Lambrinos, 1996). At the beginning the robot was inspecting, i.e. taking into the gripper and measuring the conductivity of, all the pegs. After the robot acquired inhibiting the haptic and visual attentional loops when it got close to a peg without texture, it stopped exhibiting the orientation behavior towards them.

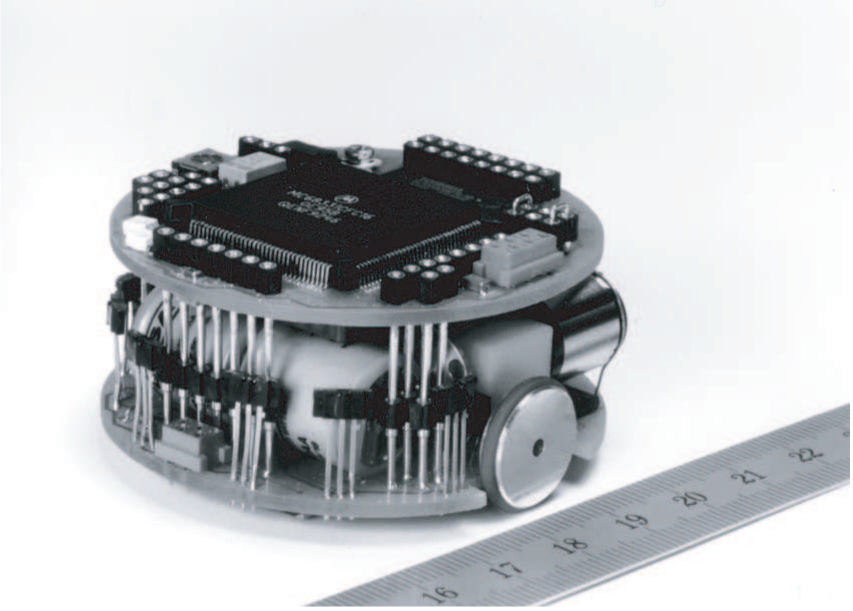

Figure 4.3: The Khepera$^{\mathrm{TM}}$robot.

4.2 The Khepera Robot

The Khepera robot is developed by the K-Team from Switzerland (K-Team, 1999b). Due to its compact size, versatility, and robustness, it has been used in many experiments in embodied AI and machine learning. Figure 4.3 depicts the Khepera robot used in the experiments.

The Khepera robot is 32 mm high, 55 mm in diameter, and weighs 70 grams. It has eight infrared sensors, six of them on the front and two at the back, which can serve as ambient light sensors or proximity sensors. The robot has two wheels, which can be controlled separately. In order to go forward, the wheels are set to the same speed, whereas turning is achieved by assigning different speeds to the wheels. Like the Lego RCX, the Khepera has an on-board memory which can store programs of up to 256 Kilobytes, but one can also control the robot with a program on the computer. The processor is a Motorola 68331. The robot runs either on the rechargeable NiCd batteries that can drive it for half an hour, or connect to an external power source. When the robot has to use an external power source and connect to the computer to be controlled by a program, it has to connect to an intermediate unit, which connects to the RS232 port of the computer and to the power line. The communication between the computer and the robot is done by passing text messages through the serial communication line.

The original Khepera robot can be extended by attaching additional turrets to it. This is done by connecting the turrets onto the pins at the top of the robot, which stabilize the turret and at the same time serve for the communication between the turret and the robot. The turret can be controlled and sensory values from the turret can be read using the same serial link with the robot. One of the possible extensions is the gripper turret (K-Team, 1999a). The gripper turret can be used as an arm to pick up objects of up to around 50 grams. The sensors on the gripper are an optical barrier inside the gripper for object detection and an electrical conductivity sensor for measuring the conductivity of the gripped object.

The program controlling the robot has been written in the Java programming language. Java is currently one of the most widely used object-oriented languages, and there is strong community and corporate support behind it, providing programming tools and resources. The external libraries used in the programming of the robot included the Java Communications API, which provided classes for communicating through the RS232 port[5] and the Khepera API by Pär Spjuth[6], with some minor modifications.

4.3 Behavioral Category Acquisition Experiments

The main idea of the experiments reported in this thesis was to replicate the SMC experiments, albeit with a simpler learning mechanism. The experiments involve the autonomous learning of categories by a robot, where the acquired categories are behavioral, because they do not correspond to any entities built in by the experimenter. In the following, the setup and the details of the software will be explained.

4.3.1 The Setup and Software

The setting involved simple wooden blocks inside a field delineated with wooden slabs, similar to the setting in the SMC experiments. The learning mechanism is again an exemplar-based one, due to the reasons explained in Section 3.3.2, and is very similar to the one programmed for the EROSAL project. The differences are due to the differences between the environment in which the robots operated, and the kind and number of behaviors necessitated by the tasks. TINAH operated in a Skinner box in which it had to go only left and right, and decide to peck or not. The tasks the Khepera robot had to carry out were more complex. In contrast to TINAH, it used more behaviors, and the reinforcement scheme was more complicated.

The main constituents of the learning mechanism are as follows:

- Distance measure

[5] The Java Communications API can be downloaded from `http://java.sun.com/products/javacomm/downloads/index.html`.

[6] The Khepera API can be downloaded from `http://hem.passagen.se/foggy/khepera/index.html`. A detailed documentation is also available at the same web site.

- Arbitration mechanism
- Behavior selection strategy
- Database
- Behaviors

In addition to these, there is a reinforcement scheme, which runs as a part of the experiment.

The distance measure computes the similarity of two states. A state, which is a representation of the condition the robot finds itself in at any given moment, contains data from the following sensors packed into an array:

- Resistivity
- Arm position
- Presence of object
- Proximity sensors
- Ambient light sensors

In case a gripper turret is not attached to the robot, the states do not contain the first three entries, because sensors on the turret are responsible for supplying these data. The similarity mechanism takes in two such states as arguments and returns the distance between them. The Euclidean distance measure was used in the Khepera experiments, as in the TINAH experiments[7]. The distance between two states I and I', which both consist of an array of integers with the ith element being I_i, is as follows:

$$d(I, I') = \sum_{i=1}^{n} (I_i - I'_i)^2 \tag{4.4}$$

The similarity of two images is then the exponentially weighted distance:

$$s(I, I') = e^{-d(I, I')} \tag{4.5}$$

The database consists of exemplars stored by the arbitration mechanism. Each exemplar is of the form $m = \{s_t, b, r, s_{t+1}\}$. Here, b refers to the behavior that executed before the exemplar was formed, r to the reinforcement, and s_t and s_{t+1} to the states in which the robot found itself when the behavior started execution and was finished, respectively. An exemplar is formed only when nonnegative reinforcement is received. When a behavior finishes, a new behavior is picked in the following way. For each behavior b, the exemplars in

[7] Most of the equations presented here are actually very similar to the ones presented in Section 3.3.2 on page 58.

the memory which have the behavior b are collected in a set M_b. The response strength for behavior b is then calculated as follows:

$$R_b = \sum_{m \in M_b} s(I_m, I) \cdot r_m \tag{4.6}$$

In Equation 4.6 I_m stands for the first state in the exemplar m, I for the current state, and r_m for the reinforcement of exemplar m. The calculated strengths are then weighted to arrive at the execution probabilities for each behavior b in the set of behaviors B:

$$P_b = \frac{R_b}{\sum_{b \in B} R_b} \tag{4.7}$$

All behaviors now have a probability of execution attributed to them. These probabilities range between one and zero, and add up to unity. In order to pick one, the behaviors are ordered randomly on the line of real numbers (the ordering is not important, as will be obvious). A random number between zero and one is drawn, and the behavior which lies after the random number on the real number line is chosen for execution. In this method, the probability of a behavior being picked is proportional to its execution probability. The pseudo code for this routine is as follows:

```
Sum of probabilities $S$
Random number $R$
for All behaviors do
  Add the execution probability of the behavior to the sum of probabilities
  if The sum exceeds the random number then
    Pick the behavior for execution
  else
    Move on to next behavior
  end if
end for
```

The controlling software is essentially an infinite loop that executes a behavior, stores an exemplar in the memory when necessary, and then picks a new behavior to execute. These tasks are carried out by the arbitration mechanism, embodied in the software class `Arbitrator`. The `Arbitrator` picks out a behavior –from among the set of behaviors with which the experiment started– using the similarity measure, the behavior selection strategy and the database. The behavior selection strategy returns a mapping of behaviors to strengths. This corresponds to Equation 4.6. The components that were changed through the experiments are the behavior selection strategy, the behaviors and the reinforcement scheme. The distance measure at the beginning was the Euclidean distance, and was not changed. The behavior selection mechanism was, at the beginning, based on the inverse exponential of the distance of two states (see Equation 3.2).[8]

[8] See Atkeson et al. (1997) for a comparison of different weighting functions.

4.3.1.1 The problem of negative reinforcement

In the case of a negative reinforcement, the most plausible thing to do is to store an exemplar with the reinforcement value of -1. The problem with using a negative value of reinforcement is that it corresponds to deleting the previous values of positive reinforcement. When an exemplar with negative reinforcement is stored, its mathematical effect is diminishing the effect of any exemplars with positive reinforcement that have been stored earlier. The phenomena of extinction is revealing in this context. In the psychological phenomena of extinction, a stimulus is presented continuously without an unconditioned stimulus with which it was paired earlier. The conditioned response to the stimulus is lost after repeated presentations. Psychological evidence shows that extinction does not involve the unlearning of a conditioned-unconditioned stimulus association, but rather new learning of the inhibition of the association (see (Domjan, 1998, p.82)). The method used for implementing negative reinforcement was as follows: if a behavior received negative reinforcement when the robot was in the state s, one exemplar with each behavior other than the one that has to be negatively reinforced was stored. These exemplars had as first state the same state and a positive reinforcement of $1/(n-1)$, where n is the total number of behaviors active in the experiment. This way, the effects of an exemplar are not removed, and the robot can still learn.

One problem was the signaling of reinforcement. The path chosen was to write different reinforcement sources (called `Notifiers` in the program) which signaled reinforcement when the conditions they were controlling were satisfied. For example, the reinforcement source for obstacle avoidance gave a negative reinforcement when there was a proximity sensor value higher than a certain threshold. Another notifier gave a positive reinforcement when the distance travelled was greater than a threshold. Whether the agent should be getting reinforcement for what it is doing, and the value of this reinforcement, should actually be evaluated by an entity independent of the sensors of the robot, although the agent picks up the reinforcement with its sensors. Ideally, the mechanism that supplies the reinforcement should run on another computer, independent of the control program that runs the robot, and the evaluation of the behavior of the robot, with respect to whether giving a reinforcement or not, should be independent of the sensors of the robot.

4.3.2 Object avoidance

As a first task in order to test the potential offered by the above described system to learn, the well-known task of obstacle avoidance was selected. Obstacle avoidance is, from an engineering point of view, very easy to program into a robot. Similar to a Braitenberg vehicle that follows or avoids light (Braitenberg, 1984), one need only make the proper direct connections between the sensors and the motors. In the case of avoiding obstacles, this corresponds to adding a certain value proportional to the values of proximity sensors on one side to the motor values on the same side. *Learning* this kind of a coordination is a different matter. In this section, two models of learning such behavior are presented, with the second model improving on the first one. The models differ in the behaviors they employ, and the way negative reinforcement is given and processed.

4.3.2.1 Model 1

In the first model, the set of behaviors included behaviors that were driving forwards, backwards, or veering to the side. There were different versions of these behaviors with varying speed and duration. The reinforcement scheme was supposed to give positive reinforcement proportional to the distance travelled when the robot did not bump into an obstacle, and negative reinforcement if it bumped into one. Due to reasons explained above, there was no negative reinforcement. In the first model, any negative reinforcement from the notifier was ignored, and no exemplar was stored.

The reinforcement scheme, embodied in the reinforcement notifier, was as follows:

- Give zero reinforcement if the robot gets too close to (or bumps into) an obstacle.

- Give a reinforcement proportional to the distance travelled by the robot whenever a behavior moves the robot and there is no collusion.

This scheme was to impart to the robot the ability not to drive against obstacles, and travel forward when possible. The aim was to find out whether the robot would learn to execute the behaviors that moved it forward when there were no obstacles in front of it. The robot was expected also to learn to use the behaviors that move it a large distance when the sensors registered minimum values of obstacles, and the slower ones when there was a certain presence of obstacles.

The robot did not exhibit any learning with this configuration. Once certain behaviors that took the robot a long distance forward were executed a few times and acquired high reinforcement values, these behaviors were executed even in the cases where they would cause the robot to bump into an obstacle. Behaviors that would take the robot out of such impasses did not acquire enough reinforcement to have high probabilities which would make them a more probable choice. This pointed out that behaviors which aid the robot to get out of bump situations should receive reinforcement in such cases, which would cause them to get favored in similar situations. Another point is negative reinforcement. If there is no negative reinforcement in any form, positive reinforcement that was received in a situation is effective also in irrelevant situations, which means that a mechanism for showing that behaviors other than the ones that cause negative reinforcement are more preferable is necessary. The nonexistence of negative reinforcement avoids the differentiation of behaviors according to situations.

4.3.2.2 Model 2

There were a number of problems with the first model. One of these was the overly differentiated behaviors. This is not parsimonious. One alternative is using just atomic behaviors, which then can be combined. Instead of a behavior that takes the robot a long distance, for example, the moving forward behavior can be executed multiple times. The next model therefore involved atomic behaviors, i.e. behaviors that were relatively short and did not have counterparts that were longer or faster. There were four of these behaviors: going forward, backing up, turning left and turning right. Turning left and turning right did not move the robot forward. Moving forward, in case no obstacle was on the way, produced positive reinforcement. If an obstacle was bumped into, negative reinforcement was given, and the turning behaviors produced no reinforcement whatsoever. The reason for this is that if the robot ran into an obstacle with a behavior, turning at the same place would not get it out of there. In the second model, a negative reinforcement for a behavior meant a positive reinforcement for each of the other behaviors with the same data as in the situation in which the behavior gets a negative reinforcement, as explained above.

In order to explain the reinforcement scheme, one could use the terms of negative reinforcement zone and empty zone. A negative reinforcement zone is an area where at least one of the sensors register a proximity reading higher than the collusion threshold. An empty zone is a place where the robot has not collide into anything. The reinforcement notifier used for the second model had to give a positive reinforcement to the robot when it either travelled in an empty zone without entering a reinforcement zone, or when it successfully got from a negative reinforcement zone into an empty zone. Negative reinforcement was given by the notifier when the robot got from an empty zone into a negative reinforcement zone, but not when it stayed in a negative reinforcement zone. The reinforcement scheme was accordingly as follows:

- If the robot was in an empty zone (i.e. a place where there are no close objects) but then got into a negative reinforcement zone give negative reinforcement.

- If the robot was in a negative reinforcement zone but did not get out of it, give no reinforcement.

- If the robot was in a negative reinforcement zone but then got out of it, give positive reinforcement.

- If the robot was not in a negative reinforcement zone and did not get into one in the meanwhile, give as reinforcement the distance travelled (1 for the forward behavior, zero for anything else).

One important problem was that because the sensors have very low resolution and signal closest distance even when the robot is as far away as two centimeters from the wall, there is no difference no matter what the robot does once it's close to the wall: it is always getting negative reinforcement. This avoids any differentiation between the different behaviors, because none of them can take the robot out of the zone where it gets negative reinforcement. The solution chosen was making the behaviors that move the robot forward or backward faster, so that they cover more distance and get the robot out of the negative reinforcement zone.

The reinforcement notifier, which was working continuously and checking for reinforcement with an interval of 50 milliseconds, was modified to return a reinforcement value at the end of the execution of a behavior. The reason for this change is technical, and has to do with the limitations of the communication channel between the computer and the robot. The reinforcement mechanism has to have access to the sensor values of the robot, and these are retrieved from the robot via text messages. This interferes with the behavior mechanism, and the process of checking for the conditions of reinforcement were much slower than expected. Another problem is that the reinforcement checking mechanism is dependent upon the robot; for example, one cannot reliably know the distance travelled by the robot. Once a behavior is executed that moves the robot forward, it has to be accepted that the robot also moved in reality, whereas that does not have to be the case. The wheels of the robot actually slide on the table when the robot tries to drive against an obstacle which it can not move, and the wheel counters which register the movement of the robot register that the robot actually moved. Ideally, one could have another computer follow the robot on the desk with a camera and provide reinforcement, which can be done in the future. In the object avoidance task, the robot actually has one aim: run the behavior forward as much as possible, and whenever necessary, run other behaviors to get out of negative reinforcement zones, in order to get positive reinforcement. This is what the reinforcement scheme stipulates. As mentioned above, this actually should not be the case: the robot should aim to either cover as much distance as possible, or visit as vast an area as possible on the table. However, this requires the involvement of a reinforcement mechanism running on another computer, for which the time and resources were not available.

The robot exhibited learning in the second model. In cases where it got into a negative reinforcement zone it acquired which behaviors to execute in order to get out of it. The robot learned to back up when it was facing a wall, and to move forward when the wall was behind it. It also learned to travel forward as much as possible when it was in an open plane. Learning here refers to the differentiation of a behavior from the others in situations where it is most favorable to execute it. This differentiation is achieved through the acquisition by a behavior of a high probability of being executed in the relevant situations.

One thing that was tested for was the reinforcement value that should be given to behaviors when a behavior is to be negatively reinforced. If a complete reinforcement (1) is given to all the other behaviors as inhibition of one behavior, the strengths of the behaviors actually converge to a common value; that is, behaviors do not emerge as the best choice in certain situations, such as backing up when facing a wall or going forward when no obstacles are detected. Rather, the behaviors have very close probabilities, each probability being close to $1/n$, where n is the number of behaviors.

What the robot did not learn was not getting into a negative reinforcement zone. The negative reinforcement zone lay approximately two centimeters from the nearest obstacle, depending on the lighting conditions. Getting out of this zone was no problem for the robot. The area before the reinforcement zone, however, presented the real problem. Because the sensors registered relatively small values even at considerably small distances to an obstacle, the similarity of exemplars created when the robot was standing at a distance of approximately three centimeters to an obstacle to exemplars created in an open area was relatively high. The moving forward behavior was stored with a positive reinforcement in exemplars in an open area, which made this behavior favorable also in situations where the robot was actually relatively close to a wall, situations where the execution of the moving forward behavior would actually cause the robot to get into the negative reinforcement zone.

In the area where the robot can actually drive without ending bumping into a wall and would bump if it drove right ahead, there is a scant difference between the sensor values. In the obstacle avoidance experiments, the sensors can be used only as skin sensors. They return very similar values for the areas where the robot can actually drive forward without hitting an obstacle and the points where the robot is a distance away from a wall which can be traversed with a forward behavior. Consequently, the robot has learned very well when to backup, which is when it has a wall right in front of it, but it did not learn not to drive against a wall when it is a few centimeters away from it. The reason for this is that when the robot is a centimeters away from an obstacle, the sensors do not have values deviating from being four or ten centimeters away from it, which makes it impossible in this scheme for the moving forward behavior to be differentiated negatively from other behaviors.

4.4 Outlook

The exemplar-based methodology used in the TINAH experiments, explained in Chapter 3, has proven in that context to be adequate for building a realistic model of the categorization performance of chickens. The model was sufficient for accounting for a first set of data from the natural subjects, i.e. chickens, and exhibited learning. In the Khepera experiments presented here, a very similar learning mechanism and methodology have been used. The robot again exhibited learning, and could discriminate cases where it had to take different actions. The concept of *categorization without categories* has been observed in the Khepera experiments, too: although the learning mechanism did not contain any mechanisms for mapping sensory values to abstract categories, the robot exhibited categorization behavior, and carried out different actions in different situations. Although the robot acted differently in the proximity of an obstacle and in an open field, there were no entities that corresponded to these two situations.

When designing a classical system, one would partition the task to be carried out into modules and these modules would then be arbitrated by a central mechanism. In the lazy learning scheme presented here, the so-called arbitrator, which coordinates the selection of a behavior and the actions to undertake once a reinforcement is received, resembles the arbitrator in a classical scheme just by name. Nevertheless, a parallel partitioning of the task space is obvious in the design of the reinforcement scheme: cases are differentiated from each other, and which case should be encouraged is explicitly coded in. The most important difference is that *how* the encouraged case should be realized is not coded in; for example, whether the robot jumps over an obstacle (although not possible with a Khepera) or just drives around it is not specified by the reinforcement scheme.

The task to be carried out by the Khepera robot and the discriminations to be learned were more complicated than in the TINAH model, but were nevertheless much more simpler than the cases in classical AI. The most important commonality in the two models was the role of the underlying principles of machinery parsimony and autonomy. The robot learned the necessary distinctions by itself, without any distinctions made by the experimenter in the input data, which took the form of readings from sensors. Both models take machinery parsimony seriously, and make a special effort to avoid unnecessary constructs to have any role in the model. The aim is to arrive at a model that builds in the least amount of machinery to arrive at a certain behavior. The only distinctions from the point of view of the observer are in the reinforcement mechanism. The reinforcement mechanism decided when the robot collided with an object or travelled a distance. As mentioned earlier, these distinctions are actually external to the model, and in the ideal case, should be embodied by another observing entity (computer or human being).

Although the similarity-based methodology has proven successful in as simple a case as obstacle avoidance, it is difficult to derive any far-reaching conclusions, especially regarding the categorization of objects and more complicated behaviors. This pertains especially to complicated actions like interacting with an object, e.g. carrying certain objects to a base, and leaving others. One of the future aims is to carry out the SMC experiments, which involve categorization of objects. On the technical side of things, the obstacle avoidance task reveals the inadequacy of the proximity sensors for categorization. The proximity sensors have a very non-uniform distribution, which means that the distance between a sensor and an obstacle is not proportional to the value measured by the sensor. What's more, the sensors do not return their full value when there is a relatively short distance between them and an obstacle, but rather when the distance is approximately two centimeters, which is far from being a physical contact. In any future categorization experiments, possible objects to be categorized by the robot could be pegs of different size, as in the SMC experiments. The proximity sensors would in this case be inadequate for the robot to learn discriminating a small and a larger peg. The reason for this is that the impression a large peg would make on the sensors would not be much different from one made by a small peg. The best solution to this is to include another sensor, a camera. The images from the camera could be handled as in the TINAH model, i.e. as raw image data, and a similarity measure at the same level could be used. A more realistic assumption would be reducing the resolution of the image and producing an integer array out of the image data, and including this array in the state information in an exemplar. This way, the effect of the proximity sensors and the image data could be balanced.

One inherent problem has to do with the fact that a decision is made in time steps that are as long as the duration of a behavior. During these time steps, no sensory values are read and no decisions are made. This is similar to the plan-as-program view presented in the first chapter: once a decision is made, a behavior is carried out without making use of or avoiding any contingencies. The behaviors in this particular case are relatively short and do not require coordination. The case would be more obvious if among the behaviors there were ones which required that the robot or one of its extensions be controlled according to some cues in the environment. In this case, if there is no monitoring of the environment while a behavior is being executed, the trap of separating the decision mechanism and the execution mechanism cannot be avoided. This important distinction, although marginally relevant here, should be kept in mind once more complicated tasks are carried out and coordination of the robotic body is required.

Chapter 5

Conclusion

In the preceding chapters, a selective overview of the history and main features of cognitive science and cognitivism has been given. The aim was to put the various kind of criticisms to this framework into perspective. It was also reported on an experiment in animal learning and an accompanying model in the framework of Comparative Cognitive Robotics. Although CCR presented a principled way of building models of learning, it was found to be too restrictive, because it relies heavily on empirical data, and to create a microworld of behaviors that are relevant only insofar as they serve to bring forth the performance that has to be replicated. A different model was presented in Chapter 4. This model used the same kind of parsimonious learning method, but it did not rely on empirical data. The aim was to create an autonomous and parsimonious learning agent. In the fifth and final chapter, some general vies on AI and robotics, and ideas about in which direction New AI has to proceed will be presented.

In the first two chapters of this thesis, the major differences between the methodologies of symbolic AI and New AI have been stressed. The main aim was to contrast the physical-symbol system hypothesis, which served as the basis of the projects of cognitive science and AI, with the interactionist approach of New AI, and the physical grounding hypothesis. The physical grounding hypothesis claimed that in order for genuine meaning to be embodied in a system, symbols should be physically grounded. Apart from this fundamental difference, there is one crucial commonality: the engineering aspect of both approaches. The aim, in the end, is, in both methodologies, to create a functioning artificial system – in most cases, an agent, also in current models in the symbolic methodology. The role of engineering rigor is immense in cognitive scientific modelling: it forces one to build working systems, at the same time promoting an understanding based on getting involved with blocks of functioning parts, putting them together in different ways and solving problems on the way to building a complete system. As revealing and promising this practice may be, there are two pitfalls which it creates and are frequently not avoided. The necessity of constructing working systems can force one to build a simplistic and unrealistic model – in the sense that totally unrealistic assumptions are made

– and nevertheless stand behind it. Another pitfall is confusing the model with the real thing, finding the explanatory power of engineering systems in a simple relationship of equality.

Studies of New AI have focused for a long time on basic level competences, but it should be clear that the final aim is no less than finding out the fundamental truths about human intelligence. This is obvious in claims by Brooks and also in studies which human intelligence is seen to be concretely based on situated and embodied particulars. The question now is, what should be done in order to proceed to human level intelligence? This question can be reformulated and divided into two questions: what is human level intelligence and where does it come from?

For cognitive science, intelligence has been the ability to manipulate symbols, and there is no intelligence without symbol manipulation. This is so even when one considers the performances chosen for modelling, rather than the tools and methods – games and microworlds have been and still are the most popular subject matters of symbolic AI. The most important properties of symbolic systems, and faculties necessary for the selected performances, are compositionality, generativity and recursiveness. Compositionality ensures that the meaning of a complex expression is a result of the structure and meaning of its constituents. Generativity and recursiveness together ensure that a production system can produce an infinite set of statements through a finite set of atomic components and rules. This is what Chomsky (1965) refers to as *infinite use of finite means*, quoting from von Humboldt.

The physical-symbol system hypothesis and cognitivism are based on these well-known properties of human language. Compositionality, generativity and recursiveness are taken to be the trademarks of human intelligence, since they are also trademarks of the most human activity we know of. One result of this dependence on language, and taking for granted its logical structure, is an inherent rationalism, pointed out by Dreyfus as explained in Chapter 1. The first thing expected of a theory of intelligence is that it explain the properties seen as inherent to any system that has symbolic logic as its groundwork.[1] One way of explaining those properties is, obviously, to put all the logic into the head, and assume that intelligent behavior is just an expression of this internalized knowledge.

An interesting discussion that illustrates the dependence on the supposedly innate properties of the language capacity is the one stemming from the argument by Fodor and Pylyshyn (1988) against a connectionist account of cognition. The burden of presenting a system or methodology that actually exhibits – or has the potential to exhibit – systematicity[2] was pressed, not surprisingly, on the neural network researchers. Neural networks were aiming, at the beginning, to develop a practice of modelling cognition at a sub-symbolic level (Smolensky,

[1] This tendency to reduce every human way of engagement to logic is, as one would expect it, not inherent to AI, and is called the "decay of dialogue" by Walter J. Ong. For a revealing account see Ong (1958).

[2] Systematicity is the idea that anyone who can think a thought T can also think systematic variants of T, where the systematic variants of T are found by permuting T's constituents.

1988). Fodor and Pylyshyn (1988) argued that connectionism, which takes pride in the distributedness of its representations and the ability of its neural nets to cope with ambiguity and noise, can not account for systematicity, and therefore is not a viable contender against cognitivist models. Since then, many neural network researchers have dedicated their time and energy to the exploration of computational methods of connectionist learning which would make it possible for neural nets to become systematic. This would amount to their being able to exhibit the three characteristics mentioned above.

Although it is obvious that systematicity is a property of language, what is not so obvious is that this is due to language reflecting the internal organization of human intelligence. While cognitivism accepts that this systematicity is *in* the intelligent being, in that it is formulated as formulas and rules in a formalization that uses mental representations as units, New AI and situated AI claim that these structures are all learned and are a result of culture. Systematicity is actually a *result* of humans being able to speak language, and not the *reason* for it. Therefore, what one has to study in order to arrive at a model of human intelligence is the evolution and the acquisition of language.

The emergence of language, or the acquisition and use of symbols, has always been a matter of dispute, and scientific methods have not always been sufficient for tackling the phenomena of symbol use and language evolution. The *Société de Linguistique de Paris* officially forbid its members discussing the birth of language, because the matter stirred nothing other than insignificant speculation. The computational modelling perspective presents us now with an opportunity just like the one computers presented to psychologists in the post-war period. Thanks to multi-agent models of the phenomena of communication, we can study proposals on the evolution of language. One can model individual competences as either embodied or simulated agents, and examine the effect of any change in these competences on the evolution of language. The work of Steels presented in Section 2.1.3 is one of the most prominent examples of such research. Indeed, computational studies on the emergence of language have recently been gaining pace.

The move from cognitivist models, with internal grammatical and semantic structures, to parsimonious agents, brings with itself an accompanying shift in the way language and a theory of language is seen. The classical structuralist and Chomskian perspectives take a synchronic view of language, where the – especially syntactic – state of a given language at a given time is the subject of study. This state is seen to be static and change is not inherent into the language. The synchronic view has to be abandoned in order to make place for an evolutionist perspective on language. Instead of defining a language as an entity with clearly definable boundaries and rules, a population definition of language has to be adapted. This involves accepting a circular causality between the individual linguistic knowledge and the public language: "[t]he individual language behaviors determine 'the' language and the language co-determines the behavior of individuals" (Steels, 1999, p.144). For a review of current work on multi-agent modelling of language evolution, see Briscoe (2002), Hurford et al. (1998) and Christiansen and Kirby (2003).

In the computational models of language evolution, language is viewed as a living system, that self-organizes and evolves through the collective dynamics of agents engaged in situated verbal interactions. In the Talking Heads experiments (Steels, 1999), situatedness, the agents' sharing a common context and the aim of the language being to refer to objects and situations in the environment, was achieved by having cameras look at the same situation. In further experiments which used the same framework of language games but further aimed to study the emergence of a common grammar, situatedness was achieved again by having the agents watch a common scene, this time an animated one. Standard machine vision algorithms were used to segment the visual scenes. The scenes involved puppets and simple objects, and aimed to illustrate typical human interactions that involve agency. A typical scene was one puppet giving a plastic cube to another, or pushing a cube in the direction of the other agent (Steels, 2004).[3] The effect of situatedness revealed itself in the choice of the participle for marking the direction of the object of an action: giving and pushing an object in the direction of a puppet had the same participle for object, and that solely because the scenes, when parsed by the agents, resembled each other (Steels, personal communication).

Robotics and the study of language evolution Although situatedness is an aspect that is often stressed in studies on language evolution, embodiment is just as often neglected. The importance of situatedness is obvious: language has to be *about* something, and it is the fundamental case when it is about something in the environment shared by the communicating agents. What embodiment provides is just as important. It allows the agents to share a dynamics, and this shared dynamics is then the fundamental context that grounds all meaning (cf. the discussion on common sense in Section 1.2.1). Naturally, for this dynamics to be shared, it first has to be formed. The formation of such a common dynamics corresponds to a history of category acquisition that precedes language. The category acquisition phase involves the interaction of each agent with a shared environment, and the autonomous acquisition up of categories. The mechanism used in TINAH and the Khepera robot is a candidate for such an autonomous learning mechanism.

Studying the evolution of language in its entirety is, of course, a huge endeavor, and carrying out a divide-and-conquer strategy is necessary. In order for earlier acquired categories to be used in the communicative process, they have to be turned into symbols, that is, entities have to be created that correspond to the autonomously acquired categories. These symbols, how they are generated and communicated, and have communities of agents arrive at a common set of symbols in order to designate a common set of categories and dynamics, is one primary part of the solution to the question of the evolution and acquisition of language (Bickerton, 2003). The process of symbol –or, more generally, sign–

[3] Although this study is in evolutionary linguistics, it does not study the evolution of a grammar. The agents already have an internal grammar, and this grammar has to be formulated in a common code among the agents for the purposes of communication. What is studied is how the agents agree on a common syntax, in the course of language games.

generation and use have already been studied by semiosis and, in the biological domain, biosemiotics. The orientation towards the study of symbol acquisition will lead to new opportunities for interdisciplinary work with these fields.

One important result of the study of symbol acquisition and use concerns the nature of the cognitivist hypothesis. The process by which the meaning of symbols are grounded in the physical embodiment and social situatedness of the agent are, from a situated cognitive scientific point of view, crucial for the processing of these symbols in the cognitive machinery, especially after thought processes based on language exist. If this is so – i.e. if syntax is not enough, and semantics does not take care of itself – then the cognitivist hypothesis is wrong, and sensitivity to syntax is not enough for intelligence. This is, just like the physical-symbol system hypothesis, an empirical question.

Studies into the grounding of symbols and evolution of language also would change the way external resources such as plans are studied. As mentioned in Section 2.1.2, plans are, like most other external resources used by humans, context-bound, and require interpretation and disambiguation. Once a proper and working model of symbol grounding and their integration into individual dynamics (in Agre's words *routines*) is achieved, plans as resources can be studied properly, in contrast to solely *plans as programs*. A plan as a resource needs interpretation, just like a sentence in natural language, and this process of interpretation is intrinsically language-bound, that is, the processes which are responsible for the interpretation of language are also responsible for the utilization of any other symbolic resource like a plan.

Bibliography

Agre, P. E. (1985). Routines. AI Memo 828, MIT Artificial Intelligence Laboratory.

Agre, P. E. (1995). The soul gained and lost: Artificial intelligence as a philosophical project. *Stanford Humanities Review*, 4(2):1–19.

Agre, P. E. (1997a). *Computation and Human Experience*. Cambridge University Press, Cambridge, MA.

Agre, P. E. (1997b). Toward a critical technical practice: Lessons learned in trying to reform AI. In Bowker, G., Gasser, L., Star, L., and Turner, B., editors, *Bridging the Great Divide: Social Science, Technical Systems, and Cooperative Work*, page 131–155. Lawrence Erlbaum Associates, Mahwah, NJ.

Agre, P. E. and Chapman, D. (1987). Pengi: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 196–201, Seattle.

Agre, P. E. and Chapman, D. (1990). What are plans for? In Maes, P., editor, *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, pages 17–34. MIT Press, Cambridge, MA.

Agre, P. E. and Shrager, J. (1990). Proceedings of the 12th annual conference of the cognitive science society. In *Proc. Annual Conf. of the Cognitive Science Society*, pages 694–701, Hillsdale, NJ. Lawrence Erlbaum.

Aha, D. W. (1998). The omnipresence of case-based reasoning in science and application. *Knowledge-Based Systems*, 11:261–273.

Anderson, M. L. (2003). Embodied cognition: A field guide. *Communications of the ACM*, 149(1):91–130.

Atkeson, C., Moore, A., and Schaal, S. (1997). Locally weighted learning. *AI Review*, 11:11–73.

Bajcsy, R. (1988). Active perception. *Proceedings of the IEEE*, 8(76):996–1005.

Ballard, D. H., Hayhoe, M. M., Pook, P. K., and Rao, R. P. (1997). Deictic codes for the embodiment of cognition. *Behavioral and Brain Sciences*, 20(4):723–742.

Beer, R. D. (1990). *Intelligence as adaptive behavior: An experiment in computational neuroethology.* Academic Press, San Diego, CA.

Beer, R. D., Quinn, R. D., Chiel, H. J., and Ritzmann, R. E. (1997). Biologically inspired approaches to robotics. *Communications of the ACM*, 40(3):31–38.

Bickerton, D. (2003). Symbol and structure: a comprehensive framework for language evolution. In Christiansen, M. and Kirby, S., editors, *Language Evolution: The States of the Art.* Oxford University Press.

Block, N. (1991). Troubles with functionalism. In Rosenthal, D., editor, *The Nature of Mental States*, pages 211–228. Oxford University Press, New York.

Boden, M. A. (1990). *The Philosophy of Artificial Intelligence.* Oxford University Press, New York.

Braitenberg, V. (1984). *Vehicles: Experiments in synthetic psychology.* MIT Press, Cambridge, MA.

Briscoe, T. (2002). *Linguistic Evolution through Language Acquisition: Formal and Computational Models.* Cambridge University Press, Cambridge, MA.

Brooks, R., Breazeal, C., Marjanovic, M., Scassellati, B., and Williamson, M. (1999). The Cog project: Building a humanoid robot. In Nehaniv, C., editor, *Computation for Metaphors, Analogy, and Agents*, Lecture Notes in Artificial Intelligence 1562, pages 52–87. Springer, New York.

Brooks, R. and Stein, L. A. (1994). Building brains for bodies. *Autonomous Robots*, 1:7–25.

Brooks, R. A. (1986). A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23.

Brooks, R. A. (1989). A robot that walks: Emergent behavior from a carefully evolved network. *Neural Computation*, 1(2):253–262.

Brooks, R. A. (1990). Elephants don't play chess. *Robotics and Autonomous Systems*, 6(1&2):3–15.

Brooks, R. A. (1991a). Intelligence without reason. In Myopoulos, J. and Reiter, R., editors, *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI-91)*, pages 569–595, Sydney, Australia. Morgan Kaufmann publishers Inc.: San Mateo, CA, USA.

Brooks, R. A. (1991b). Intelligence without representation. *Artificial Intelligence*, 47:139–159.

Chapman, D. (1991). *Vision, Instruction, and Action.* MIT Press, Cambridge, MA.

Chomsky, N. (1965). *Aspects of the Theory of Syntax.* MIT Press, Cambridge, MA.

Chrisley, R. L. and Ziemke, T. (2002). Embodiment. In Nadel, L., editor, *Encyclopedia of Cognitive Science.* Macmillan, London.

Christiansen, M. H. and Kirby, S. (2003). Language eolution: consensus and controversies. *Trends in Cognitive Sciences*, 7(7):300–307.

Clancey, W. J. (1997). *Situated cognition : on human knowledge and computer representations.* Cambridge University Press, Cambridge, U.K. ; New York, NY, USA.

Clark, A. (1997). *Being There: Putting Brain, Body and World Together Again.* MIT Press, Cambridge, MA.

Clark, P. (1990). A comparison of rule and exemplar-based learning systems. In Brazdil, P. B. and Konolige, K., editors, *Machine Learning, Meta-reasoning and Logics.* Kluwer, Boston.

Cliff, D. T. (1991). Computational neuroethology: A provisional manifesto. In Meyer, J.-A. and Wilson, S. W., editors, *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pages 29–39. MIT Press, Cambridge, MA.

Connell, J. H. (1987). Creature building with the subsumption architecture. In *International Joint Conference on Artificial Intelligence (IJCAI-87), Milan*, pages 1124–1126.

Connell, J. H. (1989). *A colony architecture for an artificial creature.* PhD thesis, MIT Electrical Engineering and Computer Science.

Cook, R. G., Riley, D. A., and Brown, M. F. (1992). Spatial and configural factors in compound stimulus processing by pigeons. *Animal Learning and Behavior*, 20:41–55.

Cox, J. K. and D'Amato, M. R. (1982). Matching to compound samples by monkeys (cebus apella): Shared attention or generalization decrement? *Journal of Experimental Psychology: Animal Behavior Processes*, 8:209–225.

Crevier, D. (1996). *The Tumultuous History of the Search for Artificial Intelligence.* Basic Books, New York, NY.

Dean, J., Kindermann, T., Schmitz, J., Schumm, M., and Cruse, H. (1999). Control of walking in the stick insect: from behavior and physiology to modelling. *Autonomous Robots*, 7:271–288.

Deiwiks, C., Gergou, A., Läer, L., Land, R., Lange, S., Plate, J., and Türkmen, U. (2003). EROSAL: Empirical robot study on animal learning. MSc cognitive science student project report, University of Osnabrück.

Domjan, M. (1998). *The Principles of Learning and Behavior.* Brooks/Cole, Pacific Grove, CA. 4th Edition.

Dreyfus, H. L. (1993). *What Computers Still Can't Do.* The MIT Press, Cambridge, MA.

Dreyfus, H. L. and Dreyfus, S. (1988). Making a mind versus modeling the brain: Artificial intelligence back at a branchpoint. *Daedalus*, 117(1):15–44.

Edwards, P. N. (1996). *The Closed World: Computers and the Politics of Discourse in Cold War America.* The MIT Press, Cambridge, MA.

Espenschied, K. E., Quinn, R. D., Beer, R. D., and Chiel, H. J. (1993). Leg coordination mechanisms in the stick insect applied to hexapod robot locomotion. *Adaptive Behavior*, 1(4):455–468.

Espenschied, K. E., Quinn, R. D., Beer, R. D., and Chiel, H. J. (1996). Biologically based distributed control and local reflexes improve rough terrain locomotion in a hexapod robot. *Robotics and Autonomous Systems*, 18:59–64.

Fikes, R. E. and Nilsson, N. J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 5:189–208.

Finney, S., Gardiol, N. H., Kaelbling, L. P., and Oates, T. (2002a). Learning with deictic representations. Technical Report AIM-2002-006, MIT AI Lab.

Finney, S., Gardiol, N. H., Kaelbling, L. P., and Oates, T. (2002b). The thing that we tried didn't work very well: Deictic representation in reinforcement learning. In *18th International Conference on Uncertainty in Artificial Intelligence (UAI)*, Edmonton, Alberta.

Fodor, J., editor (1983). *The Modularity of Mind.* MIT Press, A Bradford Book, Cambridge,MA.

Fodor, J. A. and Pylyshyn, Z. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, (28):3–71.

Gardner, H. (1985). *The Mind's New Science: A History of the Cognitive Revolution.* Basic Books, New York, NY.

Goldstine, H. (1972). *The Computer from Pascal to von Neumann.* Princeton University Press, Princeton, NJ.

Grant, D. S. and MacDonald, S. E. (1986). Matching to element and compound samples in pigeons: the role of sample coding. *Journal of Experimental Psychology*, 12:160–171.

Hahn, U. and Chater, N. (1998). Similarity and rules: distinct? exhaustive? empiric. *Cognition*, 65:197–230.

Harnad, S. (1990). The symbol grounding problem. *Physica D*, 42:335–346.

Harnad, S. (1993). Problems, problems: the frame problem as a symptom of the symbol grounding problem. *Psycoloquy*, 4(34).

Harnad, S. (2003). The symbol grounding problem. In *Encyclopedia of Cognitive Science*. Nature Publishing Group/Macmillan.

Harnish, R. M. (2002). *Minds, Brains, Computers: An Historical Introduction to the Foundations of Cognitive Science*. Blackwell Publishers, Malden, MA.

Haugeland, J. (1985). *Artificial Intelligence: The Very Idea*. The MIT Press, Cambridge, MA.

Hayes, P. J. (1987). What the frame problem is and isn't. In Pylyshyn (1987), pages 123–137.

Hurford, J. R., Studdert-Kennedy, M., and Knight, C., editors (1998). *Approaches to the Evolution of Language: Social and Cognitive Bases*. Cambridge University Press, Cambridge.

Janlert, L.-E. (1987). Modeling change – the frame problem. In Pylyshyn (1987), pages 1–40.

John, R. S. (1998). Methodologische probleme der verhaltensbasierten künstlichen intelligenz aus kognitionswissenschaftlicher perspektive. Master's thesis, Arbeitsbereich Computerlinguistik und Künstliche Intelligenz, Universität Osnabrück.

John, R. S. and Werner, C. (2004a). Comparative cognitive robotics: Using autonomous robots as empirical models of animal learning. In Schaal, S., Ijspeert, A., Billard, A., Vijayakumar, S., Hallam, J., and Meyer, J.-A., editors, *From Animals to Animats 8: Proceedings of the Eighth International Conference on the Simulation of Adaptive Behavior (SAB'04)*, pages 23–32, Cambridge, MA; London, UK. MIT Press.

John, R. S. and Werner, C. W. (2004b). How TINAH learned to peck: An implementation of the comparative cognitive robotics framework. In Schaub, H., Detje, F., and Brüggemann, U., editors, *The Logic of Artificial Life: Abstracting and Synthesizing the Principles of Living Systems. Proceedings of the 6th German Workshop on Artificial Life (GWAL)*, pages 46–55, Berlin. Akademische Verlagsgesellschaft.

K-Team (1999a). *Gripper User Manual*. K-Team SA, Lausanne.

K-Team (1999b). *Khepera User Manual*. K-Team SA, Lausanne.

Lambrinos, D., Möller, R., Labhart, T., Pfeifer, R., and Wehner, R. (1999). A mobile robot employing insect strategies for navigation. *Robotics and Autonomous Systems*, 30:39–64.

Lambrinos, D. and Scheier, C. (1995). Extended Braitenberg architectures. Technical Report 95.10, University of Zurich, Computer Science Department, AI Lab.

Lambrinos, D. and Scheier, C. (1996). Building complete autonomous agents: A case study on categorization. In *Proceedings of IROS'96, IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 170–177, Osaka, Japan.

Lashley, K. S. (1951). The problem of serial order in behavior. In Jeffress, L. A., editor, *Cerebral Mechanisms in Behavior: The Hixon Symposium*, pages 112–146, New York. Wiley.

Lehnert, W. G. (1989). Possible implications of connectionism. In Wilks, Y., editor, *Theoretical Issues in Natural Language Processing*, pages 86–90. Erlbaum, Hillsdale, NJ.

Leith, C. R. and Maki, W. S. (1977). Effects of compound configuration on stimulus selection in the pigeon. *Journal of Experimental Psychology*, 3(3):229–239.

Long, D. and Fox, M. (2003). The 3rd international planning competition: Results and analysis. *Journal of Artificial Intelligence Research*, 20:1–59.

Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., and Wilson, S., editors (1996). *From animals to animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, Cambridge, MA. MIT Press/Bradford Books.

Marr, D. (1982). *Vision: A Computational Approach.* Freeman & Co., San Francisco.

Mataric, M. J. (1989). Qualitative sonar based environment learning for mobile robots. In *SPIE Mobile Robots*, Philadelphia, PA.

McCarthy, J. and Hayes, P. J. (1969). Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence*, 4:463–502.

McCorduck, P. (1979). *Machines Who Think.* W. H. Freeman, San Francisco.

McDermott, D. (1981). Artificial intelligence meets natural stupidity. In Haugeland, J., editor, *Mind Design: Philosophy, Psychology, Artificial Intelligence*, pages 143–160. MIT Press, Cambridge, MA.

McDermott, D. V. (1987). A critique of pure reason. *Computational Intelligence*, 3:151–237.

McLaren, I. P. L., Green, R. E. A., and Mackintosh, N. J. (1994). Animal learning and the implicit/explicit distinction. In Ellis, N. C., editor, *Implicit and Explicit Learning of Languages*, pages 313–332. Academic Press, New York.

McPhail, E. (1987). The comparative psychology of intelligence. *Behavior and Brain*, 10:645–695.

Medin, D. L. and Schaffer, M. M. (1978). Context theory of classification learning. *Psychological Review*, 85(3):207–238.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63:81–97. Reprinted in Miller (1967).

Miller, G. A. (1967). *The Psychology of Communication*. Basic Books, New York, NY.

Miller, G. A., Galanter, E., and Pribram, K. (1960). *Plans and the Structure of Behavior*. Holt, Rinehart and Wilson, New York.

Minsky, M. (1967). *Computation: Finite and Infinite Machines*. Prentice Hall, Englewood Cliffs, N.J.

Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, New York.

Newell, A. (1982). The knowledge level. *Artificial Intelligence*, 18:87–127.

Newell, A. (1983). Intellectual issues in the history of artificial intelligence. In Machlup, F. and Mansfield, U., editors, *The Study of Information: Interdisciplinary Messages*. Wiley, New York.

Newell, A. and Simon, H. A. (1972). *Human Problem Solving*. Prentice Hall, Englewood Cliffs,NJ.

Newell, A. and Simon, H. A. (1976). Computer science as empirical inquiry: Symbols and search. *Communications of the ACM*, 19(3):113–126. Reprinted in Boden (1990).

Nilsson, N. J. (1984). Shakey the robot. Technical Report 323, SRI A.I. Center.

Nolfi, S. and Marocco, D. (2002). Active perception: A sensorimotor account of object categorization. In Hallam, B., Floreano, D., Hallam, J., Hayes, G., and Meyer, J.-A., editors, *From Animals to Animats 7, Proceedings of the Seventh International Conference on Simulation of Adaptive Behavior*, pages 266–271, Cambridge, MA. MIT Press.

Nolfi, S. and Parisi, D. (1999). Exploiting the power of sensory-motor coordination. In *Advances in Artificial Life, Proceedings of the Fifth European Conference on Artificial Life*, pages 173–182, Berlin. Springer Verlag.

Nosofsky, R. M. (1984). Choice, similarity, and the context theory of classification. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 14:54–65.

Ong, W. J. (1958). *Ramus: Method and the Decay of Dialogue.* Harvard University Press, Cambridge, MA.

Pearce, J. M. (1994). Similarity and discrimination: A selective review and a connectionist model. *Psychological Review*, 101:587–607.

Pfeifer, R. and Scheier, C. (1997). Sensory-motor coordination: the metaphor and beyond. *Robotics and Autonomous Systems*, 20:157–178.

Pfeifer, R. and Scheier, C. (1999). *Understanding Intelligence.* MIT Press, Cambridge, MA.

Polya, G. (1957). *How to solve it.* Princeton University Press, Princeton, NJ.

Posner, M. I. and Keele, S. W. (1968). On the genesis of abstract ideas. *Journal of Experimental Psychology*, 77:353–363.

Pylyshyn, Z. (1996). The frame problem blues: Once more, with feeling. In Ford, K. M. and Pylyshyn, Z. W., editors, *The Robot's Dilemma Revisited: The Frame Problem in Artificial Intelligence*, pages xi–xviii. Ablex Publishing, Norwood, NJ.

Pylyshyn, Z. W., editor (1987). *The Robot's Dilemma: The Frame Problem in Artificial Intelligence.* Ablex Publishing, Norwood, NJ.

Riesbeck, C. K. and Schank, R. C. (1989). *Inside Case-Based Reasoning.* Erlbaum, Hillsdale, NJ.

Riley, D. A. (1984). Do pigeons decompose stimulus compounds? In Roitblat, H. L., Bever, T. G., and Terrace, H. S., editors, *Animal cognition.* Erlbaum, Hillsdale.

Roberts, L. G. (1963). Machine perception of three-dimensional solids. Technical Report 315, MIT Lincoln Laboratory.

Roitblat, H. L. and von Fersen, L. (1992). Comparative cognition: Representations and processes in learning and memory. *Annual Review of Psychology*, 43:671–710.

Rosch, E. (1975). Cognitive reference points. *Cognitive Psychology*, 7:532–547.

Rosenblueth, A., Wiener, N., and Bigelow, J. (1943). Behavior, purpose and teleology. *Philosophy of Science*, 10:18–24.

Rumelhart, D. E. and McClelland, J. L. (1986). *Parallel Distributed Processing, Volume 1: Foundations.* MIT Press/Bradford Books, Cambridge, MA.

Russel, B. and Whitehead, A. N. (1962). *Principia Mathematica.* Cambridge University Press, Cambridge.

Russell, S. J. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach.* Prentice Hall, Upper Saddle River, New Jersey, second edition.

Sacerdoti, E. D. (1977). *A Structure for Plans and Behavior.* American Elsevier, New York.

Scheier, C. and Lambrinos, D. (1996). Categorization in a real-world agent using haptic exploration and active perception. In Maes et al. (1996).

Scheier, C. and Pfeifer, R. (1995). Classification as sensory-motor coordination: a case study on autonomous agents. In *Proceedings of the third european conference on Artificial Life ECAL95*, pages 656–667, Granada, Spain.

Schmitz, J., Dean, J., Kindermann, T., Schumm, M., and Cruse, H. (2001). A biologically inspired controller for hexapod walking: Simple solutions by exploiting physical properties. *Biological Bulletin*, 200:195–200.

Searle, J. R. (1980). Minds, brains and programs. *The Behavioral and Brain Sciences*, 3:417–457.

Shanks, D. R. (1995). *The psychology of associative learning.* Cambridge University Press, Cambridge.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423,623–656.

Shepard, R. N. (1987). Toward a universal law of generalization for physical science. *Science*, 237:1317–1323.

Simon, H. A. (1996). *Models of My Life.* The MIT Press, Cambridge, MA.

Sloman, S. A. (1996). The empirical case for two systems of reasoning. *Psychological Bulletin*, 119(1):3–22.

Smith, B. C. (1996). *On the Origin of Objects.* MIT Press, Cambridge, MA.

Smithers, T. (1995). Are autonomous agents information processing systems? In Steels and Brooks (1995), pages 123–162.

Smolensky, P. (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, (11):1–74.

Steels, L. (1991). Towards a theory of emergent functionality. In Meyer, A. and Wilson, S. W., editors, *From animals to animats*, pages 451–461, Cambridge, MA. MIT Press.

Steels, L. (1994). The artificial life roots of artificial intelligence. *Artificial Life Journal*, 1(1).

Steels, L. (1995a). The artificial life route to artificial intelligence. In Steels and Brooks (1995), pages 83–121.

Steels, L. (1995b). Building agents with autonomous behavior systems. In Steels and Brooks (1995).

Steels, L. (1996a). Emergent adaptive lexicons. In Maes et al. (1996).

Steels, L. (1996b). Perceptually grounded meaning creation. In Tokoro, M., editor, *Proceedings of the International Conference on Multi-Agent Systems*, Cambridge, MA. MIT Press.

Steels, L. (1996c). Synthesising the origins of language and meaning using co-evolution and self-organisation. In Hurford, J., editor, *Evolution of Human Language*. Edinburgh University Press, Edinburgh.

Steels, L. (1997a). Constructing and sharing perceptual distinctions. In van Someren, M. and Widmer, G., editors, *Proceedings of the European Conference on Machine Learning (ECML'97)*, pages 4–13, Berlin. Springer.

Steels, L. (1997b). A selectionist mechanism for autonomous behavior acquisition. practice and future of autonomous agents. *Robotics and Autonomous Systems*, 20:117–132.

Steels, L. (1999). The puzzle of language evolution. *Kognitionswissenschaft*, 8(4).

Steels, L. (2004). Constructivist development of grounded construction grammars. In Daelemans, W., editor, *Proceedings Annual Meeting Association for Computational Linguistics Conference*, Barcelona.

Steels, L. and Brooks, R., editors (1995). *The Artificial Life Route to Artificial Intelligence*. Lawrence Erlbaum Ass., Hillsdale, NJ.

Steels, L. and Kaplan, F. (2001). AIBO's first words : The social learning of language and meaning. *Evolution of Communication*, 4(1).

Steels, L. and Vogt, P. (1997). Grounding adaptive language games in robotic agents. In Husbands, C. and Harvey, C., editors, *Proceedings of the Fourth European Conference on Artificial Life (ECAL' 97)*, London. MIT Press.

Turing, A. (1936). On computable numbers, with an application to the Entscheidungsproblem. In *Procedings of the London Mathematical Society*, volume 42 of *2*, pages 230–265.

Turing, A. (1963). Computing machinery and intelligence. In Feigenbaum, E. A. and Feldman, J., editors, *Computers and Thought*. McGraw-Hill, New York. Original work published 1950.

van Dam, J. D. F. A., Feiner, S. K., and Hughes, J. F. (1997). *Computer Graphics: Principles and Practice*. Addison-Wesley, New York, second edition.

Varela, F. J., Thompson, E., and Rosch, E. (1993). *The Embodied Mind: Cognitive Science and Human Experience.* The MIT Press, Cambridge, MA.

von Uexküll, J. (1909). *Umwelt und Innenwelt der Tiere.* J. Springer, Berlin.

Weaver, W. and Shannon, C. E. (1949). *The Mathematical Theory of Communication.* University of Illinois Press, Urbana, IL. Republished in paperback 1963.

Webb, B. (1994). Robotic experiments in cricket phonotaxis. In Cliff, D., Husbands, P., Meyer;, J.-A., and Wilson, S. W., editors, *From Animals to Animats 3: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, pages 45–54, Cambridge, MA. MIT Press.

Webb, B. (2000). What does robotics offer animal behaviour? *Animal Behavior*, 60:545–558.

Webb, B. (2001). Can robots make good models of biological behavior? *Behavioral and Brain Sciences*, 24:1033–1050.

Webb, B. and Harrison, R. R. (2002). Integrating sensorimotor systems in a robot model of cricket behavior. In McKee, G. T. and Schenker, P. S., editors, *Sensor Fusion and Decentralised Control in Robotic Systems III*, pages 113–124, Boston, MA. SPIE.

Werner, C. W. (1999). Kategorisierungsleistungen bei Haushühnern der Rasse "Chabo". Diplomarbeit am Institut für Allgemeine Psychologie in Zusammenarbeit mit dem C. und O. Vogt Institut für Hirnforschung, Heinrich-Heine-Universität Düsseldorf.

Werner, C. W., Gravemaier, B., and Rehkämper, G. (2003). A mathematical model of configural processing of integral and separable compounds by chickens. Unpublished manuscript.

Werner, C. W. and Rehkämper, G. (1999). Discrimination of multidimensional geometrical figures by chickens: categorization and pattern-learning. *Animal Cognition*, 2:27–40.

Werner, C. W. and Rehkämper, G. (2001). Categorization of multidimensional geometrical figures by chickens (Gallus gallus f. domestica): fit of basic assumptions from exemplar, feature and prototype theory. *Animal Cognition*, 4:37–48.

Werner, C. W., Tiemann, I., Cnotka, J., and Rehkämper, G. (2004). Processing of visual compound stimuli in chickens (Gallus gallus f. domestica) is not predictable by constituting features. Submitted to *Animal Cognition*.

Whitehead, S. D. and Ballard, D. H. (1991). Learning to perceive and act by trial and error. *Machine Learning*, 7:45–83.

Wilkins, D. E. (1988). *Practical Planning: Extending the Classical AI Paradigm.* Morgan Kaufmann, San Mateo, CA.

Wittgenstein, L. (1953). *Philosophical Invetigations.* Basil Blackwell, Oxford. Translated from German by G. E. M. Anscombe.

Ziemke, T. (1999). Rethinking grounding. In Riegler, A., Peschl, M., and von Stein, A., editors, *Understanding Representation in the Cognitive Sciences.* Kluwer Academic / Plenum Publishers, New York.

Ziemke, T. (2001). Disentangling notions of embodiment. *Workshop on Developmental Embodied Cognition.*

Ziemke, T. and Sharkey, N. E. (2001). A stroll through the worlds of robots and animals: Applying Jakob von Uexküll's theory of meaning to adaptive robots and artificial life. *Semiotica*, 134(1-4):701–746.

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen oder Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

Ulaş Türkmen
Osnabrück, den 17.01.2005