Symbols, patterns, and behavior: towards a new understanding of intelligence

Rolf Pfeifer
AI Lab, Computer Science Department
University of Zurich, Winterthurerstrasse 190
CH-8057, Zurich, Switzerland
phone: +41-1-257 43 20/31; fax: +41-1-363 00 35
pfeifer@ifi.unizh.ch

Abstract

Artificial Intelligence (AI) has two main goals, namely to understand intelligence and to develop systems which behave in intelligent ways. The classical approach views an intelligent agent as a symbol processor: it receives input from the environment, processes the information, i.e. manipulates symbols, and produces some output. This way of looking at intelligence has also been called the information processing approach. Recently symbol processing models have been criticized in a number of ways. It has been argued that they lack robustness, that they cannot perform in real time, that learning is mostly ad hoc and not performed in a principled way, and that due to their discrete and sequential nature they are more like digital computers rather than like brains. Moreover, it has been proposed that classical AI models suffer from a number of fundamental problems, ("symbol grounding", "frame problem", lack of "situatedness"). In this paper these problems will be reviewed and illustrated. It will be discussed to what extent connectionist models solve the problems of classical AI. It will be argued that, although they do solve some of them, they are not sufficient to answer the fundamental problems. In order to achieve the latter, it is necessary to study embodied autonomous agents which interact on their own with their environments. If we want to build interesting agents we have to observe a number of design principles. These principles will be outlined. They will be used to contrast the new view of intelligence with the traditional one.

1 Introduction

Artificial Intelligence (AI) has two main goals, namely to understand intelligence (or intelligent behavior) and to develop systems which behave in intelligent ways. In this paper the focus is on understanding intelligence, i.e. we adopt a cognitive science perspective. However, we deal with both issues. What makes the methodology of AI so productive is the synthetic character: if we are to design intelligent systems we have to understand behavior, and actually building systems helps us understand behavior in new ways. The cognitive science perspective implies that our main goal is to learn about intelligence. This goal also provides an evaluation criterion. Thus, we might prefer a program for playing chess over another one, even though its performance is worse. The program that plays worse might instantiate some psychological principles of perception and human memory that we consider important, whereas the winning program might simply be based on search.

Before we go on, we have to say what we mean by intelligence. It would be hopeless to try and define it—we are not very likely to achieve agreement. It is highly subjective and strongly depends on our expectations. If I, as an adult, play chess, nobody is very impressed. However, if a two year old played exactly like me, we would be very impressed, even though I am only a very average player. Rather than pursuing the question what intelligence *is*, we propose to replace the question by a different, more productive one: Given some behavior that we find interesting, how does it come about? What are the underlying mechanisms? If we pursue this line, we no longer have to argue whether ants are intelligent or not. We either find their behavior interesting and then it is worthwhile trying to work out the mechanisms, or we don't—and then we might not be interested in the mechanisms either.

Roughly the field of AI can be divided into three main approaches or paradigms, symbol processing AI, connectionism, and "New AI". This categorization also corresponds to a historical development with paradigm shifts in between. Symbol processing AI is based on the idea that intelligence can be viewed as the manipulation of abstract symbols. Connectionism, also called "neural networks", or "parallel distributed processing", refers to a particular type of modeling approach that is vaguely inspired by brain-like systems. "New AI"—or behavior-based AI—studies systems which are physically embodied and which have to interact on their own with the real world.

In the early days of AI—during the symbol processing period—the main interest was in thinking, reasoning, problem solving, language, i.e. in "high-level" human capabilities. Over time, the interest has shifted towards more simple or "low-level" kinds of behavior that relate more to sensory-motor capacities like perception and object manipulation. Connectionism has been strongly focusing on this area. More recently there has been yet another shift of interest, namely from systems that excel at one particular task to systems capable of performing many different tasks like navigating in an unpredictable world, learning categorizations in a real-world environment,

collecting and manipulating objects, while maintaining battery level and physical integrity, etc. From these developments a new understanding of intelligence has emerged.

Ultimately, we are interested in developing a "theory of intelligence". Given the state-of-the-art it is entirely open what this theory will look like. In classical AI it was suggested that the theory be based on the idea of symbol processing (e.g. Newell, 1990). Connectionists believe that it will be based on parallel distributed processing of patterns of activation (e.g. Rumelhart and McClelland, 1986). Recently, it has been suggested that the mathematical theory of non-linear dynamics might provide the right framework because it is capable of capturing not only aspects of the control architecture, but of the complete system, including its physics (e.g. Beer, in press; Steinhage and Schoener, in press). Another group of researchers is capitalizing on evolutionary considerations (see, e.g. Harvey et al., in press, for a review). Principles of micro-economics have also been suggested as a framework to understand intelligent behavior (e.g. McFarland and Boesser, 1993). The field is still changing rapidly and no clear winner can be foreseen. Therefore, we have chosen to capture some of the insights gained in recent years in the form of a set of compact *design principles*, rather than in the framework of a rigid formal theory.

We will proceed as follows. First we will outline the classical approach. This will be very brief, assuming that everyone is familiar with it. Then we will point out some of the problems that eventually lead to a paradigm shift. We then discuss in what ways connectionism contributes to the resolution of these problems, using a number of examples. We then present some of the design principles for intelligent systems and discuss to what extent this new view resolves some of the basic issues.

But before starting we need to make a short digression. We have found that in the literature on AI and cognitive science there is a lot of confusion about the so-called "frame-of-reference" problem. We will give a short outline using the famous example of Simon's ant on the beach.

2 The "frame-of-reference": Simon's ant on the beach

Figure 1 shows an ant walking on the beach. The example has been taken from Herbert Simon's seminal book entitled "The Sciences of the Artificial" (Simon, 1969). Let us assume that the ant is coming from the upper right corner of the picture and walking towards the lower left one, where its nest is located. The path of the ant has been marked by a line, its trajectory. The trajectory is highly complicated because the beach is full of pebbles, rocks, and other obstacles. However, this complexity is in the eye of the observer, rather than in the ant itself. Surely, the trajectory is not stored in the ant's head and so its behavior cannot be based on it. In other words, there is no trajectory functioning, say, like a plan. Rather, the mechanisms that are driving the ant's behavior may be very simple. They might be described as rules like "if obstacle sensor on left is activated, turn right" (and vice versa). These "rules" are implemented in terms of neural structures within the ant. The neural structures are embedded in the body of the ant. The interaction of the neural substrate with the environment is mediated through the body. In this interaction with the environment the apparent complexity of the trajectory emerges. We say apparent complexity, because the complexity is in the eye of the observer, rather than being a property of the agent itself.

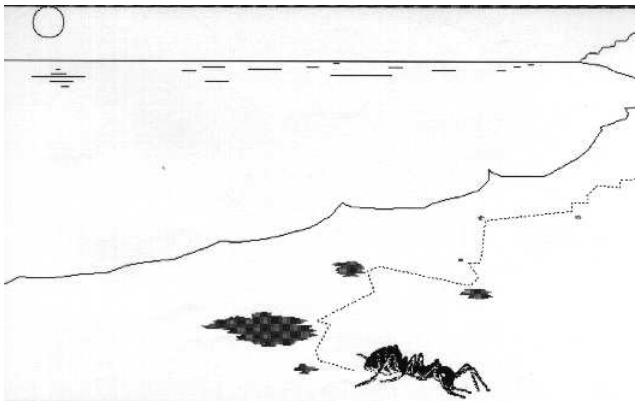

Figure 1: Simon's ant on the beach.

The trajectory we observe, refers to behavior. Behavior is always an *interaction of an agent with its environment*. It is neither a property of the agent itself, nor a property of the environment alone. The behavior is to be clearly distinguished from the internal (neural) mechanism that is responsible for it. In other words, behavior cannot be reduced to internal mechanism. Doing so would constitute a *category error*.

This seems almost trivially obvious. But if it is so evident, it is even more surprising that there is an enormous confusion about this problem in the entire literature. Throughout this paper we will refer to the "frame-of-reference" problem (for a detailed discussion, see Clancey, 1991).

3 Symbols: The traditional AI approach

3.1 Characterizing symbol processing

Assuming that everyone is familiar with symbol processing AI (sometimes called "classical AI" or "traditional AI"), our introduction will be very short. Symbol processing AI is based on the idea that intelligence can be viewed as the manipulation of abstract symbols. Newell and Simon (1976) proposed the so-called "Physical Symbol Systems Hypothesis" which, in essence, states that a necessary and sufficient condition for general intelligent action is that it be a physical symbol system. The term "physical" refers to the idea that symbol systems must be realized in some physical medium (paper, computer, brain) but it is irrelevant *how* they are realized. Typical examples of artificial physical symbol systems are production systems (or rule-based systems) or general

purpose programming languages like Lisp or C. Necessary means that any system lacking this property cannot be intelligent, sufficient implies that a system having this property has the potential for intelligent action. A (symbolic) representation (figure 2) in the sense of Newell refers to a situation in the outside world and obeys the "law of representation", namely:

$$\text{decode}[\text{encode}(T)(\text{encode}(X))] = T(X),$$

where X is the original external situation and T is the external transformation (Newell, 1990, p. 59). There is an encoding as well as a decoding function for establishing a mapping between the outside world and the internal representation.

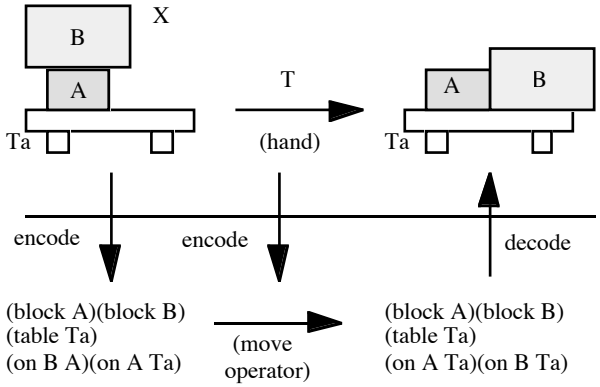

Figure 2: The "law of representation" (following Newell, 1990).

The symbol processing approach views an intelligent agent as an information processor which receives input from the environment, processes the information, i.e. manipulates symbols, and produces some output. Therefore it has also been called the *information processing approach*. Over the years, it became clear that this approach is suffering from a number of problems, which in turn lead to a paradigm shift. Let us look at some of them.

3.2 Problems of traditional symbol processing models

It has been argued that symbol processing models lack robustness (i.e. fault and noise tolerance, as well as the capacity to generalize), that they cannot perform in real time, that learning is mostly ad hoc and not performed in a principled way, and that due to their discrete and sequential nature they are more like digital computers rather than like brains. Moreover, it has been argued that classical AI models suffer from the fundamental problem of "symbol grounding" and the "frame problem", and that they lack the property of "situatedness". In the paper these problems will be reviewed and illustrated.

The well-known problems

Robustness: One symptom traditional AI symbol processing systems suffer from is lack of robustness which means that they lack noise and fault-tolerance, and that they cannot behave appropriately in new situations. Standard symbol processing models are neither noise nor fault tolerant *unless* there is explicit provision for noise and particular types of faults. The most important point concerning robustness, however is the inability to perform appropriately in novel situations, i.e. the lack of generalization capacity. If a situation arises which has not been predefined a traditional symbol processing model will break down[1].

Performance in real time: It has turned out that when systems which are based on symbol processing models are embedded in real robots they are typically too slow, i.e. they are not capable to meet real time processing demands. The reasons for this will be discussed in detail below.

Integrated learning: Traditional AI models have often been criticized because their learning mechanisms are ad hoc and imposed on top of non-learning systems. In contrast, the brain is a system which continuously learns. Humans, for example, learn always whether they like it or not. If you (the reader) read this paper you will learn something whether you like it or not, or whether you find it is useful or not. There are exceptions of classical systems where learning is an integral part of the architecture and takes place continuously like SOAR (Laird et al., 1987) but they are not representative of the majority of approaches. Moreover, SOAR, like other classical models, suffers from the fundamental problems of symbol processing systems (see below).

Sequential nature of programs: One main point of criticism has been that the architecture of today's AI programs is sequential and they work on a step-by-step basis. By contrast the human brain is massively parallel with activity in many parts of the brain at all times. Moreover, it is hard to imagine that something like symbols would be "found" in the brain. This problem is induced by the fact that current computer technology is largely based on architectures of the von Neumann type which is, at the information processing level, a sequential machine. The notion of computation abstracts from the physical realization and only considers the algorithmic level. In cognitive science the cognitivistic position makes a similar kind of abstraction: intelligent function or cognition can be studied at the level of algorithms, the physical realization of the algorithm does not matter (Putnam, 1975). We will argue later on that the physical realization indeed does matter. But we have to extend our perspective to the agent as a whole.

The criticisms of AI models presented so far are well-known. Since the mid-eighties a number of additional ones have been raised, pertaining to fundamental issues. It has been argued that traditional symbol-processing AI models suffer from the "frame problem" and the problem of "symbol grounding", and that they lack the property of "situatedness". These problems will now be reviewed in turn.

[1]There are some approaches in symbol processing AI (in the field of machine learning) which in fact do generalize to some extent but the ways in which generalization is achieved is typically ad hoc. See also below: "Integrated learning".

The fundamental problems

Traditionally AI models have been conceived primarily for artificial, virtual or formal worlds. Examples are search, formal games like checkers or chess, and theorem provers[2]. Whenever dealing with the *real* world two important aspects must be taken into account: (i) models must somehow relate to the outside world (otherwise there would be no point in building them), and (ii) the real word, in contrast to a virtual one, is constantly changing, intrinsically unpredictable and only partially knowable. The import of this real-world perspective can hardly be overestimated. It is at the heart of the fundamental problems.

The symbol grounding problem: The symbol grounding problem relates to aspect (i). It refers to the problem of how symbols acquire meaning. In AI the meaning of symbols is typically defined in a purely syntactic way by how they relate to other symbols and how they are processed by some interpreter (Newell and Simon, 1976; Quillian, 1968). The relation of the symbols to the outside world is rarely discussed explicitly. This position not only pertains to AI but to computer science in general. Except in real-time applications the relation of symbols to the outside world is never discussed. The—typically implicit—assumption made is that the potential users will know what the symbols mean (e.g. the price of a product stored in a data base). Interestingly enough this idea is also predominant in linguistics: it is taken for granted that there is some kind of correspondence between the symbols or sentences and the outside world. The study of meaning then relates to the translation of sentences into some kind of logic-based representation whose semantics is clearly defined (Winograd and Flores, 1986, p. 18). This position is acceptable as long as there is a human interpreter and it can be safely expected that he is capable of establishing the appropriate relations to some outside world: the mapping is "grounded" in the human's experience of his or her interaction with the real world.

However, once we remove the human interpreter from the loop, as in the case of autonomous agents, we have to take into account that the system needs to interact with the environment on its own. Thus, if there are symbols in the system, their meaning must be *grounded* in the system's own experience in the interaction with the real world. Symbol systems in which symbols only refer to other symbols are not grounded because the connection to the outside world is missing. The symbols only have meaning to a designer or a user, not to the system itself. It is interesting to note that for a long time the symbol grounding problem has not attracted much attention in AI or cognitive science—and it has never been an issue in computer science in general. Only with the renewed interest in autonomous robots it has come to the fore. This problem has been discussed in detail by Harnad (1990). It will be argued later that the symbol grounding problem is really an artifact of symbolic systems and "disappears" if a different approach is used.

[2]The field of robotics is an exception.

Situatedness: The concept of "situatedness" (or "situated cognition", "situated action", "situated agents"), has recently attracted a lot of interest and lead to heated debates about the nature of intelligence and the place of symbol processing systems in studying intelligence. For example, there is a complete issue of the journal Cognitive Science dedicated to "situatedness" (Cognitive Science, 1993). "Situatedness" roughly means the following. First, it implies that the world is viewed entirely from the perspective of the agent (*not* from the observer's perspective). Second, a situated agent capitalizes on the system-environment interaction. It's behavior is largely based on the current situation rather than detailed plans. It only focuses on the relevant aspects of the situation. And third, a situated agent is not merely reactive, but brings its own experience to bear on the current situation. In other words, the behavior of a situated agent will change over time. Because of these properties, situated agents can act in real time.

The perspective of "situatedness" contrasts with classical AI where the approach has been—and still is—to equip the agents with models of their environment. These models form the basis for planning processes which in turn are used for deciding on a particular action. In this view the agent perceives a situation ("sensing"), recognizes objects, draws a number of inferences about the current situation and about the potential effects of various actions, forms a plan ("thinking"), decides on a particular action and finally performs the action ("act"). This is called a "sense-think-act" cycle. This may work in a virtual world with clearly defined situations and given operators. But even there, plan-based agents run quickly into combinatorial problems (e.g. Chapman, 1987). Moreover, since the environment is only partially knowable a complete model cannot be built in the first place. Even if only partial models are developed, keeping the models up to date requires a lot of computational resources. Inspection of the problem of taking action in the real world shows that it is neither necessary nor desirable to develop "complete" and very detailed plans and models (e.g. Winograd and Flores, 1986; Suchman, 1987). Typically only a small part of an agent's environment is relevant for its action. In addition, instead of performing extensive inference operations on internal models or representations the agent can interact with the current situation: the real world is, in a sense, part of the "knowledge" the agent needs in order to act, it can merely "look at it" through the sensors.

Traditional AI systems, and most computer systems for that matter, are not situated and there is no reason why they should be because there is always a human interpreter in the loop. However, if we are interested in building systems which act directly in the real world they will have to be situated. Otherwise, given the properties of the real world, the system will not be able to perform intelligently.

The "sense-think-act" view also suggests an information processing perspective: the input is given by the sensing, the sensory information is processed, and an

output is delivered in the form of an action. This view, while appropriate for traditional computer applications, turns out to be highly inappropriate for situated agents.

The "frame problem": The "frame problem" was originally pointed out by McCarthy and Hayes (1969). It has more recently generated a lot of interest (e.g. Pylyshyn, 1987). It comes in several variations and there is not one single interpretation. The central point concerns how to model change (Janlert, 1987): given a model of a continuously changing environment, how can the model be kept in tune with the real world? Assuming that the model consists of a set of logical propositions (which essentially holds for any representation) any proposition can change at any point in time. However, the physical world is inherently constrained by the laws of physics: objects do not simply disappear, they do not start to fly without reason, etc. But ice cubes lying in the sun do disappear. Such constraints either have to be modeled explicitly or certain heuristics have to be applied. One heuristic is that we assume things do not change unless explicitly modeled. But if this latter strategy is adopted, how about a cup on a saucer when the saucer is moved? The cup will also change its position. The problem is, that there is potentially a very large number of possible inferences which can be drawn. Let us explain this using an example by Dennett (1987).

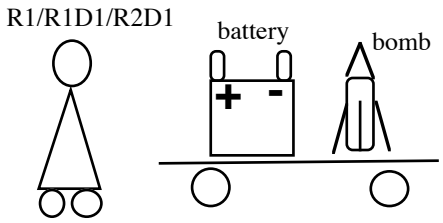

Figure 3: Illustration of the "frame problem" (following Dennett, 1987).

The robot R1 has been told that its battery is in a room with a bomb and that it must move the battery out of the room before the bomb goes off (figure 3). Both the battery and the bomb are on a wagon. R1 knows that the action of pulling the wagon out of the room will remove the battery from the room. It does so and as it is outside, the bomb goes off. Poor R1 had not realized that pulling the wagon would bring the bomb out along with the battery.

The designers realized that the robot would have to be made to recognize not just the intended implications of its acts, but also its side-effects by deducing these implications from the descriptions it uses in formulating its plans. They called their next model the robot deducer, or short R1D1 and did the same experiment. R1D1 started considering the implications of pulling the wagon out of the room. It had just finished deducing that pulling the wagon out of the room would not change the color of the room's walls when the bomb went off.

The problem was obvious. The robot must be taught the difference between relevant and irrelevant implications. R2D1, the robot-relevant-deducer, was again tested. The designers saw R2D1 sitting outside the room containing the ticking bomb. "Do something!" they yelled at it. "I am", it retorted. "I am busily ignoring some thousands of implications I have determined to be irrelevant. Just as soon as I find an irrelevant implication, I put it on the list of those I must ignore, and …" the bomb went off. (Dennett, 1987, pp. 147-148).

We have now listed the most important problems of traditional AI models. Our considerations are not restricted to AI but apply to computer systems in general. We will now discuss two solutions which have been proposed to resolve some of these issues, namely connectionism and "New AI".

4 Patterns: The contribution of connectionism

Because traditional AI was not progressing satisfactorily any more, connectionism was highly welcomed by large parts of the research community in AI. The hope was that connectionism would resolve many of the problems of traditional symbol processing AI. Indeed connectionism does contribute in interesting ways.

For example, connectionist models are fault tolerant and noise tolerant. Because of their parallel nature they preserve most of their functionality if there is noise in the data or if certain parts of the network malfunction. But more important, connectionist models have a certain ability for generalization. They are capable of behaving appropriately even in circumstances the model has not encountered before. These are the essential factors contributing to the robustness of connectionist models.

There are additional characteristics which have contributed to their popularity. Learning is intrinsic and they are in some sense more brain-like. Connectionist models consist of large numbers of nodes and connections. Typically there are too many connections to adjust manually, so they have to be tuned through learning mechanisms. Connectionist models have attracted a lot of attention since they do not only perform what has been programmed into them, but also what they have learned. In this sense they sometimes show unexpected or "emergent" behavior. In contrast to symbolic models connectionist ones integrate learning in natural ways. Examples of neural network learning will be given below.

Another aspect which has contributed to the popularity of connectionst models is more of a psychological nature. Connectionist models have been praised for being more brain-like than traditional ones (e.g. Rumelhart and McClelland, 1986). There are many units working in parallel and they process patterns rather than symbols. However, although they do draw inspiration from reflections about the brain, they reflect brain properties only in a very remote way, if at all. If we want to model real brain function, our models must look very different (e.g. Reeke and Edelman, 1989). We will not go into this aspect any further since it is highly controversial and it is somewhat marginal for the argument to be made in this article.

In summary, connectionist models are more robust, they integrate learning, and they are somewhat more "brain-like" than classical models. Because of their parallel nature, they may also cope better with real-time demands (but this latter point is debatable).

Let us now discuss to what extent connectionism contributes to resolving the fundamental issues. Connectionist models process *patterns of activation* rather than symbols. This seems a more realistic view of what is going on in the brain than the one endorsed by symbol manipulating models. Moreover, since connectionist models can learn they could potentially learn to make their own categorization of the environment, rather than having it programmed into the system by the designer. One might think that this would provide a solution to the symbol grounding problem. We will see that this is not automatically the case.

4.1 Supervised learning

Let us illustrate our argument with NETTalk, a well-known classical connectionist model (Sejnowski and Rosenberg, 1987). NETTalk translates English text into speech using a multi-layer feed-forward backpropagation network. The architecture is illustrated in figure 4. There is an input layer, a hidden layer, and an output layer. At the input layer the text is presented. There is a window of seven slots. This window is needed since the pronunciation of a letter depends strongly on the context in which it occurs. In each slot one letter is encoded. For each letter of the alphabet there is one node in each slot. Input nodes are binary on/off nodes. Therefore, an input pattern consists of seven active nodes (all others are off). The nodes in the hidden layer have continuous activation levels. The output nodes are similar to the nodes in the hidden layer. They encode the phonemes by means of a set of phoneme features. This encoding of the phonemes in terms of phoneme features can be fed into a speech generator. For each letter presented at the center of the input window—"e" in the example of figure 4—the correct phoneme encoding is known. By "correct" we mean the one which has been encoded by linguists earlier[3]. The model starts with random connection weights. It propagates each input pattern to the output layer, compares the pattern in the output layer with the correct one and adjusts the weights according to a learning algorithm, namely backpropagation. After presentation of many (tens of thousands) patterns the weights converge, i.e. the network picks up the correct pronunciation.

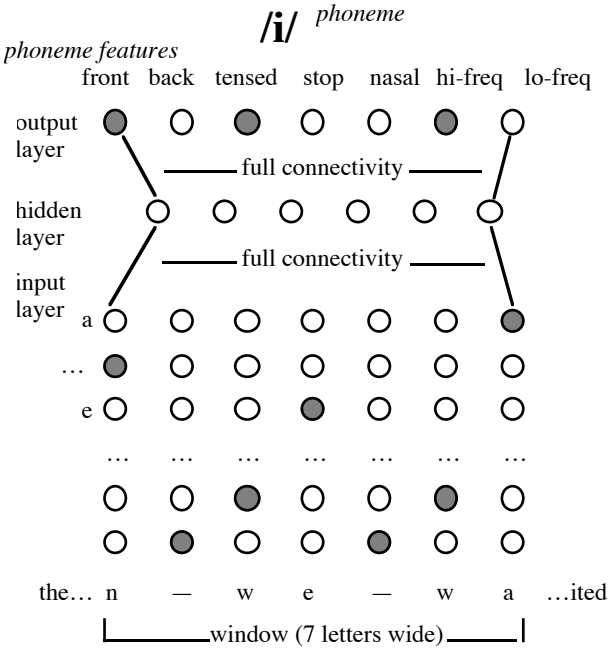

Figure 4: Architecture of the NETTalk model. Details, see text.

As demonstrated by the authors, NETTalk does indeed solve the well-known problems. "Shaking" the weights, i.e. superimposing random distortions on the weights, removing certain connections in the architecture, and errors in the encodings do not significantly influence the network's behavior. Moreover, it can handle—pronounce correctly—words it has not encountered before, i.e. it can generalize. In short, the model is *robust*. Learning is an intrinsic property of the model. Moreover, and that is one of the most exciting properties of the model, at the hidden layer certain nodes start distinguishing between vowels and consonants. In other words they are on when there is a vowel at the input, otherwise they are off. This consonant-vowel distinction has also been called "emergent".

Let us now examine the fundamental problems. Each input node corresponds to a letter. Letters are symbols, i.e. the encoding at the input layer is in terms of symbolic categories. Phoneme features are designer defined categories and thus the respective sound encodings are also symbolic. There are two points to be made. First, the system is not coupled to the environment. The interpretation of input and output is entirely up to humans who have to interpret the symbols at the input and the output. The fact that the output is fed into a speech generator is irrelevant since this has no effect on the model. Therefore, NETTalk, just like any traditional model in AI, suffers from the symbol grounding problem. It can therefore be expected that even if NETTalk is improved it will never reach a human-like performance level. It should be mentioned that the authors never claimed to be solving the symbol grounding problem with this model.

[3]In one experiment a tape recording from a child was transcribed into English text and for each letter the phoneme encoding as pronounced by the child was worked out by the linguists. In a different experiment the prescribed pronunciation was taken from a dictionary.

Second, the consonant-vowel distinction is not really emergent, but—in a sense—precoded. It can be shown (Verschure, 1992) that of those features which are used to encode vowels, only about 5% are also used to encode consonants and vice versa. In other words, the distinction is not really acquired by the system but rather (indirectly) pre-programmed by the way the examples are encoded in a symbolic way. Again, this distinction is not grounded in the model's experience but implicitly grounded in the experience of the designer of the model. Since there is no interaction with the real world (only patterns are presented to the network input layer) the frame problem and situatedness are not addressed.

From this discussion it can be concluded that supervised learning does solve a certain class of problems. But it solves the problems at an "information processing" level, rather than by interaction with the real world. Supervised learning does not resolve the issue of symbol grounding and will not lead to situated systems. The categories the model has at its disposal are given at design time once and for all. All the models can do is combine the basic categories in various ways. But the basic categories that determine how the model can interact with its environment, are fixed.

While many would probably agree that supervised models are based on designer-defined categories, there is likely to be disagreement about unsupervised models.

4.2 Unsupervised schemes

A prominent example of an unsupervised scheme is Kohonen's topological map (e.g. Kohonen, 1988a) which comes in many variations. Again, assuming familiarity, the presentation is very brief. The basic architecture is shown in figure 5. The input layer is fully connected to the map layer. In the map layer, there are lateral connections which are excitatory for close neighbors, inhibitory for those further away, and neutral for the ones yet further out. Patterns are presented to the model at the input layer and depending on the particular architecture and choice of parameters it will eventually learn a particular categorization of the input space. The details of the algorithm do not matter. What does matter is the basic principle that there is no need for the system to be given a classification of input patterns by the designer (which is why this is called unsupervised).

For example, in the "neural phonetic typewriter" (Kohonen, 1988b) the inputs are spectral patterns corresponding to pre-processed signals and the classes which are formed can be interpreted as "pseudo phonemes". Pseudo phonemes are like phonemes but they have a shorter duration (10ms rather than 40 to 400ms). In contrast to supervised learning, the designer does not predefine the classes into which the patterns need to be sorted. But does the model really acquire its own categorization in its interaction with the real world, i.e. does it really solve the symbol grounding problem? The answer is "no". Why?

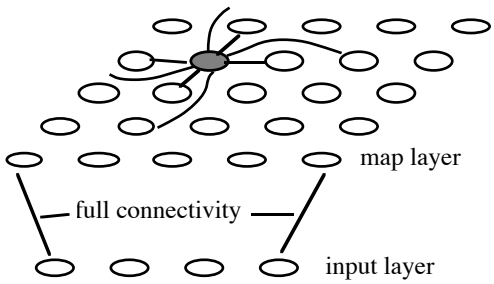

Figure 5: Basic architecture of a Kohonen network. Details, see text.

The patterns which are presented to the model have been carefully preselected by the designer. This does not imply that the designer determines the individual patterns to be presented, but he determines the types of patterns that the system should be able to process. In other words, the designer, just as in the case of a supervised model, makes a preclassification of the world in terms of what is meaningful to the system. In the case of the "phonetic typewriter" speech samples were selected to cover the space of possible phonemes, and then they were appropriately pre-processed. Clearly, non-supervised learning is an important step in the right direction because, within the preselected set of patterns, the system finds its own categories.

4.3 An attractive formalism

Because of their desirable properties (robustness, learning and generalization capacity), connectionist models or neural networks, are excellent candidates for modeling the interaction of systems with the real world. In fact, they have been widely used for signal processing, pattern recognition, and motor control. The purpose of our critical analysis was to demonstrate that connectionism per so does not automatically resolve the fundamental problems underlying the design of intelligent systems. For example, it does not explain why an agent categorizes the world in the first place, how it focuses on certain aspects of the sensory stimulation and not on others, how it chooses its behaviors depending on the situation, etc. To answer these questions, more is required. And this is what the next section is about.

5 Understanding behavior: design principles of autonomous agents

5.1 Conceptualizing intelligent agents

As pointed out initially, in symbol processing AI high-level capacities like logic, abstract problem solving, human natural language, theorem proving, reasoning and formal games were considered to be the hallmark of intelligence. There was an implicit underlying belief that once we understand the high-level processes we merely add sensors and effectors and we have a system capable of interacting with the real world. Unfortunately, this turned out not to be the case. Doubts have been raised whether this approach of focusing on high-level processes and

adding sensors and effectors later on, might not be fundamentally flawed (e.g. Brooks, 1991). Brooks suggested that if we are to understand behavior we must study physically embodied real world agents. This suggestion has lead to a research area which is rapidly growing, namely "New AI".

Researchers in AI realized that what was considered hard initially, turned out to be easy and those things which were viewed as being easy add-ons later like perceptual and motor capabilities, turned out to be hard. Connectionism is an interesting development in this respect since it started focusing on more "low-level" processes such as pattern recognition. But the interest has not only shifted to perceptual-motor skills, but to complete agents. There are a number of reasons for this.

Let us look at a system behaving in the real world, e.g. a mobile robot which has the task to collect uranium ore in an unknown environment. In order to do so it has, among many other things, to avoid obstacles and recognize uranium ore. As it is moving its sensors receive continuously changing physical stimulation and this stimulation is largely determined by what the agent currently does. And what the agent currently does in turn determines, together with the sensory stimulation and the internal state, what it will do next. There is nobody to tell the agent what the relevant patterns of sensory activation are. Unlike supervised and non-supervised neural networks, there is no neat set of training patterns: the agent has to decide from its very own, situated perspective. This constitutes an entirely different set of problems. The design principles have been devised in order to conceptualize agents behaving in a real, physical world.

5.2 Design principles of autonomous agents

Types of explanations

Remember that our ultimate goal is to understand principles of intelligence. There is a kind of "meta principle" that has to be endorsed if the design principles are to make sense. It states that agents always be evaluated from three different perspectives, namely functional, learning and development, and evolutionary. Experience has shown that these three perspectives contribute in complementary ways to our understanding.

The *functional* perspective[4] explains why a particular behavior is displayed by an agent based on its current internal and sensory state. Often, this kind of explanation is used in engineering. But also in cognitive science it is highly productive. Just remember Simon's ant on the beach, where it is surprising how seemingly complex kinds of behavior result from very simple mechanisms. The *learning and developmental* perspectives not only resort to internal state, but to some events in the past in order to explain the current behavior. They provide an explanation of how the actual behavior came about. The distinction between learning and development is that development includes maturation of the organism, whereas learning is more general and does not necessarily include change of the organism. *Evolutionary* explanations provide reasons why a particular capacity of an agent is there in the first place, e.g. why it might be beneficial to have a vision system. All these types of explanations can be applied to individuals, but also to groups or whole societies of individuals.

Classical symbol processing models have mostly provided explanations at the functional level. The problem with classical models was, that often no clear distinction was made between internal mechanisms and behavior, because there simply was no behaving organism. Machine learning, in particular connectionist models, have adopted a learning perspective. But the explanations were often relatively uninteresting because the training patterns were prepared by the designer. The evolutionary perspective is a more recent development in AI.

Classes of principles

There are three classes of design principles. An overview is given in Table 1. The first class concerns the kinds of agents and behaviors that are of interest from a cognitive science perspective. We stress the cognitive science perspective since from an engineering perspective, other types of agents are typically of greater interest (at least at the moment). The second class concerns the agent itself, its morphology, its sensors and effectors, its control architecture, and its internal mechanisms. The third class contains principles that have to do with ways of thinking and proceeding, with stances, attitudes, and strategies to be adopted in the design process. Because of space limitations we will only illustrate some of them with a few case studies (for more detail, see Pfeifer, 1996b).

Table 1: Summary of design principles

Principle	Name
Types of agents of interest, ecological niche and tasks	
1	The "complete agents" principle
2	The "ecological niche" principle
Morphology, architecture, mechanism	
3	The principle of parallel, loosely coupled processes (the "anti - homunculus" principle)
4	The "value" principle
5	The principle of sensory-motor coordination
6	The principle of "ecological balance"
7	The principle of "cheap designs"
Strategies, heuristics, stances, metaphors	
8	"Frane-of-reference" principle
9	"Constraints" principles
10	Compliance with principles
	etc.

[4]The term "functional" is used in different ways. Here the term is used to distinguish one level of explanation from a learning/developmental and an evolutionary one.

Type of agents, ecological niche, and tasks

Initially we argued that it may not be a good idea to define what we mean by intelligence and that it is perhaps better to define the kinds of agents and behaviors that we are interested in and then look for the mechanisms. This class of principles tries to characterize the kinds of agents that are most worthwhile being investigated. As pointed out before, in classical AI the tasks of interest pertain to high-level thinking. In connectionism they were more in the areas requiring pattern processing, i.e. sensory-motor based. We saw that the study of these tasks alone, still does not suffice to understand intelligence. Thus, this class of principles states that the agents of interest are autonomous (i.e. independent of external control), self-sufficient (i.e. can sustain themselves over extended periods of time), embodied (i.e. they are realized as a physical system), and situated (i.e. the interaction with the environment must be controlled by the agent itself). Moreover, the agent must be able to bring in its own experience in dealing with the current situation. This illustrates the shift of interest in research topics mentioned at the beginning. The basic idea of this set of principles is that if we are to make progress in the study of intelligence, it is these kinds of systems that we must study.

Keep in mind that these design principles only make sense if your primary interest is in cognitive science, i.e. in understanding the basic principles of intelligence. If the main goal is engineering, different principles have to be applied.

5.3 Functional explanations

The example that we will discuss illustrates functional explanations and a principle from class three, namely the "frame-of-reference" principle. Earlier, we discussed the "frame-of-reference" problem using Simon's ant on the beach. The design principle states that in designing and building agents, we have to take the "frame-of-reference" principle into account very carefully. The case study involves a number of simple self-built autonomous robots, the Didabots (Maris and Schaad, 1995). There is an arena with a number of styropor cubes and some Didabots (figure 6). The Didabots are programmed as simple Braitenberg vehicles with only one type of sensor for proximity. All they can do is avoid obstacles. Now look at the sequence of pictures shown in figure 7.

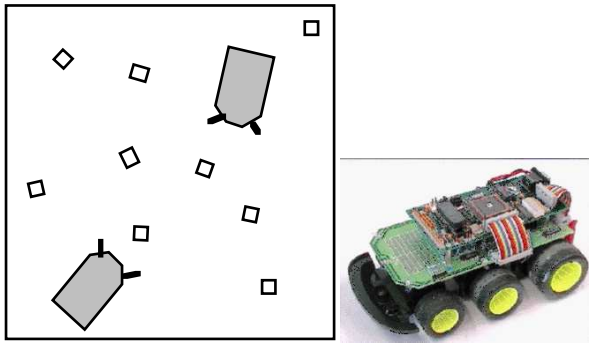

Figure 6: Arena with Didabots and styropor cubes. On the right, a picture of a Didabot is shown.

Initially the cubes are randomly distributed. Over time a number of clusters are forming. At the end there are only two clusters and a number of cubes along the walls of the arena. What would you say the robots are doing?

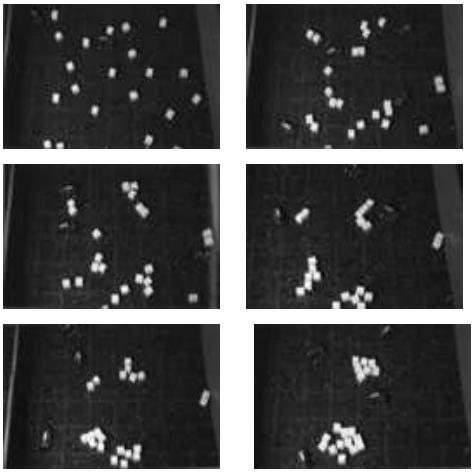

Figure 7: Sequence of situations after a few minutes each. The whole process until a relatively stable situation is achieved lasts roughly 20m.

"They are cleaning up". "They are trying to get the cubes into clusters". These are answers that we often hear. They are fine if we are aware of the fact that they represent an observer's perspective. They describe the behavior. The second answer is a bit problematic since it attributes an intention by using the word "trying". Because we are the designers, we can say very clearly what the robots were programmed to do: to avoid obstacles!

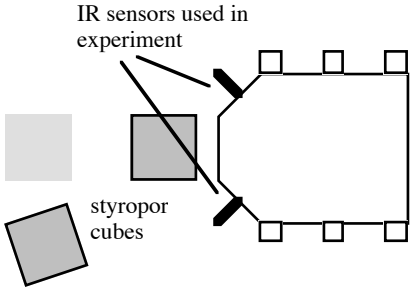

Figure 8: Explanation of the "cleaning up" behavior of the Didabots.

The Didabots only use the two sensors which are marked in black, namely front left and front right. Normally they move forward. If they get near an obstacle within reach of one of the sensors (about 20cm) they simply turn toward the other side. If they encounter a cube head on, neither the left nor the right sensor measure any reflection and the Didabot simply continues moving forward. At the same time it pushes the cube. But it pushes the cube because it doesn't "see" it, not because it was programmed to push it.

For how long does it push the cube? Until the cube either moves to the side and the Didabot loses it, or until it

encounters another cube to the left or the right. It then turns away, thus leaving both cubes together (figure 8). Now there are already two cubes together, and the chance that another cube will be deposited near them is increased. Thus, the robots have changed their environment which in turn influences their behavior.

While it is not possible to predict exactly where the clusters will be formed, we can predict with high certainty that one or two clusters will be formed in environments with the geometrical proportions used in the experiment (systematic experiments have been reported by Maris and te Boekhorst, submitted). Thus, we can make predictions, but they are of a different nature than what we are normally used to, say, in physics.

The kind of phenomenon that we have seen in this experiment is also called self-organization. The behavior of the individual leads to a global change, namely the arrangement of the cubes, which in turn influences the behavior of the individuals. As is well-known, self-organization is a ubiquitous phenomenon in the world around us, in biological, social, economic, engineering, and inanimate physical systems.

Because this phenomenon of cluster formation is stable and consistently occurs, given the right conditions and proportions, we can in fact design a cleaning robot without programming explicitly a representation of cleaning into the robots. The task of cleaning is in our minds as designers, it does not have to be in the "minds" of the robots. To use another buzzword, the behavior of such an agent is sometimes called emergent, and the engineering principle "designing for emergence" (Steels, 1991).

5.4 Developmental explanations

Learning and development—a frame-of-reference issue

Developmental explanations (explanations in terms of learning) refer not only to the current situation, but to events in the past. Again, there is a frame-of-reference issue, here. Think about learning, for a moment. What do we mean by the term? Assume that there is an environment with small and large pegs. Initially, the robot will try to pick up all the pegs. After some time the robot only picks up the small ones and ignores the large ones. We call this the current behavior. We then say that the robot has *learned* a distinction between small and large pegs. We can explain the current behavior, for example, by saying that the robot has encountered small and large pegs along the way and found that the large ones are too heavy to pick up. We resort to the history of the interaction of the agent with its environment, in other words, to its individual "experience". But we are *not* saying anything about the internal mechanism. This is not necessary to define learning.

We will proceed by reviewing a number of design principles. They can be applied to all three types of explanations, functional, developmental, and evolutionary. We will first focus on the developmental aspects. In subsection 5.5 we will briefly illustrate some evolutionary considerations.

The "value" principle

This principle states that the agent has to be embedded in a value system, and that it must be based on self-supervised learning mechanisms employing principles of self-organization. If the agent is to be autonomous and situated it has to have a means to judge what is good for it and what isn't. This is achieved by a value system, a fundamental aspect of every autonomous agent.

There is an implicit and an explicit aspect of the value system. In a sense, the whole set-up of the agent constitutes value: the designer decides that it is good for the agent to have a certain kind of locomotion (e.g. wheels), certain sensors (e.g. IR sensors), certain reflexes (e.g. turn away from objects), certain learning mechanisms (e.g. selectionist learning), etc. These values are implicit. They are not represented explicitly in the system. To illustrate the point, let us look at reflexes for a moment. Assume that a garbage collecting robot has the task to collect only small pegs and not large ones. Moreover, it should learn this distinction from its own perspective. The agent is equipped with a number of reflexes: turning away from objects, turning towards an object, and grasping if there has been lateral sensory stimulation over a certain period of time. The value of the first reflex is that the agent should not get damaged. The second and the third reflex increase the probability of an interesting interaction. Note that this interpretation in terms of value is only in the eye of the designer—the agent will simply execute the reflexes.

These reflexes introduce a bias. The purpose of this bias is to speed up the learning process because learning only takes place if a behavior is successful. If the behavior is successful, i.e. if the agent manages to pick up a peg, a value signal is generated. In this case, an *explicit* value system is required. In this way, the intuition that grasping is considered rewarding in itself, can be modeled.

According to the "value" principle, the learning mechanisms have to be based on principles of self-organization, since the categories to be formed are not known to the agent beforehand. Examples are competitive schemes (e.g. Kohonen, 1988a; Martinetz, 1994), or selectionist ones (Edelman, 1987).

This view of value systems and self-organization contrasts with classical thinking. The metaphor of information processing that underlies traditional AI and cognitive science, cannot accommodate self-organization. The "value" principle is supported by many references (e.g. Edelman, 1987; Pfeifer and Verschure, 1992; Pfeifer and Scheier, in press; Thelen and Smith, 1994). It is closely related to the principle of sensory-motor coordination and ecological balance.

The principle of sensory-motor coordination

This principle states that the interaction with the environment is to be conceived as a sensory-motor coordination. Sensory-motor coordination involves the sensors, the control architecture, the effectors, and the agent as a whole. A consequence of this principle is that classification, perception, and memory should be viewed as

sensory-motor coordinations rather than as individual modules (e.g. Dewey, 1896; Douglas, 1993; Edelman, 1987).

One of the fundamental problems of visual perception is object invariance. One and the same object—a peg in the case of our robot—leads to large variations in (proximal) sensory stimulation: the latter strongly depends on distance, orientation, lighting conditions, etc. Normally, perception is viewed as a process of mapping a proximal (sensory) stimulus onto some kind of internal representation. The enormous difficulties of classical computer vision to come to grips with the problem of invariances suggests that there may be some fundamental problems involved. Viewing perception as sensory-motor coordination has a number of important consequences.

From an information theoretic view, the sensory-motor coordination leads to a dimensionality reduction of the high-dimensional sensory-motor space (Pfeifer and Scheier, in press). This reduction allows learning to take place even if the agent moves. In fact, movement itself is beneficial since through its own movement, the agent *generates* correlations in the interaction with the environment. The second important aspect of sensory-motor coordination is the generation of cross-modal associations, including proprioceptive cues originating from the motor system (Thelen and Smith, 1994; Scheier and Lambrinos, 1996).

Figure 9: Infant categorizing objects and building up concepts while engaged in sensory-motor coordination.

Additional support for the principle of sensory-motor coordination comes from developmental studies. There is a lot of evidence that concept formation in human infants is directly based on sensory-motor coordination (Thelen and Smith, 1994; Smith and Thelen, 1993; see figure 9). The concepts of humans are thus automatically "grounded". Similarly, if this principle is applied to artificial agents, the latter will only form fully grounded categories. The symbol grounding problem is really not an issue—anything the agent does will be grounded in its sensory-motor coordination. Note that the terms categorization and concept building are entirely observer-based. They relate only to the behavior of the infant, not to any sort of internal mechanism.

There is another kind of approach that closely relates to this principle, namely active vision (e.g. Ballard, 1991). Vision is not seen as something that concerns only input, but movement is considered to be an integral aspect.

As already alluded to, this view contrasts with the traditional view of perception as a process of mapping a proximal stimulus onto an internal representation. In the view proposed here, the object representation is in the sensory-motor coordination. "Recognizing" an object implies re-enacting a sensory-motor coordination. Most objections to this view of perception have their basis in introspection. The latter has long ago been demonstrated to be a poor guide to research (Nisbett and Wilson, 1977). This principle is supported by numerous research contributions (e.g. Ballard, 1991; Dewey, 1896; Douglas, 1993; Edelman, 1987; Thelen and Smith, 1994; Smith and Thelen, 1993; Scheier and Lambrinos, 1996; Pfeifer and Scheier, in press; Scheier and Pfeifer, 1995)

The principle of "ecological balance"

The principle of "ecological balance" states that there has to be a match between the "complexity" of the sensors, the actuators, and the neural substrate. Moreover, it states that the tasks have to be "ecologically" adequate. The way the term "complexity" is used here, appeals to our everyday understanding: a human hand is more complex than a forklift, a CCD camera more complex than an IR sensor.

From this principle we can get considerable leverage. Let us look at an example illustrating how *not* to proceed. Assume that we have a robot with two motors and a few IR sensors, say the robot Khepera™. In some sense, this design is balanced due to the intuition of the engineers that built it (except that its processor is too powerful if it is fully exploited). Assume further that some researchers have become frustrated because with the IRs they can only do very simple experiments. They would like to do more interesting things like landmark navigation.

The next logical step for them is to add a CCD-camera. It has many more dimensions than the few IR sensors. The rich information from the camera is transmitted to a central device where it is processed. This processing can, for example, consist in extracting categories. But the categories are formed as a consequence of a sensory-motor coordination. Because the motor system of the agent is still the same, the resulting categories will not be much more interesting than before (although they may be somewhat different). Trying to build categories using only the visual stimulation from the camera (not as a sensory-motor coordination) would violate the principle of sensory-motor coordination. Classical computer vision has violated this principle—and the problems are well-known. It would be a different story if, together with the CCD camera, additional motor capabilities would have been added to the robot, like a gripper or an arm of sorts. Figure 10 shows a balanced design on the left, an unbalanced one in the middle, and again a more balanced one on the right.

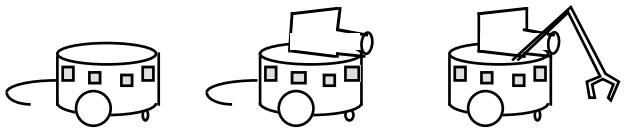

Figure 10: Balanced design on the left, unbalanced design in the middle, and again more balanced design on the right.

An approach that is fully compatible with the principle of "ecological balance" is the Cog project for developing a humanoid robot (Brooks, 1994; Brooks and Stein, 1993). More sophistication on the sensor side (two eyes, each with a camera for peripheral and foveal vision), is balanced by more complexity on the motor side. The arm and the hand are quite sophisticated. Moreover, the head and the eyes can all move which leads to a system of a very large number of degrees of freedom. A lot of the processing is done peripherally, and the central processing capacity is not inflated artificially. It is not surprising that Cog fulfills this design principle. It was Brooks who pointed out that tasks need to be ecologically appropriate (Brooks, 1990). In particular he argued that "elephants don't play chess." We couldn't agree more.

Important evidence for this principle comes also from studies in infant psychology by Bushnell and Boudreau (1993). Their results suggests that there is in fact a kind of co-evolution in the sensory-motor development of the infant. Roughly speaking, acuity of visual distinctions highly correlates with precision of motor movement.

Again, this view sharply contrasts with traditional AI and cognitive science, where intelligence was seen as centralized information processing, with no, or very little consideration given to the physical set-up. A concept like "ecological balance" would not make sense in that framework. References supporting this principle include Brooks, 1991, 1994; Pfeifer, 1995; Smith and Thelen, 1993; Bushnell and Boudreau, 1993.

The principle of parallel, loosely coupled processes

Let us just mention very briefly this principle without going into any detail. In essence, it states that intelligence is emergent from a large number of parallel, loosely coupled processes. These processes run asynchronously and are largely peripheral, requiring little or no centralized resources. It could also be called the "anti-homunculus" principle. One of the main claims here is that coherent behavior can be achieved without central control. A beautiful example that fully endorses this principle is, again, the Cog project (Brooks, 1994; Brooks and Stein, 1993). In our own work we have applied this principle to all our agents (e.g. Scheier and Pfeifer, 1995; Pfeifer and Scheier, in press).

This principle contrasts sharply with classical thinking where a centralized seat of intelligence is assumed. Classical thinking does not object to parallel processes (as we have seen in connectionism). The objection is that coherence cannot be achieved unless there is central integration. The classical view that maintains integration is necessary is especially predominant in psychology, in particular cognitive psychology.

A lot of research in the field of cognitive science and "New AI" supports the principle of parallel, loosely coupled processes (e.g. Braitenberg, 1984; Brooks, 1991; Brooks and Stein, 1993; Maes, 1991; Steels, 1992; Scheier and Pfeifer, 1995; Pfeifer and Scheier, in press).

The principle of "cheap designs"

The last principle that we will discuss is the one of "cheap design". It states that good designs are "cheap". "Cheap", as used here, has several meanings. For the purposes of this paper, it means parsimonious. Moreover, this parsimony is to be achieved through exploitation of the physics. A nice illustration is insect walking. Leg coordination in insects does not require a central controller. There is no internal process corresponding to global communication between the legs, they communicate only locally with each other (e.g. Cruse, 1991). But there *is* global communication between all the legs, namely through the environment. It is mediated by a physical process, not by an information process (or a process of signal transfer) within the agent. If the insect lifts one leg, the force on all other legs is changed instantaneously because of the weight of the insect. This saves the insect a lot of—unnecessary—neural substrate. Another example of a cheap design are the Didabots which are cleaning up the arena of styropor cubes. There is ample evidence supporting this principle (e.g. e Brooks, 1991; Cruse, 1991; Horsewill, 1992; Franceschini et al., 1992; Pfeifer, 1993, 1995; Thorpe and Imbert, 1989.

5.5 Evolutionary explanations

We do not want to overstress this point, but some of the design principles have interesting evolutionary interpretations. Take, for example, the principle of "ecological balance". Natural designs are ecologically balanced. It seems that evolution favors balanced designs. Recent developments in evolutionary robotics have suggested simulated evolution as a design principle (e.g. Harvey et al., in press). It would be interesting to see whether eventually balanced designs will emerge from these efforts. Of course, this would require evolving complete agents, not only control architectures (as is currently done).

In our research we have mostly been focusing on the functional and the learning/developmental perspectives. In the future we will include evolutionary principles into our considerations.

6 Discussion

We have now completed our argument. We began by pointing out some fundamental problems of symbol processing models and defined the "information processing view". We then showed to what extent connectionist models resolve some of these issues. They represent an important development in the right direction, because they process patterns of activation rather than symbols. But we saw that connectionist models—for the

better part—remain within the information processing paradigm. They typically are based on basic (symbolic) categories that are defined by the designer. This holds for supervised as well as for non-supervised schemes. A very different kind of thinking is needed if we are to understand and design systems which have to interact with the real world. And it seems, that the history of AI, cognitive science, and robotics has taught us that intelligence always requires the interaction with the real physical world.

The paradigm of "New AI", of employing embodied physical agents (typically in the form of autonomous robots) as a research tool, helps us asking the right, questions. The questions that we have to ask, relate to behavior of complete agents. It is amazing how much our view of intelligence changes with this perspective. All of a sudden, it seems possible to overcome some of the fundamental problems of the traditional approach.

An example is the symbol grounding problem. As we have seen, the categories and concepts an agent acquires, will be grounded if we focus on sensory-motor coordination. We have not "resolved" the symbol grounding problem because it does not need to solved. But we have shown how we can design agents without getting trapped in it. Likewise, it will not be possible to entirely eliminate the frame problem. However, the principle of "ecological balance" tells us that we should not artificially increase the complexity of the neural substrate (which would be necessary if the agent were to build sophisticated models of the environment) if the sensory-motor system remains the same. Thus, if we observe this design principle, we are much less in danger of building models that are too complex for the specific agent-environment interaction. Or recall the computational problems involved in perception. By viewing perception as sensory-motor coordination, the computational complexity can be dramatically reduced. Note that as a side-effect of applying the design principles, real-time performance increases because less processing is required.

There is another point of concern. Initially, when discussing the classical approach we introduced the notion of representation. There are these mappings between the outside world and the agent (called "encode" and "decode"). Connectionst models typically do not deal appropriately with these mappings, because they are given by the input and output categories of the model (letters and phoneme features). They are predefined by the designer, not acquired by the model itself. In real-world agents, this mapping is mediated by the physics of the agent. It turns out that if we are trying to interpret the weight patterns and activation patterns in a neural network, this is only possible if we know how the sensory and motor systems function, and *where they are physically positioned on the robot*. Trying to find where categories are represented—remember that the categories are observer-based (frame-of-reference)—in a network, is a task that can only be achieved if it is exactly known how this network is embedded in the agent. Otherwise, activation levels and connection strengths have *no meaning*: they cannot be abstracted out. Thus, representation is no longer a property of a formalism, or of a mapping, but a property of a complete agent.

As a last point for discussion let me anticipate your questions, namely "will this approach of 'New AI' ever scale up? Will we ever be able to solve the same kinds of problems that we have been able to solve using the classical approach?" The answer depends on what we mean by the sentence "that we have been able to solve using the classical approach". It could mean that we have been able to build systems that support humans in their work. In this case, the research goal has not been to develop *intelligent* systems but useful computer systems. If the intended meaning is that we have been able to model human expertise, I would argue that—perhaps with some exceptions—classical models only capture very limited aspects of human intelligence (or expertise), and perhaps not the most interesting ones.

And this brings me to the concluding remark. It has often been suggested that for the low-level competences, the sensory-motor aspects, connectionist models, or "New AI" style models might be appropriate, but that for the high-level part we may need to resort to symbol processing concepts. While from an engineering perspective there is little to be argued if it works, from a cognitive science perspective, it can be predicted that this approach will not lead to interesting insights. The reason is that this way of proceeding constitutes a category error: on the one hand the categories are built up by the agent itself, i.e. they are agent-based, whereas the ones used in the symbolic system, are designer-based—once again, a "frame-of-reference" issue.

The design principles outlined above do not cover all the insights of the very rich field of "New AI". But we do believe that they capture a large part of the most essential aspects of what has emerged from pertinent research. The principles described may seem somewhat vague and overly general, but they are enormously powerful as heuristics, providing guidelines as to what sorts of experiments to conduct next and what agents to design for future experiment. In order to achieve some degree of generality we have deliberately left out a lot of detail. These principles not only help us evaluate existing designs, but they get us to ask the right questions.

In the future we might be looking for something more formal, than merely a set of verbally stated design principles. Eventually, this will certainly be necessary. But what this "theory" will look like, is entirely open.

What is needed right now is an in-depth discussion of these design principles. They have to be revised and the list of principles has to be augmented.

7 Conclusions

I hope that it has been shown, that if we are to understand intelligence, we need more than the classical tools of symbol processing AI. But connectionism alone will not solve the fundamental problems either. We need to take the physics of the agent and how it interacts with its environment into account. In other words we need to go

beyond the information processing metaphor. We have outlined a number of principles that will hopefully form the basis for a productive discussion.

Acknowledgments

This paper includes materials from various sources. The "Encyclopedia of Microcomputers" (Pfeifer, 1996a), the "Proceedings of the 4th International Conference on the Simulation of Behavior: From Animals to Animats" (Pfeifer, 1996b), and the "Journal of Robotics and Autonomous Systems" (Pfeifer, 1995). It also contains material from a presentation given at ER'96 at the Canadian Embassy in Tokyo, April 1996. The research reported was performed largely under the auspices of the Swiss National Science Foundation, grants # 20-40581-94 and 32-37819.93. I am grateful to Fumio Mizoguchi and Ryuichi Oka for inviting me to write this paper. I would like to thank Christian Scheier, Dimitri Lambrinos, and Ralf Salomon for their inspiring discussions and valuable comments on the manuscript.

References

Ballard, D.H. (1991). Animate vision. *Artificial Intelligence*, **48**, 57-86.

Beer, R. (in press). The dynamics of adaptive behavior: a research program. To appear in: *Robotics and Autonomous Systems, Special Issue on "Practice and Future of Autonomous Agents",* R. Pfeifer, and R. Brooks (eds.).

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

Brooks, R.A. (1990). Elephants don't play chess. *Robotics and Autonomous Systems*, **6,** 3-15.

Brooks, R.A. (1991). Intelligence without representation. *Artificial Intelligence*, **47**, 139-160.

Brooks, R.A. (1994). Coherent behavior from many adaptive processes. In: D. Cliff, P. Husbands, J.-A. Meyer, and S.W. Wilson (eds.). *From animals to animats 3. Proc. SAB'94,* 22-29.

Brooks, R.A., and Stein, L.A. (1993). Building brains for bodies. Memo 1439, MIT Artificial Intelligence Laboratory, Cambridge, Mass.

Bushnell, E.M. and Boudreau, J.P. (1993). Motor development in the mind: The potential role of motor abilities as a determinant of aspects of perceptual development. *Child Development*, **64**, 1005-1021.

Chapman, D. (1987). Planning for conjunctive goals. *Artificial Intelligence*, **32**, 333-337.

Clancey, W.J. (1991). The frame of reference problem in the design of intelligent machines. In K. van Lehn (ed.). *Architectures for intelligence*. Hillsdale, N.J.: Erlbaum.

Cognitive Science (1993), **17**. Norwood, N.J.: Ablex Publ.

Cruse, H. (1991). Coordination of leg movement in walking animals. In: J.-A. Meyer, and S.W. Wilson (eds.). *From animals to animat 1. Proc. SAB'90,* 105-119.

Dennett, D. (1987). Cognitive wheels. In Z.W. Pylyshyn (ed.) (1987). *The robot's dilemma. The frame problem in artificial intelligence*. Norwood, N.J.: Ablex (2nd printing 1988).

Dewey, J. (1896). The reflex arc concept in psychology. *Psychol. Rev.*, **3** (1981) 357-370; Reprinted in: J.J. McDermott (ed.) *The Philosophy of John Dewey*. Chicago, IL: University of Chicago Press, 136-148.

Douglas, R.J., Martin, K.A.C., and Nelson, J.C. (1993). The neurobiology of primate vision. *Bailliere's Clinical Neurology*, **2**, No. 2, 191 - 225.

Edelman, G.E. (1987). *Neural Darwinism. The theory of neuronal group selection*. New York: Basic Books.

Franceschini, N., Pichon, J.M., and Blanes, C. (1992). From insect vision to robot vision. *Phil. Trans. R. Soc. Lond. B*, **337**, 283-294.

Harnad, S. (1990). The symbol grounding problem. *Physica D*, **42**(1-3), 335-346.

Harvey, I., Husbands, P., Cliff, D., Thompson, A., Jakobi, N. (1996). Evolutionary robotics: the Sussex approach. *Robotics and Autonomous Systems, Special Issue on "Practice and Future of Autonomous Agents"*, R. Pfeifer, and R. Brooks (eds.).

Horsewill, I. (1992). A simple, cheap, and robust visual navigation system. In: J.-A. Meyer, H.L. Roitblat, and S.W. Wilson (eds.). *From animals to animats 2. Proc. SAB'92,* 129-137.

Janlert, L.E. (1987). Modeling change—the frame problem. In Z.W. Pylyshyn (ed.) (1987). *The robot's dilemma. The frame problem in artificial intelligence*. Norwood, N.J.: Ablex (2nd printing 1988).

Kohonen, T. (1988a). *Self-organization and associative memory*. Berlin: Springer.

Kohonen, T. (1988b). The "Neural" phonetic typewriter. *IEEE Computer*, 1988, March, 11-22.

Laird, J.E., Newell, A., and Rosenbloom, P.S. (1987). SOAR: an architecture for general intelligence. *Artificial Intelligence,* **33**, 1-64.

Maes, P. (1991). A bottom-up mechanism for behavior selection in an artificial creature. In: J.-A. Meyer, and S.W. Wilson (eds.). *From animals to animats 1. Proc. SAB'90,* 238-246.

Maris, M., and Schaad, R. (1995). The didactic robots. University of Zurich, AI Lab Techeport, #95.05.

Maris, M., and te Boekhorst, R. (submitted). Exploiting physical constraints: heap formation through behavioral error in a group of robots (submitted to IROS'96).

Martinetz, T. (1994). Topology representing networks. *Neural Networks*, **7**, 505-522.

McCarthy, J., and Hayes, P.J. (1969). Some philosophical problems from the standpoint of artificial intelligence. In: B. Meltzer, and D. Michie (eds.), *Machine Intelligence,* **4**, 463-502.

McFarland, D., and Bösser, M. (1993). *Intelligent behavior in animals and robots*. MIT Press.

Nisbett, and Wilson (1977). Telling more than we can know: verbal reports of mental processes. *Psychological Review*, **84,** 231-259.

Newell, A. (1990). Unified theories of cognition. Cambridge, Mass.: Harvard University Press.

Newell, A., and Simon, H.A. (1976). Computer science as empirical inquiry: symbols and search. *Comm. of the ACM*, **19**, 113-126.

Pfeifer, R. (1993). Cheap designs: exploiting the dynamics of the system-environment interaction. Three case studies on navigation. In: Conference on *"Prerational Intelligence"*. Center for Interdisciplinary Research, University of Bielefeld, 81-91.

Pfeifer, R. (1995). Cognition—perspectives from autonomous agents. *Robotics and Autonomous Systems*, **15**, 47-70.

Pfeifer, R. (1996a). Symbols, patterns, and behavior: beyond the information-processing metaphor. In A. Kent and J.G. Williams (eds.). *Encyclopedia of Microcomputers*, Vol. **17**, 253-275. New York: Marcel Dekker, Inc.

Pfeifer, R. (1996b). Building "Fungus Eaters": Design principles of autonomous agents. *Proc. SAB'96.*

Pfeifer, R., and Scheier, C. (in press). Sensory-motor coordination: the metaphor and beyond. To appear in: *Robotics and Autonomous Systems, Special Issue on "Practice and Future of Autonomous Agents",* R. Pfeifer, and R. Brooks (eds.).

Pfeifer, R., and Verschure, P.F.M.J. (1992). Distributed adaptive control: a paradigm for designing autonomous agents, in F.J. Varela, and P. Bourgine (eds.) *Proc. ECAL-92*, 21-30.

Putnam, H. (1975). Philosophy and our mental life. In H. Putnam (ed.) *Mind, language and reality: philosophical papers, vol. 2*. Cambridge: Cambridge University Press.

Pylyshyn, Z.W. (ed.) (1987). *The robot's dilemma. The frame problem in artificial intelligence*. Norwood, N.J.: Ablex (2nd printing 1988).

Quillian, R. (1968). Semantic memory. In M. Minsky (ed.). *Semantic information processing*. Cambridge, Mass.: MIT Press.

Reeke, G.N. Jr., and Edelman, G.M. (1988). Real brains and artificial intelligence. *Daedalus*, **117**(1), 143-173.

Rumelhart, D., and McClelland J. (1986). *Parallel distributed processing*. Cambridge, Mass.: MIT Press.

Scheier, C., and Lambrinos, D. (1996). Categorization in a real-world agent using haptic exploration and active vision. *Proc. SAB'96.*

Scheier, C., and Pfeifer, R. (1995). Classification as sensory-motor coordination: a case study on autonomous agents. *Proc. ECAL-95,* 657-667.

Sejnowski, T.J., and Rosenberg, C.R. (1987). Parallel networks that learn to pronounce English text. *Complex Systems*, **1**, 145-168.

Simon, H.A. (1969). *The sciences of the artificial.* Cambridge, Mass.: MIT Press.

Smith, L.B., and Thelen, E. (eds.) (1993). *A dynamic systems approach to development. Applications*. Cambridge, Mass.: MIT Press, Bradford Books.

Steels, L. (1991). Towards a theory of emergent functionality. In: J.-A. Meyer, and S.W. Wilson (eds.). *From animals to animats 1. Proc. SAB'90,* 451-461.

Steels, L. (1992). The PDL reference manual. VUB AI Lab memo 92-5.

Steinhage, A., and Schöner, G. (in press). Self-calibration based on invariant view recognition: Dynamic approach to navigation. To appear in: *Robotics and Autonomous Systems, Special Issue on "Practice and Future of Autonomous Agents",* R. Pfeifer, and R. Brooks (eds.).

Suchman, L. (1987). *Plans and situated action*. Cambridge University Press.

Thelen, E. and Smith, L. (1994). *A dynamic systems approach to the development of cognition and action*. Cambridge, Mass.: MIT Press, Bradford Books.

Thorpe, S.J., and Imbert, M. (1989). Biological constraints on connectionist modelling. In R. Pfeifer, Z. Schreter, F. Fogelman-Soulié, and L. Steels (eds.). *Connectionism in Perspective*. Amsterdam: North-Holland, 63-92.

Verschure, P.F.M.J. (1992). Taking connectionism seriously: the vague promise of subsymbolism and an alternative. *Proceedings of The Fourteenth Annual Conference of The Cognitive Science Society*.

Winograd, T. and Flores, F. (1986). *Understanding computers and cognition*. Reading, Mass.: Addison-Wesley.