

# The artificial life roots of artificial intelligence

Luc Steels

Artificial Intelligence Laboratory

Vrije Universiteit Brussel

Pleinlaan 2, B-1050 Brussels, Belgium

E-mail: `steels@arti.vub.ac.be`

November 4, 1993

## Abstract

Behavior-oriented AI is a scientific discipline that studies how behavior of agents emerges and becomes intelligent and adaptive. Success of the field is defined in terms of success in building physical agents that are capable of maximising their own self-preservation in interaction with a dynamically changing environment. The paper addresses this artificial life route towards artificial intelligence and reviews some of the results obtained so far.

Official reference: Steels, L. (1994) The artificial life roots of artificial intelligence. Artificial Life Journal, Vol 1,1. MIT Press, Cambridge.

## 1 Introduction

For several decades, the field of Artificial Intelligence has been pursuing the study of intelligent behavior using the methodology of the artificial [104]. But the focus of this field, and hence the successes, have mostly been on higher order cognitive activities such as expert problem solving. The inspiration for AI theories has mostly come from logic and the cognitive sciences, particularly cognitive psychology and linguistics. Recently, a subgroup within the AI community has started to stress embodied intelligence and made strong alliances with biology and research on artificial life [59]. This is opening up an ‘artificial life route to artificial intelligence’ [112], which has been characterised as Bottom-Up AI [19], the Animat approach [133], Behavior-based AI [108], or Animal Robotics [75]. These terms identify a loose network of engineers and biologists who share the common goal of understanding intelligent behavior through the construction of artificial systems. The researchers also share a growing number of assumptions and hypotheses about the nature of intelligence. In view of the strong links with biology and complex systems theory, the research has so far received more attention in the Artificial Life community than in the Artificial Intelligence field itself.

The aim of this paper is to review this approach and identify some major unresolved

issues. Given that substantial engineering efforts and non-trivial experimentation is required, the first solid experimental and technical results have only recently begun to appear. Good sources for tracking the field are the conferences on the Simulation of Adaptive Behavior ([80], [79]) and the associated journal [102], the conferences on Artificial Life ([59], [60], [124], [30]), and the associated journal [4]. There are also occasional contributions to international conferences on AI (such as IJCAI, AAAI, or ECAI), neural networks (NIPS), or robotics (IEEE). Reports of some milestone workshops have been published ([65], [123], [112], [113]).

Section 2 of the paper delineates the artificial life approach to artificial intelligence. Section 3 identifies the fundamental units of this approach, which are behavior systems. Section 4 and 5 focus on contributions towards a central theme of artificial life research, which is the origin of complexity through emergent functionality. A short review of some other issues concludes the paper.

## **2 Delineating the field.**

### **2.1 The subject matter is intelligent behavior.**

The phenomena of interest are those traditionally covered by ethology and ecology (in the case of animals) or psychology and sociology (in the case of humans). The behavior by an individual or a group of individuals is studied, focusing on what makes behavior intelligent and adaptive and how it may emerge. 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 [106]. Behavior is intelligent if it maximises preservation of the system in its environment. The main emphasis is not on the physical basis of behavior, as in the case of neural network research, but on the principles that can be formulated at the behavioral level itself. An example of a theory at the behavioral level is one which explains the formation of paths in an ant society in terms of a set of behavioral rules without reference to how they are neurophysiologically implemented [91]. Another example is a study of how certain behavioral strategies (such as retreat when attacked) and their associated morphological characteristics are evolutionary stable [72].

Given this emphasis on behavior, the term *behavior-oriented* seems appropriate to distinguish the field, particularly from the more knowledge-oriented approach of classical AI. It will be used in the rest of the paper.

## **2.2 The methodology is based on building artificial systems.**

Scientists traditionally construct models in terms of a set of equations which relate various observational variables and hypothesised theoretical variables. Technological advances in the second half of this century have resulted in two additional types of models:

- *Computational models*: These consist of a process-oriented description in terms of a set of datastructures and algorithms. When this description is executed,

i.e., the algorithm is carried out causing the contents of the datastructures to be modified over time, phenomena can be observed in the form of regularities in the contents of the datastructures. If these synthetic phenomena show a strong correspondence with the natural phenomena, they are called simulations and the process descriptions constitute a theory of the natural phenomena.

- *Artificial models:* One can also construct a physical device (an artifact) whose physical behavior gives rise to phenomena comparable to the natural phenomena in similar circumstances. The device will have components with a particular structure and functioning which have been put together in a particular way. The design and implementation of these components and their mode of combination constitutes another possible way to theorise about the phenomena.

Computational models and artificial models, or, what Pattee [92] calls simulations and realisations must be clearly distinguished. For example, it is possible to build a computational model of how a bird flies, which amounts to a simulation of the environment around the bird, a simulation of the aerodynamics of the body and the wings, a simulation of the pressure differences caused by movement of the wings, etc. Such a model is highly valuable but would however not be able to fly. It is forever locked in the datastructures and algorithms implemented on the computer. It flies only in a virtual world. In contrast, one could make an artifact in terms of physical components (a physical body, wings, etc.). Such an artifact would only be viewed as

satisfactory if it is able to perform real flying. This is a much stronger requirement. Very often results from simulation only partially carry over to artificial systems. When constructing a simulation, one selects certain aspects of the real world that are carried over into the virtual world. But this selection may ignore or overlook essential characteristics which play a role unknown to the researcher. An artificial system cannot escape the confrontation with the full and infinite complexity of the real world and is therefore much more difficult to construct.

The term ‘artificial’ in ‘artificial life’ (and also in ‘artificial intelligence’) suggests a scientific approach based on constructing artificial models. The methodological steps are as follows: a phenomenon is identified (e.g. obstacle avoidance behavior), an artificial system is constructed that has this as competence, the artificial system is made to operate in the environment, the resulting phenomena are recorded, these recordings are compared with the original phenomena. Potential misfits feed back into a redesign or re-engineering of the artificial system.

Although Artificial Intelligence is sometimes equated with the simulation of intelligent behavior, this is too narrow an interpretation. The goal is to build artifacts which are ‘really’ intelligent, i.e. intelligent in the physical world, not just intelligent in a virtual world. This makes the construction of robotic agents that must sense the environment and can physically act upon the environment, unavoidable. Particularly if sensori-motor competences are studied. This is why behavior-oriented AI researchers insist so strongly on the construction of physical agents [21], [130].

Performing simulations of agents (as in [15]) is of course an extremely valuable aid in exploring and testing out certain mechanisms, the way simulation is heavily used in the design of airplanes. But a simulation of an airplane should not be confused with the airplane itself.

### **2.3 Behavior-oriented AI is strongly influenced by biology.**

We have already identified two key ingredients of the behavior-oriented approach: the study of intelligent behavior, and the methodology of constructing artificial systems. The third ingredient is a strong biological orientation. Intelligence is seen as a biological characteristic, and the "core of intelligence and cognitive abilities is [assumed to be] the same as the capacity of the living" ([124], backcover).

The biological orientation clearly shows up in the way intelligence is defined. The 'classical' AI approach defines intelligence in terms of knowledge: A system is intelligent if it maximally applies the knowledge that it has (cf. Newell's principle of rationality [87]). The behavior-oriented approach defines intelligence in terms of observed behavior and self-preservation (or autonomy) (see e.g. [124], [76]). It is based on the idea that the essence of biological systems is their capacity to continuously preserve and adapt themselves [71]:

*The behavior of a system is intelligent to the extent that it maximises the chances for self-preservation of that system in a particular environment.*

The drive towards self-preservation applies to all levels of complexity: genes, cells, multi-cellular structures, plants, animals, groups of animals, societies, species. Behavior-oriented AI focuses upon the behavior of organisms of the complexity of animals. Systems of this complexity are called agents. When several of them cooperate or compete, we talk about multi-agent systems.

In order to explain how a system preserves itself even if the environment changes, adaptivity and learning are corollary conditions of viable intelligent agents:

*A system is capable of adapting and learning if it changes its behavior so as to continue maximising its intelligence, even if the environment changes.*

The biological orientation also shows up in a focus on the problem how complexity can emerge. The origin of order and complexity is a central theme in biology [53] and is usually studied within the context of self-organisation [95], or natural selection [16]. Behavior-oriented AI research is focusing on the concepts of emergent behavior and emergent functionality as a possible explanation for the emergence of functional complexity in agents. These concepts will be discussed in more detail later. A preliminary definition is as follows:

*A behavior is emergent if it can only be defined using descriptive categories which are not necessary to describe the behavior of the constituent components. An emergent behavior leads to emergent functionality if the*



*behavior contributes to the system's self-preservation and if the system can build further upon it.*

Behavior-oriented AI studies the origin of complexity at different levels: from components and complete agents to multi-agent systems. Systems at each level maximise their self-preservation by adapting their behavior so that it comes closer to the optimal. Co-adaptation ensures that different elements at one level contribute to the optimality of the whole. At every level there is cooperation and competition: Different agents cooperate and compete inside a multi-agent system. Different behavior systems cooperate and compete inside the agent. Different components cooperate and compete to form coherent behavior systems. So the ingredients of cooperation, competition, selection, hierarchy, and reinforcement, which have been identified as crucial for the emergence of complexity in other areas of biology [59], are found at the behavioral level, making it possible to carry over results from other biological disciplines to behavior-oriented AI and vice versa.

All of the elements of the above definitions for intelligence, adaptivity, and emergence can be quantitatively and objectively established. We can quantify the aspects of the environment which act as pressures on the system considered, the success in self-preservation, the optimality of particular behaviors with respect to their contribution to self-preservation, and the success of adaptation and learning to improve this optimality. All this has already been illustrated by McFarland and Boesser [76]. We can also quantitatively identify the onset of emergence once a suitable mathematical

framework exists for defining the notion of a minimal description. An example of such a framework can be found in Chaitin’s work on algorithmic complexity (see the discussion in [89]) The objective nature of these definitions makes them preferable to those relying on the subjective assignment of knowledge or on subjective criteria of similarity to human intelligence as in the Turing test.

## **2.4 Behavior-oriented AI is complementary to other approaches to artificial intelligence.**

The behavior-oriented approach is complementary to the currently dominating trend in AI (also known as the classical approach) which is almost exclusively concentrated on the problems of identifying, formalising, and representing knowledge [38]. The emphasis on knowledge leads almost automatically to a focus on disembodied intelligence. Classical AI systems therefore do not include a physical body nor sensing or acting. If intelligent robots have been considered (as in [90]), sensing and action has been delegated to subsystems which are assumed to deliver symbolic descriptions to the central planning and decision making modules. Moreover knowledge-oriented theories do not include environmental pressures on the self-preservation of the agent, and the role of adaptivity and emergence is taken over by the programmer. However, the claim (made for example in [66]) that the classical, knowledge-oriented approach works only for “simulated toy problems” and makes too many simplifying assumptions (e.g., static environments, single tasks, etc.) is simply not true. Objective

results achieved in knowledge engineering for large-scale, extremely challenging real world problems (like the assignment of train engines and personnel to routes taking into account a large number of possibly conflicting constraints, or the diagnosis of printed circuit boards assembled in digital telephone switch boards) cannot and should not be dismissed.

The behavior-oriented approach is also complementary to the artificial neural network approach, which is based on an even more radical bottom-up attitude because it focuses on the physical basis of behavior and hopes that this is sufficient to explain or synthesise intelligence [56], i.e., that no separate behavioral level is necessary. The distinction between the two fields is of course a matter of degree. Behavior-oriented researchers heavily make use of neural network techniques to implement certain aspects of an overall design and some neural network researchers are beginning to consider the problem of building complete agents (cf. Edelman's NOMAD [34]).

There are obviously strong ties between behavior-oriented AI and robotics, because the construction of physical agents is seen as a *conditio sine qua non* for applying the method of the artificial properly. But the two fields should not be equated. The goal of robotics is to identify, design, and engineer the most reliable and most cost-effective solution for a sensori-motor task in a particular, usually fixed and known, environment [17]. Behavior-oriented AI uses the tools of roboticists to study biological issues, but very different criteria for success apply.

## 2.5 The rest of the paper focuses on emergence.

A review of the field can be organised along several lines. One way would be to look at the progress towards the achievement of specific competences, for example, the different approaches for ‘navigation towards a target’: using potential fields [7], cognitive maps with landmarks [70], phonotaxis [129], global reference frames [86], pheromone trails or agent chains [41], and so on. Another approach would be to review the large amount of work on building technical hardware and software platforms which now make it possible to execute experiments easily and at low cost [35], [31], [50]. This technical work is in some way a revival of earlier cybernetics work by Walter [128] and Braitenberg [18] but now with better hardware and more advanced software. Yet another way is to look at progress on the theoretical questions outlined earlier, for example the definition and use of optimality criteria [76], or the development of quantitative behavioral descriptions using techniques from complex systems theory [89].

These overviews would all be valuable but require much more space than available here. Instead, we will focus on how behavior-oriented AI may contribute to the field of Artificial Life as a whole, and more specifically to its central research theme, which is the origin of complexity. The focus on the interaction between physical agents and the world through sensing and effecting introduces a special perspective which is not found in other Artificial Life work. The emergence of complexity must come through the dynamics of interacting with an infinitely complex, dynamically

changing, real world and not only through the internal dynamics as in the case of cellular automata for example.

In order to further limit the scope of the paper, we will only focus on how the behavior of a single agent is established. There is a lot of fascinating work on multi-agent systems and often it is not even possible to study single agents without taking other agents into account. Nevertheless, a review of work on multi-agent systems would have doubled the size of the present paper.

### **3 Behavior systems**

When studying multi-agent systems (like ant-societies), the units of investigation are clearly visible. But the units causing the behavior of a single agent are not directly observable. Sensors, neurons, networks of neurons, propagation processes and actuators, are the obvious building blocks. But many of these must work together and interact with structures and processes in the environment in order to establish a particular behavior, and the same components may dynamically be involved in many different behaviors. This is the reason why it is so difficult to bridge the gap between neurology and psychology.

There is a growing consensus in behavior-oriented AI research that behavior systems be considered as the basic units [19]. Other terms for the basic behavioral unit are task-achieving module [68], or schema [5].

To define the notion of a behavior system we have to make a distinction between a functionality, a behavior, a mechanism, and a component:

- *Functionalities*: A functionality is something that the agent needs to achieve, for example locomotion, recharging, avoiding obstacles, finding the charging station, performing a measurement, signalling another agent. Other terms used for functionality are task, goal, and competence. Functionalities belong to the descriptive vocabulary of the observer.
- *Behaviors*: A behavior is a regularity in the interaction dynamics between an agent and its environment, for example, maintaining a bounded distance from the wall, or having a continuous location change in a particular direction. One or more behaviors contribute to the realisation of a particular functionality. Behaviors belong also to the descriptive vocabulary of the observer. Looking at the same agent in the same environment, it is possible to categorise the behavior in different ways. This does not mean that behavior characterisation is subjective. It can be defined and measured fully objectively.
- *Mechanisms*: A mechanism is a principle or technique for establishing a particular behavior, for example, a particular coupling between sensing and acting, the use of a map, an associative learning mechanism.
- *Components*: A component is a physical structure or process which is used to implement a mechanism. Examples of components are body parts, sensors, actuators, datastructures, programs, communication hardware and software.

A *behavior system* is the set of all mechanisms which play a role in establishing a particular behavior. The structures of a behavior system which can undergo a change due to learning are usually called behavior programs. Observed behavior will of course depend almost as much on the state of the environment as on the mechanisms and components of the agent. Often the name of the behavior system indicates the functionality to which it contributes. But strictly speaking, we should be more careful. For example, there could be a ‘homing in’ functionality achieved by a ‘zig zag behavior’ towards a goal location which is the result of a ‘phototaxis mechanism’. Phototaxis means that the goal location has a light source acting as beacon and that the robot uses light sensors to minimize the distance between itself and the beacon. The reason why we need to be careful in mixing functional and behavior terminology is because the same behavior system may contribute to different functionalities.

Behavior systems may be very simple, implementing direct reflexes between sensing and action (as in [19]). They may also be more complex, building up and using cognitive world maps (as in [70]). When enough complexity is reached, a large collection of interacting behavior systems may resemble a society of interacting agents [84]. Each behavior system is most adapted to a particular class of environments. This environment can be characterised in terms of a set of constraints [48] or cost functions [75].

Note that a behavior system is a theoretical unit. There is not a simple one-to-one

relation between a functionality, a behavior, and a set of mechanisms achieving the behavior. The only thing which has physical existence are the components. This is obvious if emergent functionality comes into play (see section 4 and 5). On the other hand, behavior systems form a real unit in the same way that a society forms a real unit. The interaction between the different mechanisms and the success in the behavior to achieve tasks which contribute to the agent's self-preservation give a positive enforcement to all the elements forming part of a behavior system.

### **3.1 Behavior systems should be viewed as living systems.**

In view of the biological orientation discussed earlier, it is not surprising that many behavior-oriented AI researchers view behavior systems very much like living systems. This means that behavior systems are viewed as units which try to persevere themselves. An analogy with cells which are the smallest biological autonomous units helps to make this concrete (Fig. 1). A cell consists of a group of biochemical structures and processes. The processes are guided by genes, which are themselves represented as molecular structures inside the cell. The processes take place in interaction with material outside the cell which is passing through the cell membrane in both directions. Cells may change their internal structure and functioning, to a certain limit, and thus adapt to the surrounding environment [97].

A behavior system consists also of a set of dynamic and static structures. The structures include physical components like sensors and body parts, as well as net-



cell	behavior system
biochemical processes	transformation processes
biochemical structures	electrical signals and states
genes	behavior programs
incoming material	energy transduced by sensors
outgoing material	energy transduced by actuators
adaptation to cell environment	adaptation to external environment

Figure 1: Comparison between cells and behavior systems. Cells are guided in part by genes whereas behavior systems are guided in part by a behavior program.

works, temporary states and electrical signals propagating in these networks. The internal processes combine and transform signals. These transformation processes are guided by a behavior program which is itself a (distributed) physical structure and can thus be subjected to processes that change it. The transformation processes are partially caused by energy coming from the outside through sensors which convert this to internal energy and they produce signals impacting the actuators which convert internal energy to mechanical energy so that there is a continuous in and outflow of energy to the environment. Behavior systems that change their internal structure and functioning are better adapted to the environment and may better work together with other behavior systems. The main criterion for survival of a behavior system is its utility for the complete agent.

This comparison between cells and behavior systems illustrates several points. (1) It emphasises in the first place that the components of behavior systems are physical systems and that behavior is a physical phenomenon. There are extreme functionalist tendencies in AI (and also in Alife) that equate intelligence or living with

disembodied abstractions, but this is not intended here. (2) The behavior programs and the transformation processes can be interpreted in information processing terms, but that is not necessary and may occasionally be harmful [107]. (3) The transformation processes can be implemented as computational processes but then only if we remind ourselves that computational processes are physical processes, that happen to be instantiated in a physical system of a certain organisation that we call a computer.

The comparison also emphasises the dynamical aspects. Like a cell, a behavior system is continuously active and subjected to in and out flow of energy. Like a cell, a behavior system adapts continuously to changes in the environment. Moreover comparing behavior programs with genes immediately suggests the use of selectionist principles as a way to arrive at new behavior systems without prior design (see section 5.).

A concrete example for obstacle avoidance in an artificial agent may be helpful to clarify the discussion (Fig. 2). Obstacle avoidance can be achieved by a behavior system which maintains a certain distance from obstacles. The components of this behavior system include a left and right infrared sensor, which emit infrared light and capture the reflection coming from obstacles; a translational and rotational motor, which are connected with the wheels and can thus steer the robot left or right; and a behavior program which causes processes to transform the changes in detected infrared reflection into changes in the motor speeds. As already suggested in Brait-



Figure 2: Typical robotic agent used in behavior-oriented AI experiments. The robot has a ring of infrared sensors and a ring of bumper sensors. It has additional light sensors and microphones. There is a translational motor for forward/backward movement and a rotational motor for turning left or right. The agent has a central PC-like processor and dedicated hardware for signal processing and interdevice communication.

enberg [18], obstacle avoidance can be achieved by a direct coupling between infrared reflection and rotational motor speed. If the amount of reflection increases on one side, then the rotational motor speed going in the same direction increases. In a real world environment adaptation is necessary because infrared reflection depends on changing environmental circumstances (for example amount of background infrared in the environment or battery level). Adaptation can here be achieved by incorporating structures that act as ‘weights’ on the effect of increased reflection. When the weights become higher, less reflection will have a greater impact. The weights can be subject to change depending on environmental conditions using Hebbian learning mechanisms (see section 5).

### **3.2 Some guidelines are known for designing behavior systems.**

At the moment, the design of behavior systems for artificial agents is very much an art and the complexity reached so far is still limited. But there are some strong tendencies among practitioners suggesting a set of design guidelines. Here are some examples of these guidelines:

*Guideline 1: Make behavior systems as specific as possible.*

One of the important lessons from classical AI research is the specificity-generality trade-off. More specific knowledge, i.e. knowledge more strongly tailored to the task and the domain, is more effective than generic mechanisms, such as general problem

solvers or universal representation schemes. Success in expert systems has depended almost entirely on the encoding of situation-specific knowledge. This trade-off also applies to behavior systems. Rather than trying to build a general purpose vision module for example, it is much more effective to tailor the sensing and actuating to a particular task, a particular domain, and a particular environment. Of course such a solution will not work outside its ‘niche’. But it will perform well and in a very cost effective way, as long as the conditions are appropriate. A good illustration of this design guideline is a visual navigation system developed by Horswill [49]. He has shown that by making a set of strong assumptions about the environment, the complexity of visual interpretation can be drastically reduced. One example is the detection of the vanishing point which in theory can be done by identifying edges, grouping them into line segments, intersecting the segments and clustering on the pairwise intersections. Horswill shows that each of these activities can be highly optimised. For example, although in general edge detection is complex and computationally intensive, a simple algorithm based on a gradient threshold will do, if the edges are strong and straight. This work goes in the direction of the theory of visual routines [122], which has abandoned the idea that there is a general purpose vision system and proposes instead a large collection of special purpose mechanisms which can be exploited in particular behavior systems.

Specialisation and the pressure to act in real time suggests a horizontal organisation, as opposed to a vertical or hierarchical organisation, typical for more classical ap-

proaches [21]. In a vertical organisation, the different modules perform specific functions like vision, learning, world representation, communication or planning. This leads to a sense-think-act cycle which does not guarantee real-time response when needed. In a horizontal organisation, every module combines all these functions but specialised and optimised with respect to a particular behavior in a particular environment. This is reminiscent of horizontal organisations now becoming more common in corporations [93].

*Guideline 2: Exploit the physics.*

Surprisingly, it is sometimes easier to achieve a particular behavior when the physics of the world, the morphology of the body, and the physics of the sensors and the actuators of the agent are properly exploited [21]. This is already the case for obstacle avoidance. A robot may be equipped with bumpers which cause a (sudden) slow down and an immediate retraction in a random direction. This may get the robot out of situations which appear to be dead end situations in simulations. Another good illustration of this design principle can be found in [129]. She has developed a model in the form of an artificial system for navigation based on the phonotaxis behavior of crickets. Webb points out that the determination of the direction in crickets is not based on intensity or phase differences, which would require complex neural processing, but on an extra tracheal tube which transfers vibration from one ear to the other. The length and characteristics of this tube are such that the indirectly arriving sound and the directly arriving sound interfere to give the final

intensity, which varies strongly with the direction of the sound. This is an example where “sensory mechanisms exploit the specificity of the task and the physics of their environment so as to greatly simplify the processing required to produce the right behaviour” ([129], 1093). Many more biological examples how physics may ‘solve’ problems, so that additional processing can be minimised, can be found in [3] and [127].

*Guideline 3: Do not think of sensing and acting in terms of symbol processing.*

The classical AI approach has been criticised because the symbols and symbol structures on which planning and decision making are based are not grounded in the real world [43]. The problem is that unequivocally decoding sensory data into a symbol and turning a command without error into its intended action may be unsolvable; not in principle but in practice. Behavior-oriented AI cannot escape the grounding problem. But a novel solution is proposed. Rather than trying hard to establish a better correspondence between symbols (like distance or turn with a given angle) and the physical properties of the robot in the environment, it is also possible to dispense altogether with the idea that a symbolic interpretation is necessary [107]. For example, rather than having a rule of the sort “if the distance is greater than  $n$ , then turn away at a certain angle  $a$ ”, a dynamical coupling between infrared reflection and path deflection, implemented for example as differences between left and right motor speed, can be set up. This coupling is designed without reference to concepts like ‘distance’ and ‘turn away’. It is therefore truly subsymbolic.

*Guideline 4. Simple mechanisms may give rise to complex behavior.*

Another strong tendency in the field is to make the mechanisms underlying a behavior system as simple as possible and to rely strongly on the interactions between different mechanisms and the environment to get the required behavior. This theme underlies other work in Artificial Life as well, and is related to the topic of emergence which is discussed more extensively in section 4 and 5. This tendency to search for simple mechanisms is particularly strong in the dislike of complex ‘objective’ world models [22]. The de-emphasis of complex representations is shared by researchers criticising cognitivism [125], and is related to the trend for situated cognition [115], which hypothesises that intelligence is the result of simple situation-specific agent/environment mechanisms that are strongly adapted to moment-to-moment decision making. Some biologists have called this the TODO principle: do whatever there is to do at a particular moment, instead of making complex representations and following elaborated plans [47].

It can be expected that many more design guidelines will become explicated as experience in building robotic agents continues. Some more extensive overviews can be found in [69], [22], [94], [66], a.o.



### **3.3 Different approaches are explored for designing the behavior programs.**

Although there seems to be a consensus in the field that behavior systems are appropriate units, different avenues are explored regarding the best way to design the underlying behavior programs. They fall roughly in four groups: neural network approaches, algorithmic approaches, circuit approaches, and dynamics approaches.

#### *Neural networks approaches*

Several researchers use artificial neural networks, in order to stay close to plausible biological structures ([5], [27], [94]). This approach is strongly related to biological cybernetics, and neuroethology [15]. A neural network consists of a set of nodes linked together in a network. Each node receives input from a set of nodes and sends activation as output to another set of nodes. Some inputs could come immediately from sensors. Some outputs are linked with actuators. The links between nodes are weighted. When the sum of the weighted inputs to a node exceeds a threshold, activation propagates to the output nodes. There are many variants of neural networks, depending on the type of propagation and the adaptation mechanism that is used for changing the weights [56]. Usually a single neural network (even with multiple layers) is not enough to build a complete robotic agent. More structure is needed in which different neural networks can be hierarchically combined. Several architectures and associated programming languages have been proposed. One of the best worked out examples is reported by Lyons and Arbib [62]. It centers around

the schema concept [7].

An advantage of neural network approaches is that they immediately incorporate a mechanism for learning. A disadvantage is that the global search space for an agent is too big to start from zero with neural network techniques. Much more initial structure must typically be encoded which is sometimes difficult to express in network terms.

### *Algorithmic approaches*

Other researchers have stayed closer to the methods traditionally used in computer programming so that powerful abstraction mechanisms can be used to cope with the complexity of programming complete robotic agents. One of the best known examples is the subsumption architecture [19]. The subsumption architecture makes two fundamental assumptions: (1) behavior programs are defined algorithmically, (2) there is a hierarchical but distributed control between different behavior systems based on subsumption relations.

The algorithmic descriptions in the subsumption architecture use a Turing-compatible formalism in the form of an augmented finite state machine (Fig. 3). An augmented finite state machine has a set of registers which can hold discrete values. On a robot, some of the registers hold the most recent value obtained from sensors. Others contain action parameters to be sent as fast as possible to the actuators. An augmented finite state machine also has a set of states in which the automaton can be. Operations consist of changing the contents of a register or mov-

ing to a new state. These operations can be controlled by first checking whether a condition on the state of the registers is true. An important feature of the finite state machines used by Brooks is access to a clock. This introduces an additional kind of operation: wait for a certain period of time and resume operation after that. It gives a handle on the difficult problems in fine-tuning the temporal aspects of behavior.

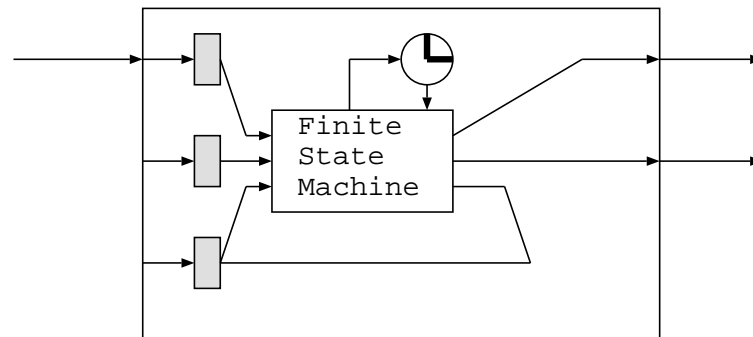


Figure 3: The augmented finite state automata have a set of internal registers, inputs and outputs, and a clock. The automaton cycles through a set of states.

In a single agent there will be a collection of behavior systems whose behavior programs are defined in terms of augmented finite state machines. The term subsumption refers to the way different behavior systems are made to operate together. It is assumed that in principle each behavior system is self-controlled, i.e. it is always active and moving through its different states conditioned by the incoming sensory signals. However one behavior system may inhibit that input values flow into the automaton or that action parameters are sent to the actuators. Inhibition is done by an explicit subsumption link which is under the control of the behavior system

(Fig. 4, Adapted from [19]).

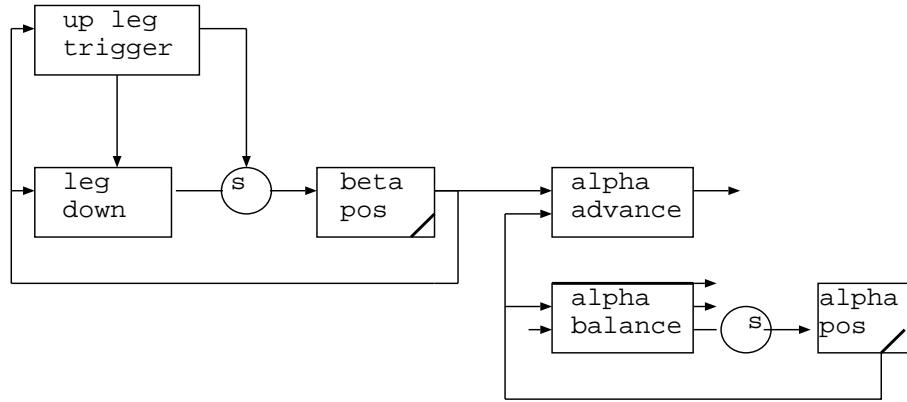


Figure 4: Partial network of finite state automata for the locomotion of a 6-legged robot. The boxes are state variables. Boxes with a line in the bottom right corner are finite state automata. Alpha balance is another network. Nodes marked s establish a subsumption relation. For example activation of ‘up leg trigger’ inhibits the inflow of ‘leg down’ to the ‘beta pos’ automaton.

In a concrete agent, the number and complexity of the finite state automata quickly grows to hundreds of states and registers. A higher level language, known as the behavior language [20], has therefore been designed to make the definition of large collections of behavior systems possible. Many of the low-level details of programming finite-state automata are thus removed and consequently more complex applications can be tackled. The behavior language and the subsumption architecture have been implemented on various computational platforms (mostly of the 68000 family) in different robotic hardware structures.

The recognized advantages of the subsumption architecture are as follows: (1) a universal computational formalism is used which gives a high degree of freedom and

expressability to the developer, (2) subsumption allows the modular build up of new competences by layering one behavior system on top of another.

Some of the disadvantages are: (1) algorithmic descriptions are more difficult to acquire or adapt (although see the work on genetic programming by Koza [58] discussed in section 5.), (2) an algorithmic specification makes it more difficult to get smooth behavior because conditions are expressed in terms of discrete thresholds, (3) the subsumption relation works well for basic sensori-motor competence, like 6-legged locomotion, but seems weak to regulate the interaction of more complex behavior systems which cannot be fined-tuned in advance.

### *Circuit approaches*

A third approach stays closer to electrical engineering by assuming that behavior programs, in order to be as efficient as possible, should take the form of combinatorial circuits [2], [100]. This approach potentially leads to direct hardware implementation using VLSI. A combinatorial circuit consists of a set of components which perform a transformation from inputs to outputs. The outputs of one component may be inputs to one or more other components, thus forming a network. Each component is very simple, performing Boolean operations, equality tests, etc. On an autonomous robot, the inputs would be connected to sensory signals and the outputs to action parameters. Signals propagate through the network thus relating sensing to action. A language (called REX) has been developed to describe circuits. Compilers and interpreters exist which allow REX-defined circuits to run on physical robots.

To make programming circuits more tractable, Rosenschein and Kaelbling [100] have developed a higher level language which is based on a logical formalism known as situated automata. A translator has also been developed that transforms expressions expressed in this logical formalism into circuits.

A circuit approach has a number of advantages from an engineering point of view. For example, performance can be predicted in terms of propagation steps needed. However, the circuit is completely fixed at run-time and it is less clear how continuous adaptation or the creation of new circuits can take place on-line.

*Dynamics approaches.*

Yet another approach is based on the hypothesis that behavior systems should be viewed as continuous dynamical systems instead of discrete computational systems as in the algorithmic approach. This dynamics approach has been put forward by a number of researchers (see e.g. [107], [114]). It is more in line with standard control theory, which is also based on dynamical systems [42]. Artificial neural networks are a special case of dynamical systems and can therefore be incorporated easily in this paradigm.

An example of a worked out dynamics architecture is described in [114]. It supports the formulation of processes and their combination in the design of complete behavior systems. Each process establishes a continuous relationship between a set of quantities. The quantities are either sensory signals, action parameters, or internal states. A process is always active. A collection of processes can be described in

terms of a set of differential equations. Because of the implementation on digital computers, the differential equations are turned into difference equations which can be directly implemented, similar to the way cellular automata are discretised versions of continuous systems [120]. Each process partially determines the change to a quantity enacted at the next time step, as a function of current values of the same or other quantities. At each computation cycle, all the changes are summed and the values of all the quantities take on their new values. The cycle time depends on the speed of the processor and the number of processes. There is no addressable global clock, as in the subsumption architecture. The complexity of the agent will therefore be bound by its computational power. When the cycle time becomes too slow, reactivity is no longer guaranteed.

A programming language PDL has been developed to make the implementation of behavior systems using this dynamics architecture more productive (Fig. 5). The PDL compiler links with the necessary low level software modules to handle sensory input and action parameter output. It maintains the different internal quantities and performs the basic cycle of determining all changes (by running the processes) and then summing and enacting the changes. PDL has been implemented on different PC-like hardware platforms for quite different robotic hardware structures.

```
void down_to_default_speed (void)
{
```

```

if (value(forward_speed) > 10)
    add_value(forward_speed,-1);
}

```

```

void up_to_default_speed (void)
{
    if (value(forward_speed) < 10)
        add_value(forward_speed,1);
}

```

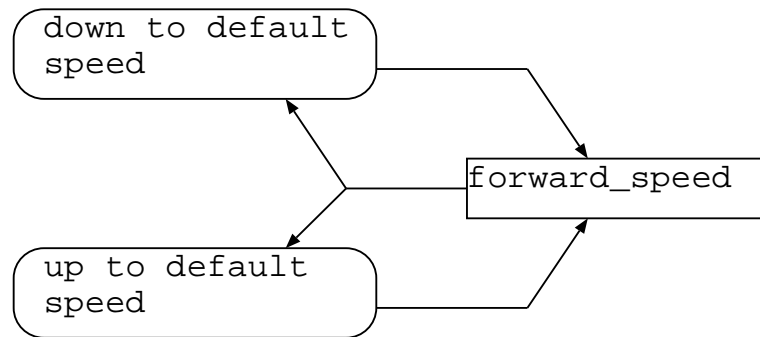


Figure 5: Process descriptions in PDL implementing a process network that will maintain the default forward speed at 10 by increasing or decreasing the speed in increments of 1.

A dynamics architecture approaches the problem of combining and coordinating different behavior systems differently from the subsumption architecture. Control is also distributed but one behavior system can no longer influence another one through a subsumption link. Instead, each behavior system is active at all times and the combined effect is added at the level of actions. For example, if one behavior system



influences the motors strongly to go left and the other one weakly to go right, then there will be a left tendency. The unsupervised combination of different behavior systems poses no special problems when they are orthogonal. It also poses no problem when temporal relations are implicitly present. For example, an infrared-based obstacle avoidance behavior system will necessarily become active before a touch-based obstacle avoidance behavior system because the infrared sensors will ‘see’ the obstacle earlier. No explicit control relations are therefore needed. When behavior systems are not orthogonal or not temporally ordered by the interaction dynamics, (partial) control of the actuators must take into account the fact that other behavior systems will have an impact at the same time. In these cases the interaction must be regulated by structural coupling [71] or co-adaptation: Behavior systems develop in the context of other behavior systems and their internal structure and functioning hence reflects this context. More complex control situations require the introduction of motivational variables which causally influence behavior systems and which have a dynamics on their own.

The advantages of a dynamics architecture are as follows: (1) The dynamical systems paradigm is closer to descriptions used in physics, biology, and control theory. This is an advantage because it makes it easier to carry over results from these fields (for example on adaptive processes). (2) Dynamic control leads in general to smoother behavior because it is not subject to sudden state changes due to discrete conditions. (3) Additive control does not enforce a layering. All behavior systems are at the same

level. In many cases, it is therefore easier to add behavioral competence than with a subsumption architecture. In some cases, it is more difficult because a structural coupling must be established.

Some of the disadvantages of a dynamics architecture are as follows: (1) Thinking in terms of dynamical systems instead of algorithms requires quite a shift from the viewpoint of developers who are used to algorithmic programming. Higher level abstractions still need to be developed. (2) The developer cannot explicitly control the timing of actions. This is an advantage because it removes one aspect of complexity. It is also a disadvantage because the problem of timing must be handled in different ways, for example by restructuring the behavior systems so that there is as little cascading as possible, or by decomposing behavioral competences in other ways.

There is still quite some work needed on additive control structures, particularly for hierarchical behavior systems, i.e. behavior systems that control a set of other behavior systems which are possibly internally temporally ordered. Work by Rosenblatt and Payton [99] and Tyrrell[?] shows the direction in which this is being explored.

These four different approaches to the design and implementation of behavior programs (neural networks, algorithms, circuits, dynamical systems) will undoubtedly be explored further in the near future, and new approaches may come up. In any case, all approaches need more high level abstractions to hide complexity and allow re-use

of large chunks from one experiment to another.

## 4 Emergent Behavior

Agents can become more complex in two ways. First, a designer (or more generally a designing agency) can identify a functionality that the agent needs to achieve, then investigate possible behaviors that could realise the functionality, and then introduce various mechanisms that sometimes give rise to the behavior. Second, existing behavior systems in interaction with each other and the environment can show side effects, in other words emergent behavior. This behavior may sometimes yield new useful capabilities for the agent, in which case we talk about *emergent functionality*. In engineering, increased complexity through side effects is usually regarded as negative and avoided, particularly in computer programming. But it seems that in nature this form of complexity buildup is preferred. Emergent functionality has disadvantages from an engineering point of view because it is less predictable and therefore appears less certain to a designer. Moreover the side effects are not always beneficial. But for an agent operating independently in the world, it has advantages because less intervention from a designing agency is needed. In fact, it seems the only way in which an agent can autonomously increase its capability. This is why emergent functionality has become one of the primary research themes in behavior-oriented AI. It is also the research theme that has the most connections to other areas of Artificial Life.

## **4.1 Emergence can be defined in terms of the need for new descriptive categories.**

Many researchers in the Alife community have attempted to define emergence (see for example [36], [59], [24], [111], [8]). For the present purposes, we will define emergence from two viewpoints: that of the observer and that of the components of the system.

From the viewpoint of an observer, we call a sequence of events a behavior if a certain regularity becomes apparent. This regularity is expressed in certain observational categories, e.g. speed, distance to walls, changes in energy level. A behavior is emergent if new categories are needed to describe this underlying regularity which are not needed to describe the behaviors (i.e. the regularities) generated by the underlying behavior systems on their own. This definition is compatible with the one used in chemistry and physics (see for example [88]). Thus the regularities observed in the collective behavior of many molecules requires new categories like temperature and pressure over and above those needed to describe the motion of individual molecules. Whether a behavior is emergent or not does not change according to this definition, with respect to who acts as observer, nor is it related to an element of unpredictability or surprise. Moreover it is not necessary that the two descriptions (the emergent behavior and the behavior of the individual components) are at different levels, although that is not excluded.

Emergence can also be defined from the viewpoint of the components implicated

in the emergent behavior [111]. We can make a distinction between controlled and uncontrolled variables. A controlled variable can be directly influenced by a system. For example, a robot can directly control its forward speed, although maybe not with full accuracy. An uncontrolled variable changes due to actions of the system but the system cannot directly impact it, only through a side effect of its actions. For example, a robot cannot directly impact its distance to the wall. It can only change its direction of movement which will then indirectly change the distance.

We can also make a distinction between a visible variable and an invisible variable. A visible variable is a characteristic of the environment which, through a sensor, has a causal impact on the internal structures and processes and thus on behavior. For example, a robot may have a sensor that measures distance directly. Distance would then be a visible variable for this robot. An invisible variable is a characteristic of the environment, which we as observers can measure but the system has no way to sense it, nor does it play a role in the components implicated in the emergent behavior. For example, the robot could just as well not have a sensor to measure distance.

For a behavior to be emergent, we expect at least that the regularity involves an uncontrolled variable. A stricter requirement is that the behavior (i.e. the regularity) involves only invisible variables. So, when a behavior is emergent, we should find that none of the components is directly sensitive to the regularities exhibited by the behavior and that no component is able to control its appearance directly.

A further distinction can be made between emergent behavior upon which the system does not build further, and semantic emergence [24] or second order emergence [9], in which the system is able to detect, amplify, and build upon emergent behavior. The latter can only happen by operating on the behavior programs which causally influence behavior, similar to the way genetic evolution operates on the genes. The remainder of this section discusses first order emergence. The next section (section 5) looks at semantic emergence.

## **4.2 The most basic form of emergent behavior is based on side effects.**

The first type of first order emergence occurs as a side effect when behavior systems are made to operate together in a particular environment (Fig. 6).

Consider the task of wall following. The behavioral regularity needed for this task is to have a bounded distance between the agent and the wall. This regularity can be achieved in a directly controlled, non-emergent way, by measuring the distance and using feedback control to steer away or towards the wall. Note that in this case the distance is required to describe the behavior causing wall following and that distance is a visible variable.

Maintaining a distance from the wall can be achieved in an emergent way by the simultaneous operation of two behavior systems (as demonstrated by Nehmzow and Smithers [85]). The first one achieves regular obstacle avoidance, for example in

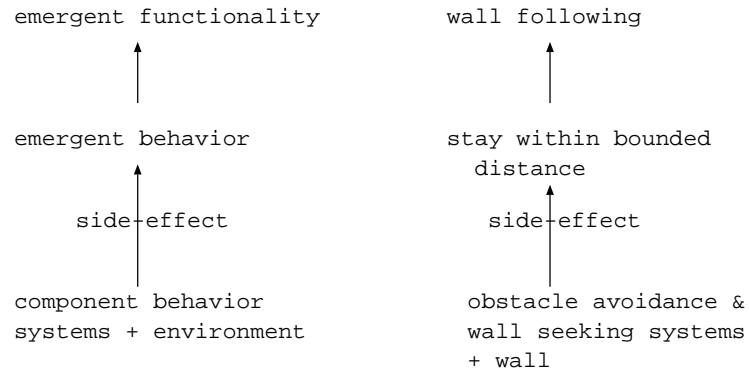


Figure 6: Left. Emergent behavior occurs as a side-effect of the interaction between behaviors and the environment. New descriptive categories are needed to describe it. Right. Example for wall following resulting from the operation of the obstacle and wall seeking behaviors.

terms of a dynamic coupling between infrared reflection and deflection of the path as described earlier. The second behavior system exhibits wall seeking. This behavior system maintains an internal variable  $c$  which reflects ‘the motivation of making contact with the left wall’. The variable  $c$  decreases to 0 when contact is made with the left wall (sensed by infrared reflection) and moves up otherwise. It influences the deflection of the forward motion path towards the wall. The higher  $c$ , the stronger the deflection. The two behavior systems together implement an attraction and repulsion behavior which added up and in the presence of a (left) wall gives the desired (left) wall following behavior (Fig. 7). An analogous behavior system is needed for making contact with a right wall.

Wall following is emergent in this case because the category ‘equidistance to the

(left/right) wall' is not explicitly sensed by the robot nor causally used in one of the controlling behavior systems.

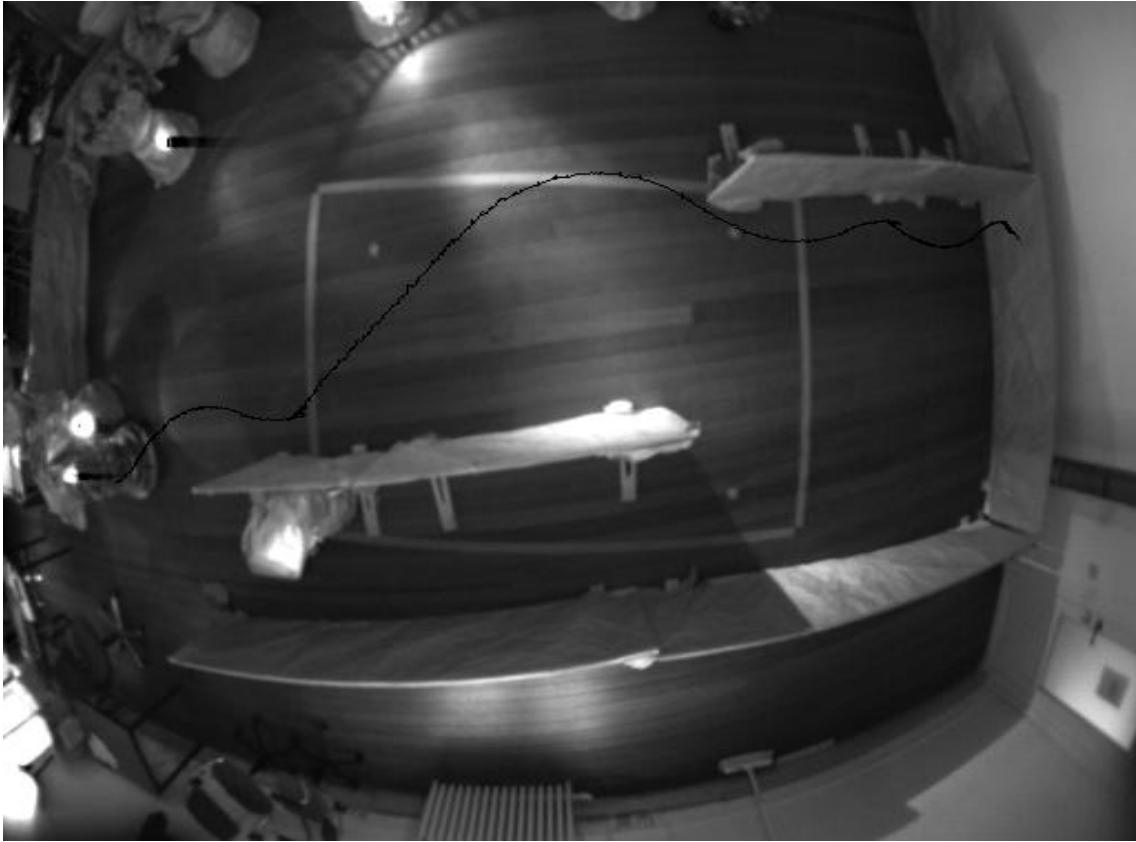


Figure 7: Emergent wall following implemented by an obstacle avoidance and a wall seeking behavior system interacting together within a particular environment. The image is taken from the ceiling and shows the robot arena. The path of the robot is automatically detected through a camera.

Emergent behavior has two advantages compared to directly programmed behavior. (1) No additional structure is needed inside the agent to get additional capabilities. We therefore do not need any special explanations how the behavior may come about. (2) Emergent behavior tends to be more robust because it is less dependent



on accurate sensing or action and because it makes less environmental assumptions. For example, the wall following mechanism described above continues to work even if the robot is momentarily pushed aside, if the wall is interrupted, or if the wall has a strong curvature. Emergent behavior usually has also disadvantages, for example, it is typically less efficient.

Here is a second example of emergent behavior. Suppose we want an agent that is able to position itself accurately between two poles which are part of a charging station. The charging station has an associated light source and the agent has two light sensitive sensors. The agent starts with two behavior systems: one based on phototaxis resulting in a zig-zag behavior towards the light source (and therefore the charging station), and one achieving obstacle avoidance by retracting and turning away when sensing an obstacle.

Because the agent may approach the charging station from any direction, it might seem that an additional positioning behavior is required, which makes sure that the agent enters the charging station between the two poles. However, a positioning behavior system is not necessary. The obstacle avoidance behavior causes retraction and turning away when the poles are hit. Because the robot is still attracted by the light source, it will again approach the charging station but now from a new angle. After a few trials, the robot enters the charging station as desired. The positioning behavior is emergent because position relative to the poles of the charging station is irrelevant to describe the behavior of the implicated behavior systems (obstacle

avoidance and phototaxis). There is no separate structure in the agent that is measuring position with respect to the poles and causally influences motion based on this measurement. The positioning behavior nevertheless occurs reliably without any additional structure in the agent (Fig. 8).

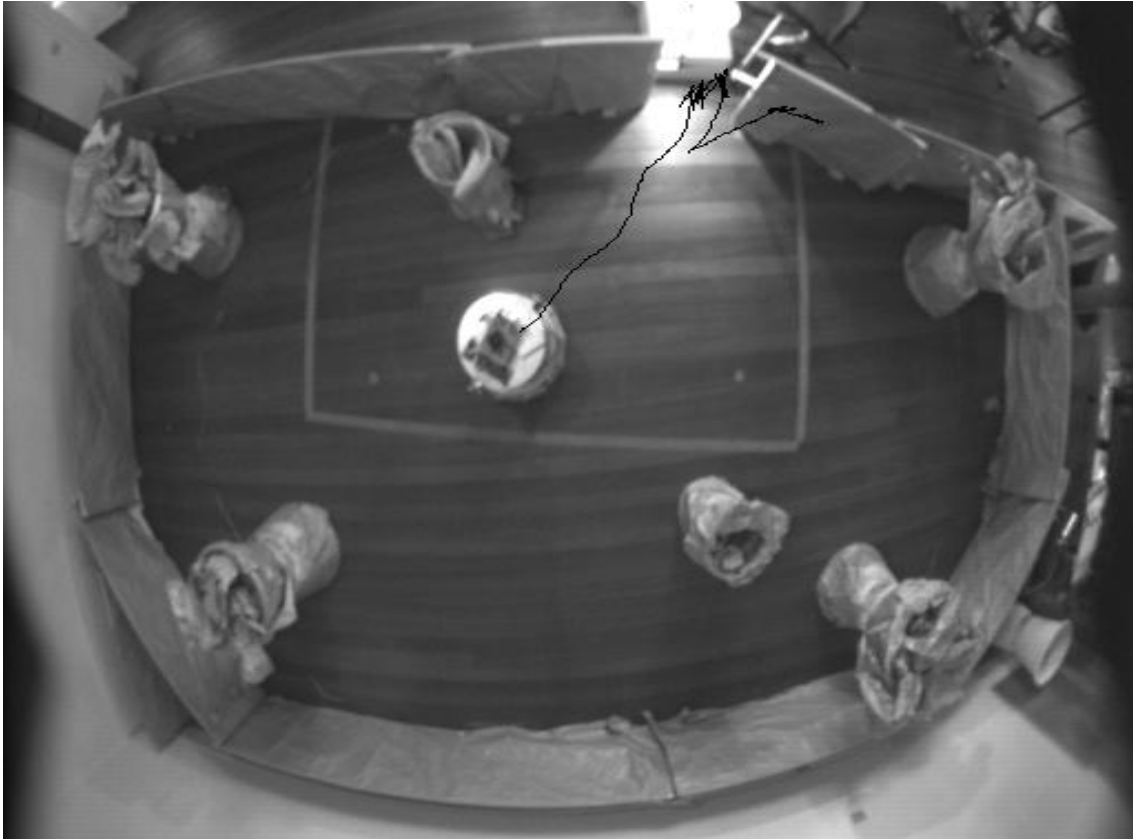


Figure 8: Zig-zag behavior towards light source and positioning behavior between two poles of a charging station located at the middle top of the figure. The positioning behavior is achieved in an emergent way by the interaction of two behavior systems. One homing in on the light source through phototaxis and the other performing touch-based obstacle avoidance.

### 4.3 A second form of emergent behavior is based on spatio-temporal structures

A second case of (first order) emergence is based on temporary spatio-temporal structures (Fig. 9). These structures themselves emerge as a side effect of interactions between certain actions of the agent and the environment. Local properties of the temporary structure in turn causally influence the observed behavior. The temporary structure is also emergent in the same sense as before, i.e. new descriptive categories are needed to identify the structure. These categories are not needed to describe the behavior of the underlying behavior systems that are causing the structure to appear nor are they sensitive to the structure as a whole. Also the behavior that results from making use of the structure is emergent because new descriptive categories are required that play no causal role in the underlying behavior systems.

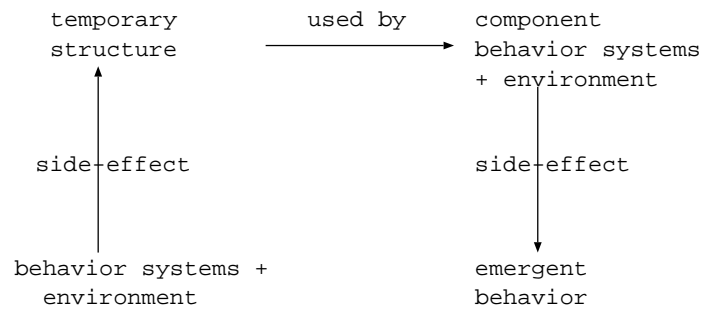


Figure 9: A second type of emergence is based on the formation of a emergent temporary structure which is then exploited by other behavior systems to establish the new emergent behavior.

This phenomenon is most easily observed in multi-agent systems but can also be used for establishing behaviors of a single agent. The classical example for multi-agent systems is the formation of paths. It has been well studied empirically not only in ant societies [91] but also in many other biological multi-element systems [10]. It is also well understood theoretically in terms of the more general theory of self-organisation [89]. The phenomenon has been shown in simulation studies [108], [29], [32] and recently on physical robots [14].

The temporary structure in the case of path formation in ant societies is a chemical pheromone gradient deposited in the environment. Ants are attracted to the pheromone and therefore have a tendency to aggregate along the path. Ants deposit the pheromone as they are carrying food back to the nest and are therefore responsible for the pheromone gradient in the first place. The pheromone dissipates so that it will disappear gradually when the food source is depleted. This emergent temporary structure is the basis of a derived emergent behavior, namely the formation of a path, defined as a regular spatial relation among the ants (Fig. 10). The path, as a global structure, is emergent because it is not needed to describe the behavior of the individual agents. Neither do any of the agents recognize the fact that there is a path. The agents operate uniquely on local information of the pheromone gradient. Only the observer sees the global path. The efficient transport of food benefits the multi-agent system as a whole and thus contributes to its self-preservation.

A difference with the examples discussed in the previous paragraph is that the

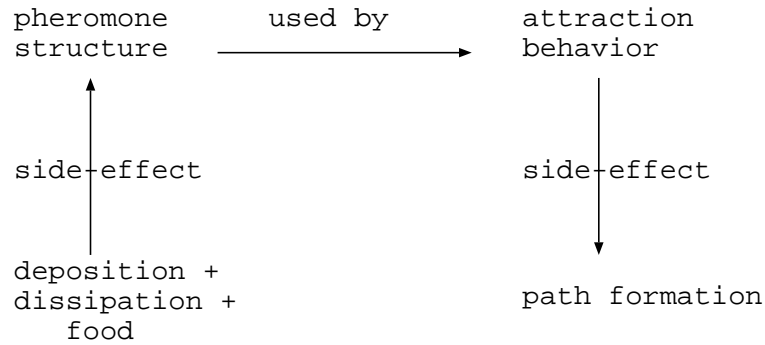


Figure 10: Path formation in ant societies is a classical example of emergent behavior due to the formation of a temporary structure. The structure in this case is a chemical pheromone gradient to which the ants are attracted.

emergent temporary structure sustains itself: As more ants are attracted to the pheromone concentration, there is a higher chance that they will carry back food and therefore deposit more pheromone. This increases the concentration of pheromone which will attract even more ants, and so on. So there are three forces in the system: build up of the path (by depositing pheromone), break down (by dissipation), and autocatalysis (through the chance of increased build-up) (Fig. 11). These forces are recognised as the essential ingredients for emergent temporary structures in general [111],[59].

Emergent temporary structures have also been used in individual agents. For example, several researchers have explored the creation of gradient fields over analogical representations of the environment. The best known example are potential fields [7], [6]. A potential field is a dynamical temporary structure created over an analogical representation of the environment by various repulsion and attraction forces.

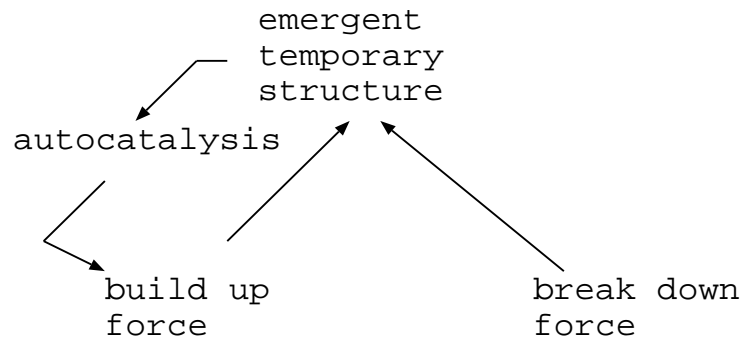


Figure 11: Emergent phenomena usually involve a force that builds up the phenomenon, a force that breaks it down, and an autocatalytic process so that the temporary structure builds upon itself.

The attraction force may come from the location of the desired goal towards which the agent wants to move. Repulsion may be generated by processes that are linked to the sensing of obstacles. Locomotion is influenced by the combined impact of attraction and repulsion forces (Fig. 12 from [7], p.99).

Other types of dynamics have been explored to generate and maintain emergent temporary structures to aid in navigation, for example fluid mechanics so that a fluid flow between the agent's location in an analogical map and the goal location emerges [28], or reaction-diffusion dynamics to generate concentration gradients that can be exploited in navigation or movement control [109].

The creation of temporary structures through a self-organising mechanism that combines build-up, break-down, and feedback giving rise to autocatalysis, has been used also for other aspects of intelligent behavior. For example, Maes [64] describes an action selection system (which maybe should be better called a motivational system)

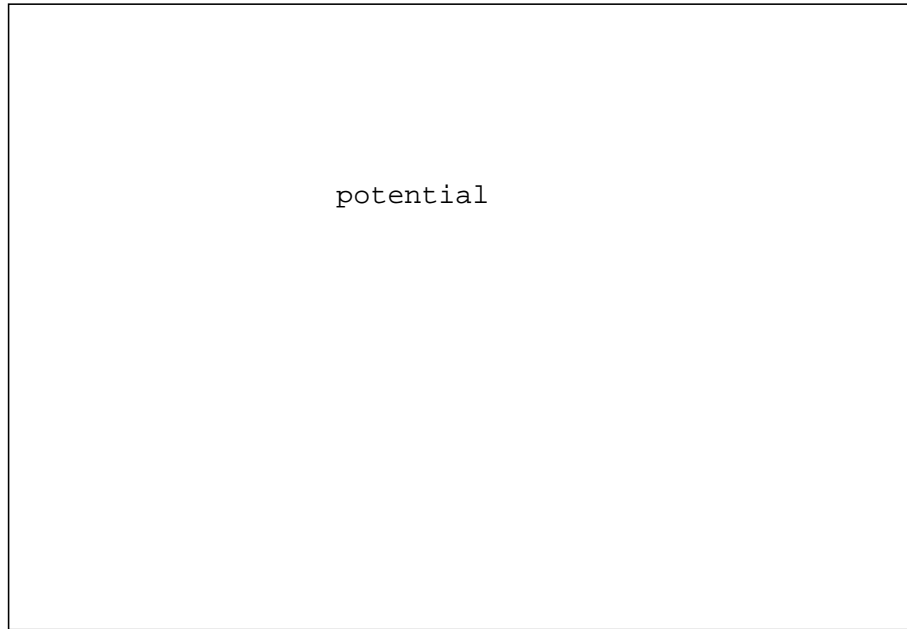


Figure 12: A potential field is a temporary structure created over an analogical representation of the world. The structure consists of vector fields which can either attract or repel robot movement. The sum of all the fields generates a path which the robot can follow. The example shows repulsion from two obstacles and a left and right wall.

in which the strength of a motivation is subject to positive enforcement (for example when the conditions for its satisfaction are sensed to hold) or negative enforcement (for example if contradicting motivations are active). These two processes generate a temporal ordering of the strength of motivations and consequently between the strength with which an action should get priority in execution. There is also a feedback mechanism: as motivation builds up, it will be able to inhibit competitors more effectively and thus gain additional strength. The temporary strength differences can be used by a decision module to determine which action will be selected next.

A particularly fascinating application of this mechanism for modeling spinal reflex behaviors of the frog is reported by Giszter[39]. The behaviors include limb withdrawal, aversive turns, and wiping away of noiceptive stimuli. The latter requires for example different simpler component behaviors: optional flexion, then place motion and then whisk motion. Each of these has certain conditions that need to be satisfied and each will make certain conditions true. If one behavior needs to be executed (for example ‘place motion’) it will pass activation along its predecessor link to ‘optional flexion’ thus raising its level of activation. When flexion executes it will establish conditions that make ‘place motion’ executable, and so on.

Another example of the creation of temporary emergent structures for a frame recognition system is reported in [110]. Each frame has a particular strength which corresponds to the applicability of the frame in a particular situation. There is an activation/inhibition dynamics and autocatalytic processes which create a temporal ordering on the frames so that the most appropriate frame for the given situation progressively gets the highest strength.

## **5 Emergent functionality**

The examples in the previous section showed that complexity may arise as a side effect of the operation of simpler mechanisms, but they do not indicate how there could be a progressive build up of more complexity. The only way this can happen is by the formation of new behavior systems. There is so far very little progress



in this area and new ideas are needed. Lack of progress comes partly from the practical difficulties in working with real physical agents, but these difficulties will progressively be alleviated as the technology matures. The real challenge is to find mechanisms which do not strain the limited resources of the agent and let the agent remain viable in the environment as it builds up more complexity.

## **5.1 There are severe difficulties in using existing artificial neural network techniques or evolutionary methods.**

At first sight, it may seem that mechanisms developed in artificial neural network research or genetic algorithms can be applied in a straightforward manner to the development of new functionality in autonomous agents. This is however not the case.

Let us first look at supervised learning, i.e. learning with the aid of examples or counterexamples. One of the best known supervised learning algorithms is back propagation [101]. Behavior programs could be represented as artificial neural networks associating sensory signals to actuator outputs. Changes in behavior programs could then be based on the error between the desired outcome and the outcome derived from using the association. For example, if  $X$  is a sensory signal,  $Y$  an action, and  $w$  the weight with which  $X$  influences the action, then an error would be the difference between the expected action and the action produced by  $Y = wX$ . There exist methods for adapting the weights  $w$ , which will lead to convergence [98]. Con-

vergence means that, given a consistent set of sense-act pairs, the learning method will settle on a stable set of weights that ‘correctly’ relates X with Y. It can be shown that certain functions (such as the XOR function) require a multi-layered network [83]. The weights in multi-layered networks can still be learned if the error is back-propagated through nodes at successive layers based on the relative contribution of each node to the derived outcome [101]. See [56] (chap 5) for a review of these and other supervised learning methods.

Although supervised learning methods have been demonstrated to be successful in simulation experiments, their application to autonomous agents runs into several problems. The first difficulty is that the methods require an adequate computation of the error and therefore a good enough prediction of what the actual outcome should be. A robot that is crashing into the wall gets feedback that there was a control error but cannot necessarily compute what would have been the right control actions to avoid the crash. Supervised learning methods therefore require a teacher which is more intelligent than the agent. But this is in contradiction with the objective of understanding how complexity might have arisen in the first place. A second difficulty is that the dynamics of weight adaptation requires a large amount of resources. The learning time grows very rapidly with the complexity of the network and an unrealistically high number of presentations of the correct sense-act pattern is typically required ([131], p. 87). A third difficulty is that not all networks will learn. If the network is too complex (too many layers or too many nodes) or too simple, it

will not be able to generalize. Too many presentations may degrade performance. Moreover the input and output representation must be carefully chosen, increasing the role of the designing agency ([131], 87). These difficulties explain why no one has as yet been able to convincingly use supervised learning methods on autonomous physical robots.

Another major neural network mechanism is known as reinforcement learning [118]. Reinforcement learning methods increase (and decrease) the probability that a particular association between sensing and acting will be used, based on a reward or reinforcement signal. The reinforcement signal is produced as a direct or indirect consequence of the use of the association. Many different associations may play a role in a particular behavior, and there may be a delay between a behavior and its (positive or negative) consequences. This introduces a credit assignment problem [82]. Early proposals ranked the possible situation-action associations, selected the best one (possibly with some variation to avoid local minima), and increased or decreased the probability of future choice depending on the effect of the chosen action ([132], [12]). More recent mechanisms go in the direction of having the agent develop a more sophisticated representation of the result of an action. For example, a prediction of reward is introduced, or a prediction of (long-term) cumulative reward, i.e. return [118]. A technique useful for learning temporal chains is to hand out reinforcement to the last action and from there back to previous associations which played a role. This technique is known as the bucket brigade algorithm and

originally due to [46].

Reinforcement learning methods have been shown to be capable of impressive learning behavior in simulations or engineering contexts [81], but there are again serious difficulties in the application to physical autonomous agents. The first major difficulty lies in the determination of the reinforcement signal. It is unrealistic to assume that the agent gets a clear scalar reinforcement signal after each action or series of actions. The second difficulty is that reinforcement learning assumes a trial-and-error search to find a viable association. Unless the agent is already close to the desired behavior, it may take quite a while before such an association is discovered [52]. The third difficulty is the credit assignment problem. Proposed solutions all go in the direction of new complexity (in the form of models of return, or in more recent cases world models predicting return [118], [61]). Often many simplifying assumptions are made about the nature of sensory interpretations or actions. For example, most methods assume that it is possible to select each time the ‘best’ action. But agents always execute several actions at the same time and in many cases actions (like turn left) are abstractions from the viewpoint of the designer which do not correspond to explicit commands in the robot, particularly not in dynamics architectures. Despite these difficulties, there are some preliminary experiments on physical mobile robots ([67], [52]). The general conclusion seems to be that “current reinforcement-learning algorithms can be made to work robustly on simple problems, but there are a variety of dimensions in which they must be improved before it will

be possible to construct artificial agents that adapt to complex domains” ([52], p. 46).

Supervised learning or reinforcement learning are both constructivist techniques: they modify weights based on minimising the error or on reinforcement. The alternative is known as selectionism: a complete behavior system is generated, for example by mutation or recombination based on existing behavior systems, and then tested as a whole. This mechanism is similar to evolution by natural selection as operating on the genes.

Evolutionary development has been shown in other areas of Artificial Life to be an extremely powerful source for generating more complexity (see e.g. [96]). It has also been proposed by some neurobiologists to be the major mechanism underlying the formation of new structure (and therefore functionality) in the brain [33], [25]. Evolutionary algorithms have been worked out in great detail and studied from a mathematical point of view (see the review in ([11]). The major variants are genetic algorithms [40] usually operating on classifier systems [45] and evolution strategies [103]. Applications have focused mostly on parameter optimization [63]. More recently, higher level descriptions as opposed to bitstrings have been used for the representation of the algorithm that needs to be derived and as a result more complex algorithms have been generated [58].

Evolutionary techniques start from a population of individuals (which in the present case would be equal to behavior systems) which each derive a different solution in

the space of possible solutions. The population is initialized in an arbitrary fashion. There is a fitness function which is defined over the space of all individuals. Individuals with higher fitness reproduce more often and thus the distribution of individuals of a certain type in the population changes. Reproduction means that copies are made, possibly after mutation (which introduces a random change), or recombination (which combines parts of two algorithms). Because mutation and recombination may potentially result in a better algorithm and because this algorithm will then be further reinforced by the selection step, the overall process evolves towards better and better regions of the search space.

Although this technique has resulted in very impressive results in an engineering context, the application to the development of autonomous agents poses some serious difficulties. The first problem is that genetic evolution requires quite a number of computational resources. The different individuals in the population need to be represented in memory and a large number of cycles is required to arrive at working, let alone optimal, solutions. This is a problem for a robot which has to remain viable, and thus maintain real-time responses within limited resource constraints. Consequently most researchers so far follow an off-line approach ([26], [57]). The genetic algorithm runs on a computer external to the robot. When a valid solution is found, it is loaded and integrated in the other behavior systems. Thus Koza [57] has shown how to derive the behavior programs for wall following and obstacle avoidance which were earlier demonstrated to function on a real robot programmed

in the subsumption architecture [70]. The primitive building blocks of the behavior programs are in this case the sensory inputs and action parameter outputs, Boolean connectives, conditionals, and the subsumption primitives. Brooks [23] has however criticised these results, mostly because the primitive building blocks were well chosen (based on an analysis of a known solution) and simplifying assumptions were made concerning the Boolean nature of certain conditionals.

Off-line evolution creates a new problem, which is the gap between the virtual world of the simulator and the real world. Koza [57] uses a very simple virtual world. Cliff, Husbands and Harvey [26] use a much more sophisticated simulation to test out the fitness of a solution. But, as Brooks [23] points out, the gap between simulated and real world will always remain quite large. One possible way out is to use the real robot as soon as reasonable solutions have been discovered. An example application of this technique is discussed in [105]. The application concerns the optimisation of path planning. Each robot is assumed to have a (static) map of the world that contains the obstacles and the goal towards which the robot needs to navigate. The genetic algorithm is used to search for a path towards the goal. The path is then executed and its quality evaluated with respect to effective use. Based on this new evaluation and additional information derived from the execution of the path, a new path is derived again using genetic techniques. The obvious problem with this approach is that only few solutions can be tried which diminishes the chances that a good solution is found in a genetic way.

There is another yet more fundamental problem with current evolutionary techniques which is the definition of the fitness function. The search towards a solution critically depends on the prior definition of this fitness function. But this introduces an important role for the designer. In the context of emergent functionality, we expect that the fitness function should be subject to evolution and should be local to the organism that evolves (as is indeed the case in [96]). Cariani [24] calls this pragmatic emergence.

## **5.2 A selectionist approach may be the key for generating emergent functionality.**

Although convincing examples of emergent functionality on physical robots operating in the real world do not exist, we are beginning to see the glimpses of it and breakthroughs can be expected soon. These examples build further on the techniques discussed in the previous paragraphs but combine them in a novel way.

When we study synthetic examples of emerging complexity, like that of Ray [96], we see that they are based on selectionist mechanisms and that they have in addition two crucial features:

1. There is enough initial complexity to make a viable organism and there are many diverse organisms. The build up of complexity is as much due to the competitive interaction between organisms as to their interaction with the world.
2. The ecological pressures on organisms are real and come partly from other organ-



isms. In other words, there are no predefined or static fitness functions or rewards, as assumed in genetic algorithms and reinforcement learning. There is no teacher around as assumed in supervised learning.

To make selectionism work for robots, it seems therefore appropriate to draw a parallel between organisms and behavior systems. This means that we should not concentrate on the acquisition of a single behavior system (e.g. for locomotion or obstacle avoidance), but that there should be many diverse behavior systems which are complementary but still in competition. Paradoxically it might be easier to develop many behavior systems at once than to concentrate on one behavior system in isolation.

Each behavior system, except for the basic reflexes, should remain adaptive, just as each individual organism remains adaptive (within the bounds of the genotype). New behavior systems begin their life by accessing the same or other visible variables and the same or other controlled variables. The new behavior system monitors the situation and adapts itself so as to have a similar impact on the controlled variables as the base line behavior systems, but using other sensory modalities. The important point here is that the generation of the new behavior system does not take place by trial and error.

Most of the time, behavior systems will have a different functionality. If there are behavior systems with the same functionality (for example obstacle avoidance), the diversity should not lie in variants of the same approach to a solution, as in the

case of genetic algorithms, but in fundamental differences in how the functionality is approached (for example obstacle avoidance using touch-based reactive reflexes versus obstacle avoidance using infra-red based classification).

We should also introduce real environmental pressures, such as limited internal energy availability, and real-time or memory constraints, in addition to real environmental pressures such as limited external energy availability, avoiding of self-damage, etc. These pressures should feedback on the formation or adaptation of behavior systems. A behavior system is less competitive if the sensory patterns to which the behavior system responds do not occur (for example its thresholds are too high), if the time to make a decision on how to influence actuation is too long so that the conditions for activation are no longer satisfied, if other behavior systems always override the influence on actuation, if many memory resources are needed, etc. There may also be behavior systems that specialise in monitoring internal and external environmental conditions and act as a ‘reaper’ [96], weakening or eliminating other behavior systems. An example of this is already shown in [86].

Large-scale experiments incorporating this approach do not exist yet. But reason for the optimism that emergent functionality may be demonstrated soon, comes from some initial experiments which show how new behavior systems may bootstrap themselves in the context of other behavior systems. Let us look at one concrete example in the context of obstacle avoidance. This example was first suggested and tested in simulation by Pfeifer and Verschuere [94]. We have since done similar

experiments on a real robot and with different sensory modalities in Brussels.

The base line behavior systems are:

- Maintain a default speed in a forward direction.
- Maintain a forward direction.
- Reverse speed if touching an obstacle in the front.
- Turn away left if touched on the right side.
- Turn away right if touched on the left side.

Following a dynamics viewpoint, each of these behavior systems establishes a continuous relationship between sensory signals and action parameters. For example, a positive default speed is maintained by increasing it, if it is below the default, or decreasing it, if it is above. Reversing the speed is done by a sudden decrease of the speed if a touch sensor is active. The positive default speed is then automatically restored by the ‘maintain a positive default speed’ system.

The two emergent behavior systems (one that will do obstacle avoidance for obstacles on the left and another one that will do the same for obstacles on the right) are sensitive to the infrared sensors and they impact the rotational motor. They use associative or Hebbian learning. In Hebbian learning, an association between two elements (for example sensors and actuators) is made stronger based on co-occurrence [44]. It is also made weaker, for example due to a constant forgetting rate. Associative learning has been extensively studied in the artificial neural network field

(reviewed in [56], chap 4)). In the present case, there will be a progressively stronger association between particular states of the infrared sensors (determined by the environment) and particular action parameters of the rotational motor (determined by the turn away left and turn away right behavior systems). We see thus a form of classical conditioning with the touch sensors as the unconditioned stimulus, and the infrared as the conditioned stimulus.

To get emergent functionality, an additional step is needed. The new behavior systems so far perform the same activity as the base line behavior systems and are therefore not yet competitive. A behavior system becomes competitive if it causes a qualitatively different action which has an additional advantage for the agent. This can happen in many ways. For example, the new behavior systems could involve only some of the controlled variables so that some actions no longer take place, giving an overall qualitatively different behavior, or a behavior system may sense more quickly the upcoming situation and thus influence the action before the already existing behavior systems.

In the present case, the infrared based obstacle avoidance system can become more competitive because the infrared sensors have a further range than the touch sensors. They can therefore react more quickly to the presence of obstacles. Due to the progressive strengthening of the association, there will be a particular point in time in which the infrared based behavior systems react earlier than the touch-based ones. This is the point where the newly emergent functionality becomes visible.

Without infra-red based obstacle avoidance, a reversing of speed took place so that the robot is backing up while turning away. This reversal of speed is no longer present when infrared based obstacle avoidance is strong enough because the agent no longer touches the obstacles. Instead, we observe a deviation away from obstacles (Fig. 13). This deviation is from the viewpoint of energy usage more beneficial to the agent.

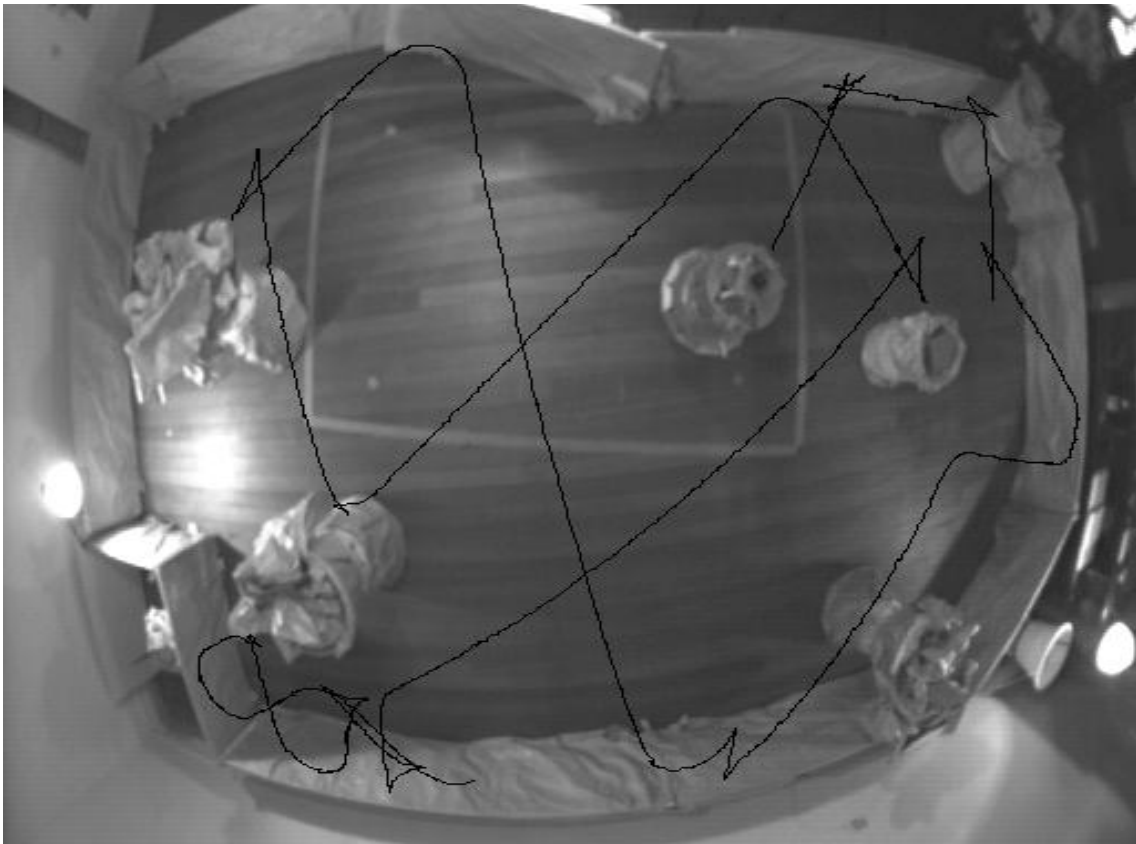


Figure 13: When the robot starts it bumps into obstacles and retracts based on a built-in touch-based obstacle avoidance behavior system. Progressively it will associate infrared signatures with rotational movement and no longer bump into obstacles.

The associative learning mechanism has an autocatalytic element because the trig-

gering due to infrared itself also enforces the association. Thus the association strengths in the new behavior systems feed on themselves and become progressively stronger (Fig. 14).

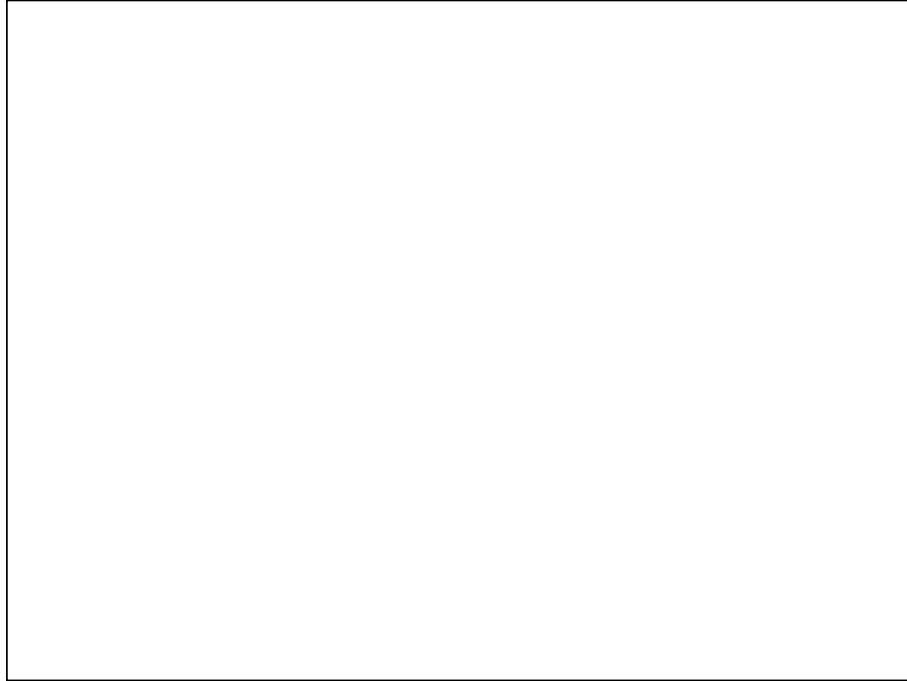


Figure 14: Evolution of the weights based on Hebbian learning. The weights determine the impact of the infrared sensors on the rotational motor. The increase feeds on itself. The decrease due to constant forgetting is also clearly visible.

This is indeed an example of emergence, according to the earlier definitions. Different sensory modalities are used compared to the original behavior systems, and there is also a qualitatively different behavior which is more beneficial to the agent. The example illustrates that emergent functionality is not due to one single mechanism but to a variety of factors, some of them related to internal structures in the agent, some of them related to the properties of certain sensors and actuators, and some

of them related to the interaction dynamics with the environment.

Because the formation of the new behavior system here happens in the context of other behavior systems, the agent always remains viable. For example, if the new infrared based obstacle avoidance behavior systems fail, the touch-based solution is still there and will immediately become active. Because the formation is guided by existing behavior systems, it evolves quickly without trial-and-error or search. All the time the agent remains viable.

It is obvious that more work is needed to achieve emergent functionality in physical autonomous robots, but current work exploring neural network techniques and evolutionary algorithms gives a good starting point. Their integration into an overall selectionist approach where diverse behavior systems compete and co-adapt give reasons for optimism.

## **6 Conclusions.**

Behavior-oriented AI research has opened up an ‘artificial life route to artificial intelligence’. It has three aspects: First, the problem of intelligence is framed within the general context of biology. Intelligent behavior is defined as maximising the agent’s chances for self-preservation. Successful adaptation and learning is defined as changes in the internal structure of the agent which maximise intelligence. Second, the tasks being explored by behavior-oriented AI researchers are very different

from those considered in classical AI. They center around sensori-motor competence and the ability to operate autonomously in a dynamically changing environment. Third, the models take inspiration from the way intelligent behavior appears to be established in natural systems. It is hypothesised that the principles that underly the living are also the ones that give rise to intelligent behavior. Many researchers hope to contribute to biology by testing out in artificial models whether certain biological hypotheses are plausible.

Behavior-oriented AI research made a slow start around the mid-eighties, but is currently exploding. Many research laboratories have now acquired the competence to build their own robots and to perform experiments. Rapid experimental progress can therefore be expected in the coming five years, if only by pursuing further the research lines that have been briefly summarised in this paper. What is particularly needed are larger scale efforts: agents with many different behavioral competences operating in ecosystems with a multitude of challenges, multi-agent systems with a sufficient number and diversity of the agents, learning methods applied over sufficiently long periods of time to get non-trivial build up of complexity, and so on. At the same time, the experimental rigour needs to be increased so that quantitative performance measures can be applied.

Given the current state of the art and the rapid evolution in mechanical engineering and computer technology, we can also expect rapid technological progress, leading towards the first real world applications, possibly in the area of environmental



monitoring, space exploration, or microsystems. In general, it takes about 10 years before a technology becomes sufficiently accepted for serious real world applications. Major hurdles are not only technical. In this respect, the state of the art of behavior-oriented AI can be compared to that of knowledge engineering in the late sixties, when the shift towards situation-specific knowledge and rule-based formalisms was taking place. It took several decades to turn these results into a solid engineering methodology and develop a number of well established industrial achievements, like XCON for configuring computer installations [74].

*Some open issues.*

There are also many open problems beyond increasing the complexity of current systems. One of them, that has hardly been addressed, concerns the relation between the mechanisms used in behavior-oriented AI and those used in knowledge-oriented AI. Some researchers (on both sides) suggest that the other approach is irrelevant to reach human-level intelligence. They then have to prove that their methods will work all the way. Other researchers believe that the symbolic level exists as an independent level, which is causally influenced by and causally influences the dynamics level. No concrete proposals operational on physical autonomous robots exist today to allow a technical discussion of the subject, but one day the problem will have to be tackled.

Another question concerns adequate formalisation and theory formation. There is already a wide literature with formal results for the mechanisms discussed earlier

(error-driven learning, genetic evolution, etc.), but application to integrated physical agents operating in real world environments will require more work. Several researchers have proposed a state-space approach for defining the dynamics of the observed behavior and the internal operation of the agent (e.g., [77], [54], [37], [119]). Once a state-space description is available, the concepts of dynamical systems theory (attractors, transients, recurrent trajectories, etc.) [1] can be used to characterise qualitatively and quantitatively behaviors and internal structures like perceptions, representations, and actions. Within this framework concepts like emergent functionality can be formalised and the results of emergent functionality better understood. At the same time work must proceed on developing formal theories to characterise the challenges in ecosystems, the optimality of behavior, and thus the chances of self-preservation of the agent [76].

The field of behavior-oriented AI research shows enormous signs of vitality. This paper focused only on a few aspects, ignoring other topics such as multi-agent systems, communication and cooperation, the formation of new sensory modalities, and so on. There is no doubt that major contributions can be expected in the coming decade, particularly as the technical tools mature and more researchers get involved.

## **7 Acknowledgement**

The writing of this paper was partly sponsored by the ESPRIT basic research project SUBSYM and the IUAP Action of the Belgian Science Ministry. Comments from

Chris Langton, David McFarland, Walter Van de Velde and Anne Sjostrom have improved the paper. Continuous discussions with members of the VUB autonomous agents group and with Rodney Brooks, David McFarland, Rolf Pfeifer, and Tim Smithers have helped to shape the viewpoints expressed in this paper. I am strongly indebted to the hospitality of the Aamodt family. The writing of this paper started in the peaceful surroundings of their Norwegian home.

## 8 References

### References

- [1] Abraham, R.H. and C. D. Shaw (1992) *Dynamics. The Geometry of Behavior*. 2nd Edition. Addison-Wesley Pub. Cy., Reading, Ma.
- [2] Agre, P. and D. Chapman (1987) *Pengi: an implementation of a theory of activity*. Proceedings of the Sixth National Conference on Artificial Intelligence. Morgan Kaufmann. San Mateo, Ca. p. 268-272.
- [3] Alexander, R.M. (1968) *Animal Mechanics*. Sidgwick and Jackson, London.
- [4] *Alife* (1994) *Journal of Artificial Life*. MIT Press, Cambridge Ma.
- [5] Arbib, M.A. and A.R. Hanson (eds.) (1987) *Vision, Brain, and Cooperative Computation*. MIT Press/Bradford Books, Cambridge Ma.

- [6] Arbib, M. A. and D.H. House (1987) Depth and Detours: An essay on visually guided behavior. In: Arbib, M.A. and A.R. Hanson (eds.) (1987) Vision, Brain, and Cooperative Computation. MIT Press/Bradford Books, Cambridge Ma. p. 129-163.
- [7] Arkin, R. (1989) Motor Schema based mobile robot navigation. Int. Journal of Robotics Research. Vol 8, 4 p. 92-112.
- [8] Assad, A. and N. Packard (1991) Emergent Colonization in an Artificial Ecology. In: F.J. Varela and P. Bourgin (eds.) Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 143-152.
- [9] Baas, N. (1993) Second order emergence. Oral communication at the second European Conference on Artificial Life, ULB Brussels.
- [10] Babloyantz, A. (1986) Molecules, Dynamics, and Life. An Introduction to the Self-organisation of Matter. Wiley, New York.
- [11] Baeck, T., H-P. Schwefel (1993) An Overview of Evolutionary Algorithms for Parameter Optimization. Evolutionary Computation. Vol 1,1. p. 1-23.
- [12] Barto, A.G. and R.S. Sutton (1991) Landmark learning: An illustration of associative search. Biological Cybernetics Vol. 42. p. 1-8.

- [13] Barto, A.G. (1990) Connectionist Learning for Control. In Miller, T.W., R.S. Sutton, and P.J. Werbos (eds.) Neural Networks for Control. MIT Press/Bradford Books. Cambridge Ma. p. 5-58.
- [14] Beckers, R. (1993) Demonstration at the Alife Meeting, Technical University of Delft.
- [15] Beer, R.D. (1990) Intelligence as Adaptive Behavior: An Experiment in Computational Neuroethology. Academic Press, Cambridge Ma.
- [16] Bonner, J.T. (1988) The Evolution of Complexity by Means of Natural Selection. Princeton University Press, Princeton, N.J.
- [17] Brady, J.M. and R. Paul (1984) Robotics Research: The First International Symposium. MIT Press, Cambridge Ma.
- [18] Braitenberg, V. (1984) Vehicles: Experiments in Synthetic Psychology. MIT Press, Cambridge MA.
- [19] Brooks, R. (1986) A Robust Layered Control System for a Mobile Robot. IEEE Journal of Robotics and Automation. 2(1). p. 14-23.
- [20] Brooks, R. (1990) The Behavior Language; User's Guide. MIT AI Lab. Memo 1127.
- [21] Brooks, R. (1991) Intelligence without reason. Proceedings of IJCAI-91. Morgan Kaufmann, San Mateo Ca. p. 569-595.

- [22] Brooks, R. (1991b) Challenges for Complete Creature Architectures. In: Meyer, J-A., and S.W. Wilson (1991) From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books. Cambridge Ma. p. 434-443.
- [23] Brooks, R. (1992) Artificial life and real robots. In: F.J. Varela and P. Bourguine (eds.) Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 3-10.
- [24] Cariani, P. (1991) Emergence and Artificial Life. In: Langton, C.G., C. Taylor, J.D. Farmer, and S. Rasmussen (1992) Artificial Life II. Proceedings of the Workshop on Artificial Life Held February, 1990 in Santa Fe, New Mexico. Addison-Wesley, Reading Ma. p. 775-797.
- [25] Changeux, J-P. (1986) Neuronal Man: The Biology of Mind. Oxford University Press, Oxford.
- [26] Cliff, D., P. Husbands, and I. Harvey (1993) Evolving Visually Guided Robots. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 374-383.

- [27] Cruse, H., U. Muller-Wilm, and J. Dean (1993) Artificial Neural Nets for controlling a 6-legged walking system. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 52-60.
- [28] Decuyper, J. and D. Keymeulen (1991) A Reactive Robot Navigation System Based on a Fluid Dynamics Metaphor. In: Varela, F.J. and P. Bourgin (eds.) (1992) Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 348-355.
- [29] Deneubourg, J-L. and S. Goss (1990) Collective patterns and decision making. Ecology, Ethology and Evolution. Vol 1. p. 295-311.
- [30] Deneubourg, J-L, et.al. (1993) Self-organisation and life: from simple rules to global complexity. Proceedings of the Second European Conference on Artificial Life. ULB, Brussels.
- [31] Donnett, J. and T. Smithers (1990) Lego Vehicles: A Technology for studying intelligent systems. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 540-569.

- [32] Drogoul, A. and J. Ferber (1993) From Tom Thumb to the Dockers: Some Experiments with Foraging Robots. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 451-459.
- [33] Edelman, G. (1987) Neural Darwinism: The Theory of Neuronal Group Selection. Basic Books, New York.
- [34] Edelman, G. (1992) Bright Air, Brilliant Fire. On the Matter of the Mind. Basic Books, New York.
- [35] Flynn, A. and R. Brooks (1989) Building Robots: Expectations and Experiences. IEEE Workshop on Intelligent Robots and Systems. IROS '89. Tsukuba. p. 236-243.
- [36] Forrest, S. (1989) Emergent computation: self-organizing, collective, and cooperative phenomena in natural and artificial computing networks. North-Holland Pub. Co, Amsterdam.
- [37] Gallagher, J. and R. Beer (1993) A Qualitative Dynamical Analysis of Evolved Locomotion Controllers. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 71-80.



- [38] Genesereth, M. and N. Nilsson (1987) Logical Foundations of Artificial Intelligence. Morgan Kaufmann, Los Altos.
- [39] Giszter, S. (1993) Behavior networks and force fields for simulating spinal reflex behaviors of the frog. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 172-181.
- [40] Goldberg, D.E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesely, Reading Ma.
- [41] Goss, S. and J-L. Deneubourg (1992) Harvesting by a group of robots. In: Varela, F.J. and P. Bourgine (eds.) (1992) Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 195-204.
- [42] Hallam, J. (1993) Playing with toy cars. In: Steels, L. and R. Brooks (eds.) (1993) The 'artificial life' route to 'artificial intelligence'. Building situated embodied agents. Lawrence Erlbaum Associates, New Haven.
- [43] Harnad, S. (1990) The symbol grounding problem. *Physica D*, 42 (1-3). p. 335-346.
- [44] Hebb, D.O. (1949) *The Organization of Behaviour*. Wiley, New York.

- [45] Holland, J.H. (1975) *Adaptation in Natural and Artificial Systems*. The University of Michigan Press, Ann Arbor, Michigan.
- [46] Holland, J.H. (1985) Properties of the bucket brigade algorithm. In J. J. Grefenstette (ed.) *Proceedings of the First International Conference on Genetic Algorithms and their applications*. Lawrence Erlbaum, Pittsburgh, Pa. p. 1-7.
- [47] Hoogeweg, P. (1989) *Mirror Beyond Mirror: Puddles of Life*. In: Langton, C.G. (1989) *Artificial Life*. Santa Fe Institute Studies in the Sciences of Complexity. Proc. Vol VI. Addison-Wesley, Reading Ma. p. 297-316.
- [48] Horswill, I. (1992) *Characterizing Adaptation by Constraint*. In: Varela, F.J. and P. Bourguine (eds.) (1992) *Toward a Practice of Autonomous Systems*. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 58-63.
- [49] Horswill, I. (1993) A simple, cheap, and robust visual navigation system. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) *From Animals to Animats2*. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p.129-136.
- [50] Jones, J.L. and A.M. Flynn (1993) *Mobile Robots. Inspiration to implementation*. A.K. Peters, Wellesley Ma.

- [51] Kaelbling, L. and S. Rosenschein (1990) Action and planning in embedded agents. *Journal of Robotics and Autonomous Systems*, 6, 35-48.
- [52] Kaelbling, L. (1992) An Adaptable Mobile Robot. In: Varela, F.J. and P. Bourguine (eds.) (1992) *Toward a Practice of Autonomous Systems*. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 41-47.
- [53] Kauffman, S.A. (1993) *The origins of order: self organization and selection in evolution*. Oxford University Press, Oxford.
- [54] Kiss, G. (1991) Autonomous Agents, AI and Chaos Theory. In: Meyer, J-A., and S.W. Wilson (1991) *From Animals to Animats*. Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books. Cambridge Ma. p. 518-524.
- [55] Kohonen, T. (1988) *Self-Organization and Associative Memory*. Springer Series in Information Sciences. Vol 8. Springer Verlag, Berlin.
- [56] Kosko, B. (1992) *Neural Networks and Fuzzy Systems. A Dynamical Systems Approach to Machine Intelligence*. Prentice Hall, Englewood Cliffs, NJ.
- [57] Koza, J. (1991) Evolving Emergent Wall Following Robotic Behavior using the Genetic Programming Paradigm. In: Varela, F.J. and P. Bourguine (eds.) (1992) *Toward a Practice of Autonomous Systems*. Proceedings of the First

European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 110-119.

- [58] Koza, J. (1992) Genetic Programming. MIT Press, Cambridge Ma.
- [59] Langton, C.G. (1989) Artificial Life. Santa Fe Institute Studies in the Sciences of Complexity. Proc. Vol VI. Addison-Wesley, Reading Ma.
- [60] Langton, C.G., C. Taylor, J.D. Farmer, and S. Rasmussen (1992) Artificial Life II. Proceedings of the Workshop on Artificial Life Held February, 1990 in Santa Fe, New Mexico. Addison-Wesley, Reading Ma.
- [61] Lin, L-J, and T.M. Mitchell (1993) Reinforcement Learning with Hidden States. In: In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats<sup>2</sup>. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 271-280.
- [62] Lyons, D.M., and M.A. Arbib (1989) A Formal Model of Computation for Sensory-Based Robotics. IEEE Trans. on Robotics and Automation. 5. p. 280-293.
- [63] Maenner, R. and B. Manderick (1992) Parallel Problem Solving from Nature, 2. North-Holland Pub. Co, Amsterdam.

- [64] Maes, P. (1989) The Dynamics of Action Selection. In: Proceedings of the 11th International Joint Conference on AI (IJCAI 89) Morgan Kaufmann, Pub. Los Altos. p. 991-997.
- [65] Maes, P. (1990) Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back. MIT Press/Bradford Books, Cambridge Ma.
- [66] Maes, P. (1993) Behavior-Based Artificial Intelligence. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats<sup>2</sup>. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books. Cambridge Ma. p. 2-10.
- [67] Maes, P. and R. Brooks (1990) Learning to coordinate behaviors. Proceedings of the Eighth National Conference on Artificial Intelligence. Morgan Kaufmann, San Mateo, Ca. p. 796-802.
- [68] Malcolm, C. and T. Smithers (1988) Programming Assembly Robots in Terms of Task Achieving Behavioural Modules: First Experimental Results. In: Proceedings International Advanced Robotics Programme, Manchester. p. 15.1-15.6.
- [69] Malcolm, C.A., T. Smithers and J. Hallam (1989) An emerging paradigm in robot architecture. In: T. Kanade, F. Groen and L. Herzberger (eds.) (1989) Intelligent Autonomous Systems 2. Amsterdam. p. 546-564.

- [70] Mataric, M. (1990) Environment Learning Using A Distributed Representation. In: Proceedings of 1990 IEEE International Conference on Robotics and Automation. p. 402-406.
- [71] Maturana, H.R. and F.J. Varela (1987) The Tree of Knowledge: The Biological roots of Human Understanding. Shamhala Press, Boston.
- [72] Maynard-Smith, J. (1982) Evolution and the theory of games. Cambridge Univ. Press, Cambridge UK.
- [73] McClelland, J.L. and D.E. Rumelhart (eds.) (1986) Explorations in Parallel Distributed Processing. MIT Press/Bradford Books, Cambridge Ma.
- [74] McDermott, J. (1982) R1: A Rule-Based Configurator of Computer systems. In: AI Journal, Vol 19,1. p. 39-88.
- [75] McFarland, D. (1992) Animals as cost-based Robots. International Studies in the Philosophy of Science, Vol 6, 2. p. 133-153.
- [76] McFarland, D. and T. Boesser (1994) Intelligent Behavior in Animals and Robots. MIT Press/Bradford Books, Cambridge Ma.
- [77] McFarland, D. and A. Houston (1981) Quantitative Ethology: the state-space approach. Pitman Books, London.
- [78] Meyer, J-A. and A. Guillot (1991) Simulation of Adaptive Behavior in Animals: Review and Prospect. In: Meyer, J-A., and S.W. Wilson (1991) From

Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/ Bradford Books. Cambridge Ma. p. 2-14.

- [79] Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p.129-136.
- [80] Meyer, J-A., and S.W. Wilson (1991) From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/ Bradford Books. Cambridge Ma.
- [81] Miller, W.T., R.S. Sutton and P.J. Werbos (eds.) (1990) Neural Networks for Control. MIT Press/Bradford Books. MIT Press, Cambridge, Ma
- [82] Minsky, M. (1961) Steps towards artificial intelligence. In: Feigenbaum, E. and J. Feldman (eds.) Computers and Thought. McGraw-Hill, New York. p. 406-450.
- [83] Minsky, M. and S. Papert (1988) Perceptrons. MIT Press, Cambridge Ma.
- [84] Minsky, M. (1985) The Society of Mind. Simon and Schuster, New York.
- [85] Nehmzow, U., T. Smithers and B. McGonigle (1993) Increasing Behavioural Repertoire in a Mobile Robot. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Sec-

- ond International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 291-297.
- [86] Nehmzow, U, and B. McGonigle (1993) Robot Navigation by Light. In: Deneubourg, J-L, et.al. (1993) Self-organisation and life: from simple rules to global complexity. Proceedings of the Second European Conference on Artificial Life. ULB, Brussels. p. 835-844.
- [87] Newell, A. (1982) The Knowledge Level. Artificial Intelligence Journal. Vol 18. p. 87-127.
- [88] Nicolis, G. (1989) Physics of far-from-equilibrium systems and self-organisation. In: Davis, P. (ed) The New Physics. Cambridge University Press, Cambridge UK. p. 316-347.
- [89] Nicolis, G. and I. Prigogine (1985) Exploring Complexity. Piper, Munchen.
- [90] Nilsson, N. (ed.) (1984) Shakey the Robot. SRI AI center. Technical Note 323.
- [91] Pasteels, J.M. and J-L Deneubourg (1987) From Individual to Collective Behaviour in Social Insects. Birkhauser, Basel.
- [92] Pattee, H. (1989) Simulations, realizations, and theories of life. In: Langton, C. (ed.) (1989) Artificial Life. Addison-Wesley, Redwood City, Ca. p. 63-77.



- [93] Peters, T. (1990) Liberation Management. Necessary Disorganization for the Nanosecond Nineties. MacMillan, London.
- [94] Pfeifer, R. and P. Verschure (1992) Distributed Adaptive Control: A Paradigm for Designing Autonomous Agents. In: Varela, F.J. and P. Bourguine (eds.) (1992) Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 21-30.
- [95] Prigogine, I. and I. Stengers (1984) Order Out of Chaos. Bantam Books, New York.
- [96] Ray, T. (1992) An Approach to the Synthesis of Life. In: Langton, C.G., C. Taylor, J.D. Farmer, and S. Rasmussen (1992) Artificial Life II. Proceedings of the Workshop on Artificial Life Held February, 1990 in Santa Fe, New Mexico. p. 325-371.
- [97] Rose, S. and S. Bullock (1991) The Chemistry of Life. Third Edition. Penguin Books, London.
- [98] Rosenblatt, F. (1962) Principles of Neurodynamics. Spartan Books, New York.
- [99] Rosenblatt, K.J. and D. Payton (1989) A Fine-Grained Alternative to the Subsumption Architecture for Mobile Robot Control. In: Proceedings of the IEEE/INNS International Joint Conference on Neural Networks.

- [100] Rosenschein, S. and L. Kaelbling. (1986) The synthesis of digital machines with provable epistemic properties. In: Halpern, J. (ed.) (1986) Theoretical Aspects of Reasoning about Knowledge. Morgan Kaufmann, Pub. San Mateo. p. 83-98.
- [101] Rumelhart, J., J. McClelland and the PDP Research Group (1986) Parallel Distributed Processing. MIT Press, Cambridge Ma.
- [102] SAB. Simulation of Adaptive Behavior Journal. MIT Press, Cambridge Ma.
- [103] Schwefel, H-P (1981) Numerical Optimization of Computer Models. Wiley, Chichester.
- [104] Shapiro, S.C. (1992) Encyclopedia of Artificial Intelligence. Second Edition. John Wiley and Sons, Inc. New York.
- [105] Shibata, T. and T. Fukuda (1993) Coordinative balancing in evolutionary multi-agent robot systems. In Varela, F.J. and P. Bourguine (eds.) (1992) Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 990-1003.
- [106] Smithers, T. (1992) Taking Eliminative Materialism Seriously: A Methodology for Autonomous Systems Research. In Varela, F.J. and P. Bourguine (eds.) (1992) Toward a Practice of Autonomous Systems. Proceedings of the

First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 31-40.

- [107] Smithers, T. (1993) Are autonomous agents information processing systems? In: Steels, L. and R. Brooks (eds.) (1993) The 'artificial life' route to 'artificial intelligence'. Building situated embodied agents. Lawrence Erlbaum Associates, New Haven.
- [108] Steels, L. (1990a) Cooperation between distributed agents through self-organisation. In: Demazeau, Y. and J-P Muller (ed.) (1990) Decentralized A.I.. North-Holland, Pub. Co., Amsterdam. p. 175-196.
- [109] Steels, L. (1990b) Exploiting Analogical Representations. In: Maes, P. (1990) Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back. MIT Press/Bradford Books, Cambridge Ma. p. 71-88.
- [110] Steels, L. (1991) Emergent frame recognition and its use in artificial creatures. Proceedings of the 10th IJCAI. Morgan Kaufmann, Pub. San Mateo.
- [111] Steels, L. (1991b) Towards a theory of emergent functionality. In: Meyer, J-A., and S.W. Wilson (1991) From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/ Bradford Books. Cambridge Ma. p. p. 451-461.

- [112] Steels, L. and R. Brooks (eds.) (1993) The ‘artificial life’ route to ‘artificial intelligence’. Building situated embodied agents. Lawrence Erlbaum Associates, New Haven.
- [113] Steels, L. (ed.) (1993) The Biology and Technology of Intelligent Autonomous Agents. NATO ASI Series. Springer Verlag, Berlin.
- [114] Steels, L. (1993b) Building Agents with Autonomous Behavior Systems. In: Steels, L. and R. Brooks (eds.) (1993) The ‘artificial life’ route to ‘artificial intelligence’. Building situated embodied agents. Lawrence Erlbaum Associates, New Haven.
- [115] Suchman, L. (1987) Plans and Situated Action. The problem of human machine interaction. Cambridge University Press, Cambridge UK.
- [116] Sutton, R.S. (1988) Learning to Predict by the Methods of Temporal Differences. Machine Learning, vol 3, p.9-44.
- [117] Sutton, R.S. (1991) Reinforcement Learning Architectures for Animats. In: Meyer, J-A., and S.W. Wilson (1991) From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books. Cambridge Ma. p. 288-296.
- [118] Sutton, R.S. (1992) Special Issue on Reinforcement learning. Machine Learning, vol 8, p. 3-4.

- [119] Todd, P.M. and S. Wilson (1993) Environment Structure and Adaptive Behavior From the Ground up. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books, Cambridge Ma. p. 11-20.
- [120] Toffoli, T. and N. Margolus (1977) Cellular Automata Machines. MIT Press, Cambridge Ma.
- [121] Tyrrell, T. (1993) The Use of Hierarchies for Action Selection. In: Meyer, J-A., H.L. Roitblatt, and S.W. Wilson (1993) From Animals to Animats2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. MIT Press/Bradford Books. Cambridge Ma. p.138-147.
- [122] Ullman, S. (1984) Visual routines. Cognition, vol 18, p. 97-159.
- [123] Van de Velde, W. (1992) Learning Robots. MIT Press/ Bradford books, Cambridge Ma.
- [124] Varela, F.J. and P. Bourgine (eds.) (1992) Toward a Practice of Autonomous Systems. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma.
- [125] Varela, F.J., E. Thompson and E. Rosch (1992) Embodied Mind. MIT Press, Cambridge Ma.

- [126] Verschure, P., B. Krose, and R. Pfeifer (1992) Distributed Adaptive Control: The Self-organization of Structured Behavior. *Robotics and Autonomous Systems*, vol 9, p. 181-196.
- [127] Vogel, F. (1989) *How Life Learned to Live*. MIT Press, Cambridge Ma.
- [128] Walter, W.G. (1950) An imitation of life. *Scientific American*, p. 42-45.
- [129] Webb, B. (1993) Modeling Biological Behaviour or ‘Dumb Animals and Stupid Robots’. In: Deneubourg, J-L, et.al. (1993) *Self-organisation and life: from simple rules to global complexity*. Proceedings of the Second European Conference on Artificial Life. ULB, Brussels. p. 1090-1103.
- [130] Webb, B. and T. Smithers (1992) The Connection between AI and Biology in the Study of Behaviour. In: Varela, F.J. and P. Bourgine (eds.) (1992) *Toward a Practice of Autonomous Systems*. Proceedings of the First European Conference on Artificial Life. MIT Press/Bradford Books, Cambridge Ma. p. 421-428.
- [131] Weisbuch, G. (1991) *Complex Systems Dynamics. An Introduction to Automata Networks*. Lecture Notes Volume II. Santa Fe Institute Studies in the Sciences of Complexity. Addison-Wesley Pub. Cy. Reading Ma.
- [132] Widrow, B., N. Gupta and S. Maitra (1973) Punish/reward: Learning with a critic in adaptive threshold systems. *IEEE Transactions on Systems, Man, and Cybernetics*. 5, p. 455-465.

- [133] Wilson, S.W. (1991) The Animat Path to AI. In: Meyer, J-A., and S.W. Wilson (1991) From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. MIT Press/ Bradford Books. Cambridge Ma. p. 15-22.