Connection Science, Vol. 16, No. 4, December 2004, 249–266

Structure and function of evolved neuro-controllers for autonomous robots

Martin Hülse, Steffen Wischmann and Frank Pasemann
Fraunhofer Institute for Autonomous Intelligent Systems (AIS),
Schloss Birlinghoven, 53754 Sankt Augustin, Germany
email: {martin.huelse, steffen.wischmann,
frank.pasemann}@ais.fraunhofer.de

Abstract. The artificial life approach to evolutionary robotics is used as a fundamental framework for the development of a modular neural control of autonomous mobile robots. The applied evolutionary technique is especially designed to grow different neural structures with complex dynamical properties. This is due to a modular neurodynamics approach to cognitive systems, stating that cognitive processes are the result of interacting dynamical neuro-modules. The evolutionary algorithm is described, and a few examples for the versatility of the procedures are given. Besides solutions for standard tasks like exploration, obstacle avoidance and tropism, also the sequential evolution of morphology and control of a biped is demonstrated. A further example describes the co-evolution of different neuro-controllers co-operating to keep a gravitationally driven art-robot in constant rotation.

Keywords: modular neuro-dynamics, nonlinear robot control, perception–action systems, structure evolution, evolution of morphology.

1. Introduction

The artificial life approach to evolutionary robotics (Nolfi and Floreano 2000) applies evolutionary techniques not only to optimize a given system with respect to desired fixed properties, but also to emphasize the emergence of solutions for a behavioural task that sometimes may not be anticipated. In particular, if the connectivity structure of neural control networks is subjected to an evolutionary process without any constraints, the appearance of feedback loops in networks, together with associated non-trivial dynamical properties, may contribute considerably to an appropriate behaviour of the system. Applying neural control to autonomous physical robots acting in open and changing environments will perhaps allow comparison with the behaviour of living systems. It may be stated that one of the goals of evolutionary robotics is to compare the functions of biological brains with the functions of artificial neural control mechanisms for animats. These ideas have already been formulated during the cybernetics area, where one of the initial postulates was to identify behaviour of living systems by means of self-regulating technological objects (Walter 1953, Wiener 1962, Ashby 1966).

Furthermore, evolutionary robotics sets up an excellent framework for testing and applying a modular neuro-dynamics approach to cognitive systems. Like other dynamical approaches to cognitive science (Beer 1995, Kelso 1995, Port and van Gelder 1995), it tries to understand cognition as a global process unfolding during the interaction with multiple external and internal processes. Strong arguments for this hypothesis arise,

Connection Science ISSN 0954-0091 print/ISSN 1360-0494 online © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/09540090412331314795

besides the findings in biological brains, from the observation that already small artificial neural networks with recurrent connectivity inherit complex dynamical features (Pasemann 2002); for example, there is a variety of oscillatory modes including chaos, and often there exists a set of different modes from which a specific mode can be accessed depending on the history of the system. But it is still difficult with today's knowledge to provide a reasonable theoretical foundation for this dynamical approach to cognition.

If one thinks about small neural networks as basic building blocks for larger control systems, i.e. as modules generating specific behaviours, the (recurrent) coupling of these non-linear subsystems can lead to many undesired or unexpected behaviours of the composed system; but it is exactly the emergence of these unpredictable attributes on which the hope of finding neural structures carrying the behaviour-relevant dynamics rests. This may be difficult to achieve, but many experiments performed in the spirit of an artificial life approach to evolutionary robotics confirm that the apparent behavioural complexity of artefacts is not caused primarily by the complexity of their neural control structures but reflects the complexity of their bodies and of the environment in which they are acting. Very simple mechanisms, which can lead to interesting life-like behaviour, have already been suggested by Braitenberg (1984) in his famous thought experiments.

Here it is presupposed that experiments in the realm of evolutionary robotics can contribute substantially to a better understanding of the basic principles of behaviour relevant neural processing. The goal therefore is to find convincing examples pointing to the realism of the modular neuro-dynamics hypothesis, which calls for the combined application of nonlinear dynamics analysis, artificial life ideas, and evolutionary computation techniques. To obtain reasonable results, the presented software package, called Integrated Structure Evolution Environment (ISEE), includes not only techniques for structure evolution, but also tools for fusion and 'lesion' experiments, for co-evolution, for taking 'electrode data', for non-linear analysis of evolved structures, and for linking different simulators and physical robot platforms to the central evolutionary algorithm.

The following section describes the adopted neural network model with its discrete-time dynamics and the utilized evolutionary algorithm. Fusion techniques for the combination of functionally segregated neuro-modules are outlined as well as techniques for the generation of robust controllers, and for analysis of structure–function relations of evolved neuro-controllers. Section 3 demonstrates some of the results obtained by the ISEE package, where emphasis is placed on small analysable neural controllers. The first example demonstrates how a simple recurrent structure generates an effective obstacle-avoidance behaviour using hysteresis phenomena. Then an example for module expansion is given, resulting in an additional light tropism. Section 3.3 describes an example where, after the evolution of the morphology of a passive walker, an evolved neural network controls a biped with minimal motoric equipment. The last example, in section 3.4, demonstrates an application of co-evolution to control the gravitational drive of a rotating ring which is realized by five movable arms. The last section summarizes the results.

2. ISEE: an integrated structure evolution environment

The neural networks discussed in the following are evolved as controllers generating specific behaviours. Working in the context of a modular neurodynamics approach to cognitive systems, these networks are conceived as basic building blocks for larger systems and are therefore termed *neuro-modules*. Their connectivity structure will be of a general recurrent type, i.e. there may exist closed loops of directed signal transmission lines. Fusion of behaviours is assumed to be realizable through the coupling of these modules. Because they are conceptualized as non-linear dynamical systems, their coupling may even result in emergent properties, which have not been observable before for the system under consideration.

A neuro-module with n units is defined as a parameterized discrete-time dynamical system given by

$$a_i(t+1) = \theta_i + \sum_{j=1}^{n} w_{ij} f(a_j(t)), \quad i = 1, \ldots, n, \tag{1}$$

where $a_i \in \mathbf{R}^n$ denotes the *activity* of neuron i, ω_{ij} the synaptic strength or *weight* of the connection from neuron j to neuron i, and $\theta_i = \bar{\theta}_i + I_i$ denotes the sum of its fixed bias term $\bar{\theta}_i$ and its stationary external input I_i, respectively. The $o_i = f(a_i)$ of a unit i is given by an appropriate sigmoidal transfer function, for instance by $f := \tanh$. Thus, a neuro-module in general has $q := n \cdot (1+n)$ parameters, represented by a parameter vector $\boldsymbol{\rho} := (\theta, \omega) \in R^q$; and every $\boldsymbol{\rho}$ defines a particular dynamical system.

2.1. *Evolution of neural systems by stochastic synthesis (ENS^3)*

Like all evolutionary strategies, the *ENS*^{3} algorithm is an implementation of a variation–evaluation–selection loop operating on a population of n neuro-modules p_i ($i = 1, \ldots, n$). Its general functionality is formally described in table 1.

For this algorithm a population is divided into two sets, parents $\mathcal{P}(t)$ and offspring $\hat{\mathcal{P}}(t)$. The parameter t refers to the generation of the population. At the beginning the evolutionary process is initialized with a set of neuro-modules. This set contains an arbitrary number of neuro-modules, which have equal numbers of input and output neurons and the same transfer function. Transfer function and input–output structure are defined as problem-specific according to the task the neuro-controllers will be evolved for. There is an additional condition: input neurons are used only as a buffer, so no backward connections to the input neurons are allowed. Apart from that, nothing else is determined, not the number of hidden neurons or their connectivity, i.e. self-connections and every kind of recurrences are allowed, as well as excitatory and inhibitory connections.

The *reproduction* operator creates m_i copies of each individual neuro-module p_i in set $\mathcal{P}(t)$. The copies represent the set of offspring $\hat{\mathcal{P}}(t)$ in generation t. The number

Table 1. *ENS*^{3} algorithm (see text for explanation).

```
Begin
  $\mathcal{P}(0)$: = set_of_initial_structures;
  $t$: = 0
  Repeat
    $\hat{\mathcal{P}}(t)$: = reproduction ($\mathcal{P}(t)$);
    variation ($\hat{\mathcal{P}}(t)$);
    evaluate ($\mathcal{P}(t) \cup \hat{\mathcal{P}}(t)$);
    $\mathcal{P}(t+1)$: = selection ($\mathcal{P}(t) \cup \hat{\mathcal{P}}(t)$);
    $t$: = $t+1$
  Until stop_criterion;
End.
```

of offspring m_i for each neuro-module p_i is calculated by the selection operator. At the beginning ($t = 0$) this number is set to one for each module in the set of initial structures.

The *variation* or mutation operator is a stochastic operator and is applied to $\hat{\mathcal{P}}(t)$; the neuro-modules of $\mathcal{P}(t)$ remains unchanged. It realizes the combinatorial optimization as well as real-valued parameter optimization (Beyer 2001). Real-valued optimization refers to the variation of the bias and weight terms. Combinatorial optimization means insertion and deletion of hidden neurons and connections. Combinatorial optimization is determined by per neuron and per connection probabilities. Each structural modification is computed according to a given probability q and a uniform distributed random variable $u(0,1)$. For example, considering the deletion of hidden neurons, a random variable u_i is computed for each existing hidden neuron h_i. If $u_i \leq q_{delNeu}$ the hidden neuron h_i is deleted. Real-valued optimization is computed by using a Gaussian distributed random variable $g(\mu,\sigma)$. A weight term w is modified according to this variable: $w + g(0,\sigma)$. Such a modification is again determined by a uniform distributed variable $u(0,1)$ and a given probability $q_{modWeight}$.

The evaluation operator is given in terms of a fitness function F. Fitness function F determines a performance or fitness value v_i for each neuro-module $p_i \epsilon \mathcal{P}(t) \cup \hat{\mathcal{P}}(t)$. To keep the structure of the evolved neuro-modules during the structure evolution in certain ranges, the fitness value v_i takes into account the number of hidden neurons h_i and connections s_i of neuro-module p_i:

$$v_i := f_i - c_{Neu} \cdot h_i - c_{Con} \cdot s_i \cdot \tag{2}$$

The value f_i is the fitness value calculated with respect to the given task. The factors $c_{Neu}, c_{Con} \geq 0.0$ are cost terms for a hidden neuron and a connection, respectively.

In each generation the evaluation operator computes the fitness value v_i for all $p_i \in \hat{\mathcal{P}}(t) \cup \hat{\mathcal{P}}(t)$. Note that, the fittest parents of generation t can have a bad performance in the next generation $t + 1$, because the initial conditions may have changed.

The *selection* operator is again a stochastic operator. It determines which neuro-module in the set $\mathcal{P}(t) \cup \hat{\mathcal{P}}(t)$ becomes a member of the parent set of the next generation $\mathcal{P}(t + 1)$. This is done in the following way: according to the fitness value v_i for each neuro-module p_i its number of offspring m_i is calculated. A neuro-module p_i is passed to the next generation iff $m_i > 0$. The participation of both parents and offspring in the selection takes into account the low success probability of combinatorial optimization problems (Beyer 2001). The progress of a combinatorial optimization can be attained only through conservation of good neuro-modules, i.e. parent individuals must not 'die' per definition.

The computation of the number of offspring m_i is rank based. That means, according to a given set of neuro-modules p_i and their fitness values v_i the number of offspring m_i is proportional to its rank in this set, not to its actual fitness value. The number m_i is derived from a Poisson distribution with parameter λ_i and this parameter is calculated as follows. Assume a population P of n neuro-modules p_i with their corresponding fitness value v_i then a function $h(x)$ is defined as:

$$h(x) := e^{-\gamma/\sigma^2 \cdot (v_{\max} - x)}, \tag{3}$$

where $v_{\max}$ denotes the maximal fitness value within the current generation, σ^2 the standard deviation of the fitness values i and γ a non-negative real value. Then the parameter λ_i for neuro-module p_i is calculated as follows:

$$\lambda_i := \frac{z}{\sum_{j=0}^{n} h(v_i)} \cdot h(v_i)\cdot \tag{4}$$

The first factor is simply for normalization, where z can be interpreted as the average population size. The introduced parameter $\gamma \in [0, +\infty]$ represents the pressure of selection. The larger the γ, the greater the number of offspring of the best neuro-modules, while the sum of all offspring remains constant in certain ranges. Therefore we can say the larger the γ, the more elite selection is forced (Dieckmann 1995).

The evolutionary process takes place in a repeat-until loop as long as a special stop criterion is not fulfilled. Up to now there has been no formal stop criterion implemented. The user has to decide when the evolution can or has to stop manually. Therefore, the user needs online monitoring of all essential process parameters, such as fitness values.

2.2. *Combination of different behaviours by structure evolution*

Assume A and B are two behavioural tasks, which should be solved by a neuro-module $M_{A,B}$. Within our approach to structure evolution we distinguish two fundamental methods to create such a module $M_{A,B}$. We call them *module expansion* and *module fusion*. Module expansion starts with a neuro-module M_A, which solves task A. Then the desired neuro-module $M_{A,B}$ is developed by a 'growing-up' process via structure evolution on module M_A. Module fusion starts with two neuro-modules M_A and M_B. Then the neuro-module $M_{A,B}$ is developed by the evolution of a connectivity structure between the modules M_A and M_B, which may include additional neurons. These two methods can easily be applied to the *ENS*^{3} algorithm, since the set of initial structures can contain neuro-modules of arbitrary type as long as they have the same type of neurons and the same number of inputs and outputs.

Additionally, both methods can be differentiated by the definition of the initial structures that the evolutionary process is started with. In general, there are three kinds of initialization that bring us to three module combination techniques (Pasemann *et al.* 2001):

(1) *Restrictive*: The structure and parameters of the neuro-modules in the initial populations stay fixed during evolution. Evolved neuro-modules will have additional connections and hidden neurons.

(2) *Semi-restrictive*: Same as restricted module combination, but now only the structure of the initial neuro-modules is fixed. Parameter values such as weights and bias terms may be modified.

(3) *Free*: Neither the initial structure nor its parameters are fixed during the evolution.

Note that, within this framework, incremental evolution (Nolfi and Floreano 2000) becomes a type of free module expansion.

2.3. *Evolving minimal and robust control structures*

Since robustness of control is one of the most important qualities we want to achieve by our approach to evolutionary robotics, we already try to enforce a robust behaviour during the process of evolution. Therefore, the boundary and environmental conditions are varied for each generation. As an indicator of robustness we chose the age of a neuro-module, meaning the number of generations that an unvaried neuro-module survived. A high age is an indication for a so-called generalist, whereas a low age indicates a specialist.

To clarify the correlation of structure, dynamical effects and behaviour there is a need for neuro-modules with a sufficiently small number of hidden neurons and connections. To derive such modules we proceed as follows. At the beginning of an evolutionary process the fitness of an individual is determined only by the success of solving the given task. Thus, neural structures can grow as long as they lead to an improved success. If the behaviour seems to be sufficient we increase the probability for deletion of neurons and connections and at the same time we introduce costs for neurons and connections. By this means we promote neuro-modules of smaller size having the same fitness.

To encourage the development of minimalism and robustness we are able to monitor all relevant statistical data during the evolutionary process, such as fitness values, ages, number of neurons and connections, population size, etc. The evolutionary environment also allows online modification of all process parameters that were introduced in the previous section. Examining age, size, fitness and observed behaviour, we stop the evolutionary process of neuro-modules if all of the mentioned criteria are matched.

2.4. *Analysing structure–function relations*

The investigation of the structure–function relation of evolved neuro-modules is supported by four main software components: (1) visualization by graphs of neuro-modules with arbitrary recurrent connectivity structure; (2) analysis of neuro-modules as dynamical systems; (3) plotting of neuron activities during the robot–environment interaction; and (4) modification of the inner structure of modules, i.e. lesion experiments. These software tools somehow reflect our approach to the clarification of the relation between neural connectivity structure, dynamical effects and robot behaviour.

Besides the analysis tool for modular neurodynamics, all components are integrated in one software package. Therefore, hypotheses about the relation between neural structure and resulting robot behaviour can be developed fast and tested.

The main reason to apply analysis of formal dynamics, neural activity data and lesion experiments is to find specific neural connectivity structures that can be understood as realizations of general control techniques which become applicable to other tasks and robot platforms.

The visualization gives a first impression of module structure. It plots the connectivity structure as a directed graph showing which input neurons and therefore which sensor qualities are required to perform the task. Additionally, it gives an overview of substructures and their connectivity. They can also help us to find well-known configurations such as super-critical self-connections and loops, which can generate non-trivial dynamical effects (Pasemann 2002) and may have relevant influence on the resulting behaviour.

The potential dynamical effects of a neuro-module as an isolated dynamical system are studied by considering the sensor inputs as parameter inputs. In particular, the parameter dependence of attractor structures is examined, i.e. relevant parameter domains for stationary, periodic or chaotic attractors are identified. Correlating these results to the observed behaviour of the controlled robot gives a first hypothesis concerning the structure–function relation for the evolved neuro-module.

Such a hypothesis can substantiated by the online plot of specific neuron activities during the robot–environment interaction. Additionally, we can fix each neuron output with an arbitrary value. This enables us to correlate neural activity to environmental properties, as they are reflected in the sensor signals, to observe special signal flow in the neuro-module, and to verify the relevance for and influence on specific neuron activities with respect to the behaviour. This somehow corresponds to the usage of electrode data taken from biological brains in animal experiments.

For final investigations of this hypothesis, 'lesion experiments' can be executed. Selected hidden neurons and connections are deleted online and parameters of all neurons can be modified while the performance of the remaining neural structure is studied.

3. Applications

3.1. *Formalizing a non-linear control technique*

In this section, we demonstrate the formalization of a control technique, which was derived from investigations on a neuro-module for obstacle avoidance. The experiments were performed on the Khepera robot (Mondada *et al.* 1993). For simulation and behaviour evaluation a two dimensional simulator (Michel 1997) was applied. We evolved control structures using only two input and two output neurons.

Input neuron I_1 has the average value of the three left proximity sensors. They are mapped linearly on to the interval $[-1, +1]$, where -1.0 presents the free space and $+1.0$ a very short distance to an obstacle or even a collision. Input neuron I_2 processes the three right proximity sensors in the same way.

Two output neurons O_1 and O_2 were used to drive the motors of the left and right wheel, respectively. We applied *tanh* as transfer function for the neuro-module to get directly positive and negative control signals as output values. A positive control signal drives the wheel forward and a negative backward.

In figure 1 one resulting module is shown, which we have selected for investigations of structure–function relations. This module is called *MRC*, for minimal recurrent controller. Although we started the evolutionary process with an empty initial structure this controller is fully connected, but does not use any hidden neurons. The most interesting structure elements of the MRC are the super-critical self-connections (>1.0) of both output neurons and the loop between them. Based on analytical investigations (Pasemann 1997, 2002) we know that such elements can produce hysteresis effects. The resulting obstacle avoidance behaviour is very robust. The robot does not collide with objects in its environment and is able to escape even from sharp corners and dead-ends.

If we plot the neuron activities of this neuro-module while the robot is running we can observe the robust switching of the output neurons despite the noisy input signals (figure 2). The neuro-module is working as a filter mainly generating two defined output values $\approx +1.0$ to drive as fast as possible forward and ≈ -1.0 to turn away from appearing objects.

The realization of several lesion experiments has shown which structural elements of the MRC are responsible for the observed filter effects. Figure 3 gives an impression of

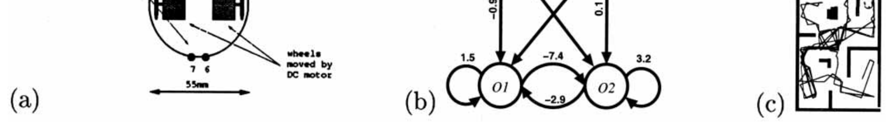

Figure 1. (a) Scheme of the Khepera robot; (b) the MRC neuro-module; (c) behaviour of the robot controlled by the MRC.

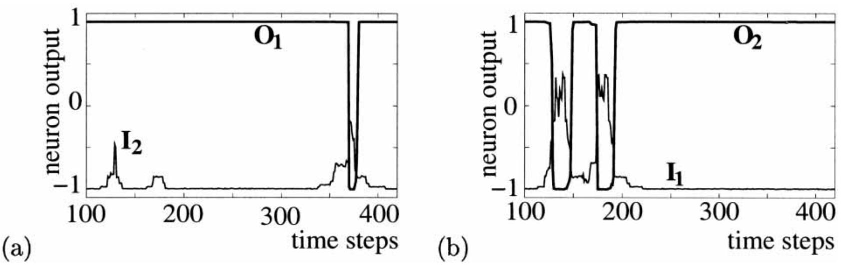

Figure 2. Neuron activities of input and output neurons while the robot moves forward and turns away from obstacle on the left and right. (a) A small increase of input signal I_2 does not influence the output signal between time step 100 and 200); but large input values of I_2, produce a defined switch to output signal ≈ -1.0 on output neuron O_2, which produce a turn to the left. (b) The same holds for the output neuron O_1 corresponding to input neuron I_2. In this case, the switch to ≈ -1.0 produce a turn to the right.

how these experiments were performed. It is demonstrated that the interplay of three different hysteresis effects corresponding to the three loops is responsible for the robust robot behaviour (Hülse and Pasemann 2002). Each effect becomes active in special situations, which are: (1) obstacle on the left; (2) on the right; and (3) sharp corners or dead-ends. They produce a turning of the robot, where the turning angle is proportional to the self-weights and the two-loop strength, respectively, i.e. to the active hysteresis domains (Hülse *et al.* 2003).

The investigations on the MRC summarize a formalization which we call the *dynamical neural Schmitt trigger* (Hülse and Pasemann 2002). The dynamical neural Schmitt trigger is a single neuron with a positive self-connection $w_s > 1$, an input E, and slowly varying bias value B, described as follows (see figure 4 for notation):

$$A(t+1) = \tanh(B + E \cdot w_d + A(t) \cdot w_s), \quad w_s > 1. \tag{5}$$

The two transition points E_1 and E_2 for external inputs are then given by:

$$E_{1,2} = \frac{\Theta_{1,2} - B}{w_d}, \tag{6}$$

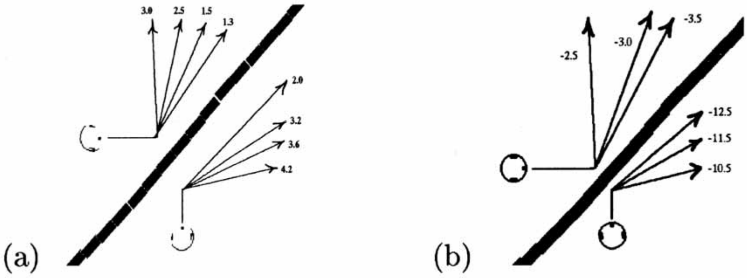

Figure 3. Some observations of the 'lesion' experiments performed with the MRC. (a) Left: resulting robot behaviour with different self-connections on O_1. (a) Right: behaviour with different self-connections of O_2. (b) Left: the connection from I_2 to O_1 is modified. (b) Right: resulting behaviour with different weight parameter for the connection from I_1 to O_2.

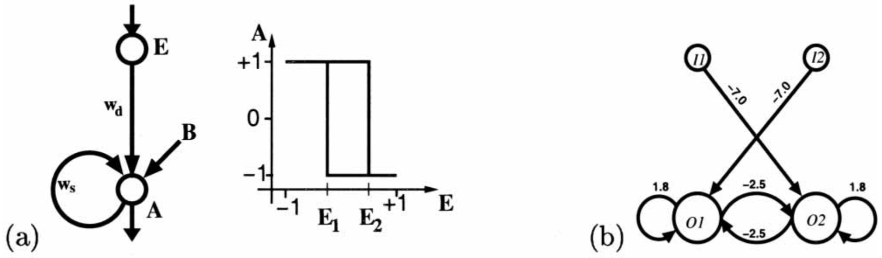

Figure 4. (a) Connectivity structure of the dynamical neural Schmitt trigger; (b) manually designed neuro-module for obstacle avoidance driven by the interplay of three hysteresis effects.

where B denotes an additional stationary (or slow) external input to the neuron (Hülse and Pasemann 2002). The calculation of the end points $\Theta_{1,2}$ of the hysteresis interval shows that its width depends on the self-connection w_s:

$$\Theta_{1,2} = \pm\left| ln\left(\frac{1+\alpha}{1-\alpha}\right) - 2w_s\alpha\right|, \quad \alpha := \sqrt{1-\frac{1}{w_s}}, \quad w_s > 1. \tag{7}$$

Figure 4(b) shows one application of the *dynamical neural Schmitt trigger*. This symmetric and reduced neuro-module contains the essential structural elements to produce the same robust obstacle avoidance behaviour as the MRC. The advantage of this module is that turning angles can be changed by manual modification of the connection parameters. In this way, the specific properties of the different robot platforms (noise intensity, sensor qualities and quantities, actor signal processing, body) can be directly taken into account. This was successfully tested and applied for various wheel-driven robot platforms, e.g. for the diverse robot platforms of the Fraunhofer Institute AIS: VolksBot, Kurt2, etc. (see http://www.ais.fraunhofer.de).

3.2. *Behaviour expansion by structure evolution*

In this experiment, the symmetric obstacle avoidance module (figure 4 (b)) is functionally enhanced to solve a light-seeking task. This is done by the restricted module expansion technique (see section 2.2). Four additional input neurons $I_3,\ldots,I_6$ are added to the original input structure. The values of these input neurons are the mean values of the two light sensors at the left (I_3), the front (I_4), the right (I_5) and the rear (I_6) of the Khepera robot (compare figure 1(a)). In contrast to the proximity sensors, the light sensor values are mapped on to the interval [0,1], where 1.0 represents darkness and 0.0 brightness.

The structure of the expanded module and the resulting behaviour are indicated in figure 5. For detail we give the weight matrix W of the inner structure.

$$\boldsymbol{W} = \begin{pmatrix} 0.0 & 0.0 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & 0.0 \\ \vdots & \vdots & & & & & & & \vdots \\ 0.0 & 0.0 & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & 0.0 \\ 5.1 & -7.0 & -2.1 & -4.2 & 6.7 & 0.4 & 1.8 & -2.5 & -3.0 \\ -7.0 & 2.3 & -1.3 & -8.1 & 3.9 & 5.1 & -2.5 & 1.8 & 4.9 \\ -3.4 & 1.0 & 9.8 & -8.5 & -2.5 & 5.3 & 0.0 & -4.4 & -1.9 \end{pmatrix}$$

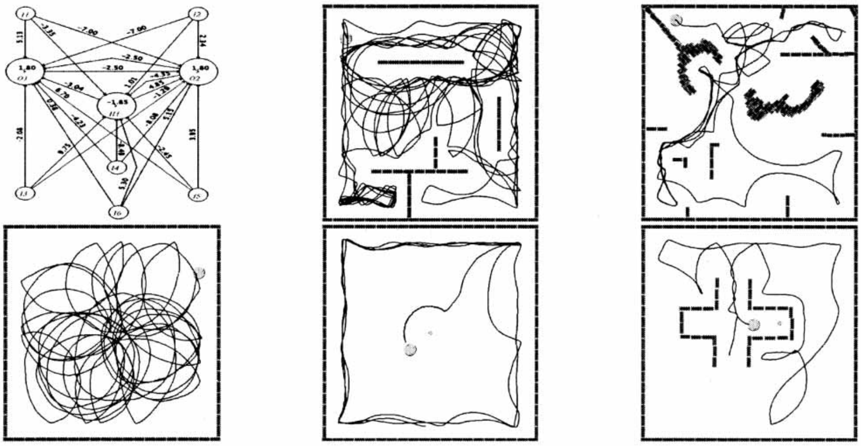

Figure 5. An effective controller for obstacle avoidance and light-seeking behaviour.

The plotted paths show the robot's capabilities of searching and approaching a light source, and the coming to a halt in front of it while it avoids collisions with other objects. Applying the module to the physical robot, the observed behaviour is qualitatively the same.

The two-neuron configuration of the additional hidden neuron H_1 and O_2 is denoted as *chaotic 2-module* (Pasemann 2002) because it allows complex dynamics such as oscillations and chaos. In the diagrams of figure 6 the plotted neuron output signals give an impression of how different oscillations become active in different situations (Hülse and Pasemann 2004). For example, figure 6(a) reflect the situation where the robot does not detect any light or obstacles. In this case, a period-five oscillation is active, which produces an average output of 0.6 on O_1 and 0.7 on O_2 (see also table 2). This configurations is related to a slightly curved movement, as shown by the robot paths. In contrast, a light source in front of the robot forces the neuro-module to generate a period-five oscillation, which produces an average output of zero on both output neurons resulting in a quasi-halt of the robot. In addition, although sensor values remain constant (except for noise), the apparent period-five oscillations produce varying amplitudes on O_1 and O_2. This causes randomly varying turning angles of the curved robot movement.

3.3. *Evolving morphology and control*

This section demonstrates that our evolutionary environment is not limited to the development of control architectures for autonomous agents. In this application we optimized the morphology of a bipedal walking device as well as the control structure.

The possibility of creating complex behaviours such as bipedal walking without any control mechanism is demonstrated in McGeer (1990) and Collins *et al.* (2001). There, rigid body dynamics were used for generating stable walking pattern. As a drawback, their gravity-driven machines were limited to walking on shallow slopes. We were inspired firstly to evolve the morphology of similar machines and, secondly, to enable them to walk on flat surface by evolving a suitable neuro-controller. The resulting walking device should use a minimal sensorimotor system and avail itself of the passive rigid body dynamics to produce energy-efficient walking.

Here, only a general survey of the experiments is given. A detailed description of the methods and results can be found elsewhere (Wischmann and Pasemann 2004). To

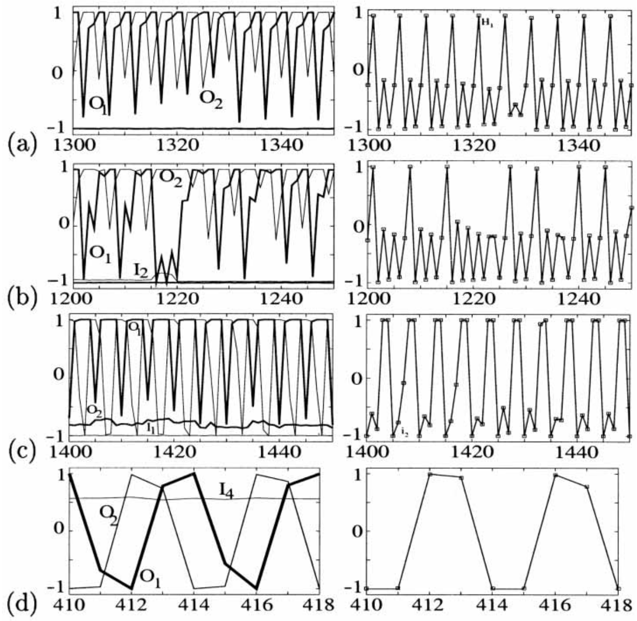

Figure 6. Output signals of neuron O_1 and O_2 (left) and the hidden neuron H_1 (right). (a) All sensor input values are minimal, e.g. no light source and no obstacle is detected. (b) The robot moves in the dark, while it detects an obstacle on the right and avoids it by a left turn. The appearing obstacle is indicated by an increasing input value I_2 (fine line). (c) The robot moves in the dark, while it detects an obstacle on the left (high input value I_1, thick line). (d) The robot stands in front of a light source ($I_4 \approx 0.55$); other input values are at their minimum. In this case, the average output of O_1 and O_2 is zero, which produces the quasi 'standing' in front of a light.

Table 2. Average value of neuron outputs for specific input configurations. These input value configurations refer to specific situations in the environment. The resulting behaviour can be derived by the average output values, e.g. $O_1 = 1.0$ and $O_2 = -1.0$ refers to a turn to the right.

	Average output							
		Obstacle			Light			
Neuron	Empty space	Right	Left	Both	In front	Left	Right	Back
I_1	-1.0	-0.9	0.2	0.1	-1.0	-1.0	1.0	-1.0
I_2	-1.0	0.7	-0.9	0.0	-0.9	-1.0	-1.0	-1.0
I_3	0.0	0.0	0.0	0.0	0.1	0.4	0.0	0.0
I_4	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0
I_5	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.0
I_6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6
O_1	0.6	-1.0	1.0	0.0	0.0	1.0	0.7	1.0
O_2	0.7	1.0	-1.0	0.0	0.0	-1.0	1.0	-0.6
H_1	-0.4	-0.1	0.9	0.0	0.0	0.6	-0.4	0.2

realize passive dynamic walking devices, first, models with one and three degrees of freedom were implemented within the physical simulation environment Open Dynamics Engine (2004). These models are illustrated in figure 7 (left).

Evolvable parameters are the dimensions and mass distributions of the single body parts and additional initial conditions, e.g. the initial impulse. We hand-crafted a neuro-module consisting of as many connections as parameters which should be optimized. In contrast to the development of neuro-controllers, aiming for a certain behaviour, the inner dynamics of the neuro-module are irrelevant for these experiments. During evolution no variation of the structure is allowed; only the weights of the given connections could be varied. These weights are mapped to the parameters of the walking device. Evaluation, reproduction and selection are processed as described in section 2.1. The fitness function takes the distance covered and the number of successful steps into account. The behaviour of the resulting morphology for both kinds of passive dynamic walking devices is sketched in figure 7 (middle and right).

Since there is no actuation at all, the gait patterns of the walking devices are generated only by their rigid body dynamics with gravity as power supply. Because these passive walker models are still limited to a very specific environment, the shallow slope, the next step was to create a sensor motor system and a controller for enabling adaptation to flat surface. The one-degree-of-freedom model was equipped with minimal actuators (three motors) and sensors (three internal, one external). Controlled by a neuro-module, it should now be able to walk on a level surface with minimal energy consumption. As demonstrated in Wischmann and Pasemann (2004), we successfully evolved a neuro-module that is able to perform this task. The presented recurrent neuro-module is again small enough for empirical analysis of its behaviour-relevant inner dynamics. This enabled us to demonstrate how closely morphology and control are related. It was shown that the gait pattern is mainly determined by the passive dynamics of the robot's body. The controller only compensates the energy loss that occurs during walking on a level surface, e.g. at heel strike, whereas, for example, the swing periods of the legs are purely passive.

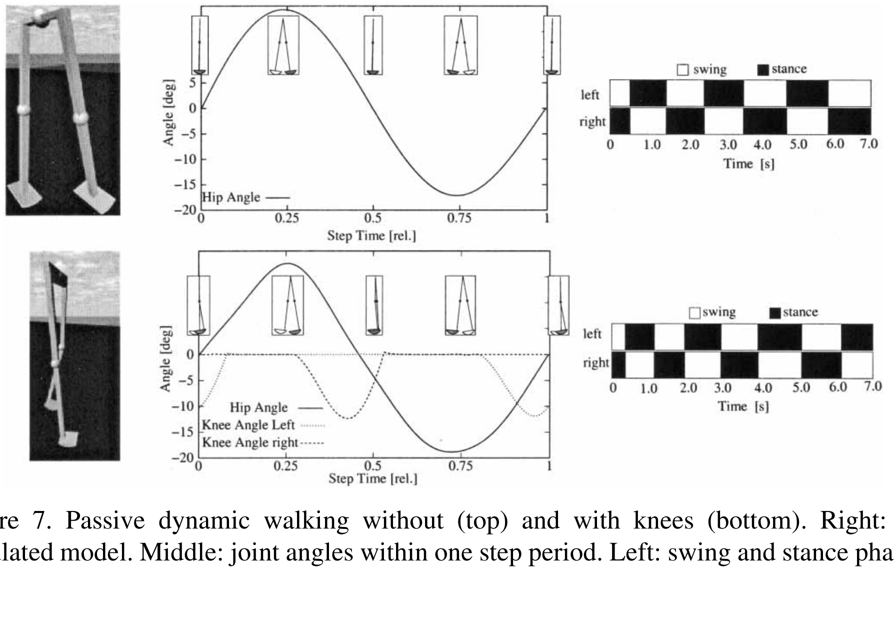

Figure 7. Passive dynamic walking without (top) and with knees (bottom). Right: the simulated model. Middle: joint angles within one step period. Left: swing and stance phases.

3.4. *Co-evolution*

To demonstrate that our evolutionary environment is also able to perform co-evolution strategies we used the art-robot *micro.eva* developed by Julius Popp (2003). As can be seen in figure 8, this gravity-driven robot consists of five arms that are connected to a ring by four hinge joints. Each arm can translate its centre of mass by a servo motor which is connected to one joint within the arm. The motors and therefore the main mass distributions are located in the arms, so the robot is able to rotate the ring by moving its masses, respectively. Owing to the simplicity of the task, just rotating as fast as possible at a constant speed, it is promising to use this robot as a benchmark system for diverse control techniques.

In our control approach we consider each arm as an autonomous agent. Agents can indirectly communicate to each other with sensors located inside the ring. The angular velocity of the ring is provided by a gyroscope, and five hall sensors are placed inside the ring. These sensors are binary switches that emit a peak if they come to pass a magnet near the bottom of the ring. A built-in potentiometer in each servo motor provides information about the actual motor position.

The ring velocity is supposed to be maximal at 0.25 rounds per second. It becomes positive for counterclockwise and negative for clockwise rotation. The five hall sensors were integrated to one sensor input, which can have five discrete values according to the different position areas. Hence, each agent is able to distinguish its position relative to these five different areas. The motor position sensor gives the actual difference angle of the servo motor to its zero position. The motor output represents the desired angle of the servo motor. All input and output signals were linearly mapped on to $[-1.0, 1.0]$ due to the use of tanh as a transfer function. Thus, the initial neuro-module structure of each agent consists of three sensor inputs (ring velocity I_1, motor position I_2, integrated hall sensor I_3) and one motor output (desired angle O_1).

The neuro-modules of the different arms are evolved in separate populations. The evolutionary process for each population is processed in the same way as described in section 2.1; but, differently, all populations are evolved simultaneously, and for the evaluation process one agent from each population has to be chosen. This choice is rank-based and determined by the fitness values. Note that here an equal population size is necessary because each neuro-module of every population has to be evaluated exactly once in each generation. The selection process for a single population is identical to the description in section 2.1. After the selection, different population sizes can occur.

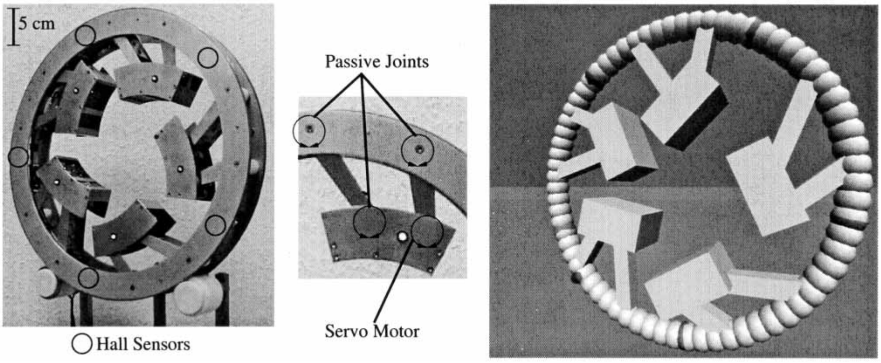

Figure 8. Art-bot *micro.eva*. Left: real robot and the location of the hall sensors. Middle: connections of a single arm to the ring. Right: physical simulation.

The maximum population size is taken and populations with a smaller number of individuals are filled with additional offspring of the parent with the highest fitness value of the according population. Since the fitness values of the offspring are not determined, this simple procedure at least assures that the fittest individuals of the previous evaluation are within the same group of individuals for the following evaluation cycle.

Owing to the fact that the five agents have to fulfil a co-operative task, the fitness function takes into account the group performance. The angular velocity $I_1(t)$ of the common body, the ring, together with its equability $I(t) - I_1(t-1)$ are evaluated and the resulting fitness value $(|I_1(t)| - |I_1(t) - I_1(t-1)|)$ is assigned to each single agent. Therefore, every agent of a group has the same fitness value, which can differ within the agents of a group if we introduce costs for hidden neurons and connections that are subtracted from the performance value. For the evaluation we modelled the robot within the physical simulation environment Open Dynamics Engine (figure 9, right).

The results of this co-evolution strategy for *micro.eva* can be seen in figure 9. Starting from generation 13 (performance was close to its maximum, $f_i = 100.0$), costs for hidden neurons and connections were introduced (see section 2.1) to minimize the structure of the neuro-modules. The resulting effect on the average number of hidden neurons and connections can be seen clearly. For four of the five populations a strong decrease was observed. After 25 generations no performance increase and structure decrease was noticed. For this reason, we stopped the evolutionary process manually and applied the fittest individual of each population to the simulated model of *micro.eva*.

By observing the behaviour of the robot generated by the five independent controllers, we could see that module 3 is important for initializing the rotation. If the ring stands still, arms 1, 2, 4 and 5 move to a position according to the initial orientation of the

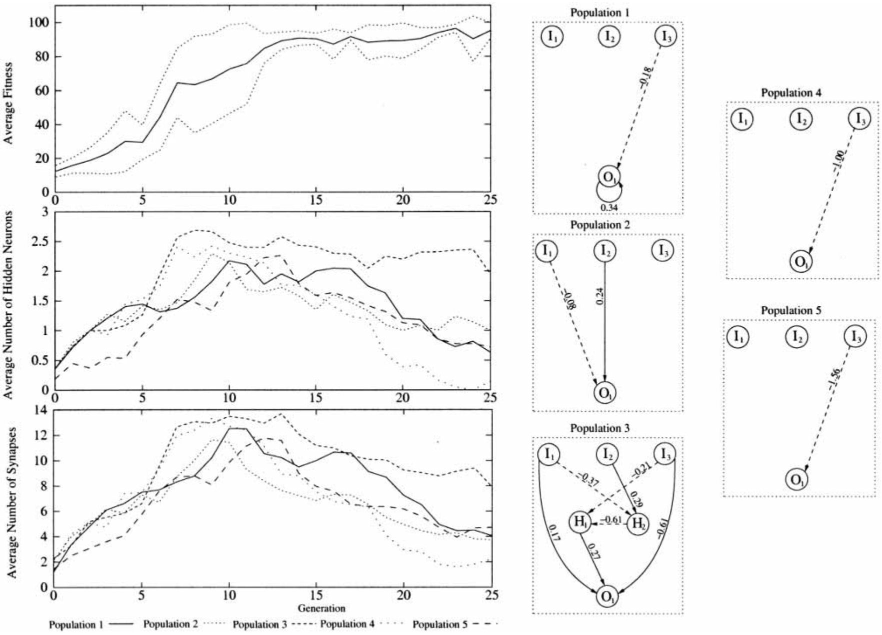

Figure 9. Left: evolution of the average fitness of all populations (top), average number of hidden neurons (middle) and synapses (bottom) for each population. Right: neuro-modules with the best performance after 25 generations.

ring and hold this position. The arm controlled by module 3 starts oscillating and thus initializes a rotation. When the ring starts to rotate the activation of input neuron I_3, which provides information about the hall sensors, changes according to the rotation. Modules 1, 4 and 5 are only driven by this sensor input and, as can be observed, they only move their arms at distinct position changes to maintain a constant angular velocity. A completely different behaviour shows module 2 by holding its arm position if the speed is nearly constant due its connections from I_1 and I_2. Therefore, it has only a passive influence to the overall performance. This experiment demonstrates that different modules with different functions are able to solve co-operatively a given task.

4. Conclusions

We have described an implementation of the *ENS* ^{3} algorithm and its applicability to various optimization and control problems in robotics. By means of four selected applications, we have demonstrated robustness and minimalism of control structures as the main qualities achieved by our artificial life approach to evolutionary robotics.

Considering embodiment as the interdependence of morphology and control (Mautner and Belew 1999, Nolfi and Floreano 2000, Pfeifer 2002), the *ENS* ^{3} was applied to optimize the morphology of a bipedal passive dynamic walking device. The optimized morphology was directly used as a platform for the development of control architectures leading to active dynamic walking. This experiment has shown that effective neuro-modules can be surprisingly small and simultaneously optimally adapted to the morphological properties of the passive dynamic walker (Wischmann and Pasemann 2004), which is also stated elsewhere for other kinds of autonomous robots (Mautner and Belew 1999, Pfeifer 2000, Bongard and Paul 2001).

The co-evolution experiments demonstrated the parallel evolution of several populations. Despite the simple evaluation method, specialization effects and co-operative behaviour of the controllers emerged. The resulting structures were intensively condensed during the evolutionary process, therefore we were able to classify three specialized behaviours and their relationship to the inner structure of the different neuro-modules. The results of this experiment imply that morphology and control can be simultaneously evolved in the same way, since morphology and control have to co-operate optimally to solve a given task.

It is mentioned in Beer (1995) that the performance of evolutionary design does not scale with the size of search space. However, more complex robot platforms and tasks may need larger neuro-modules or, in general, control structures with a larger number of parameters, i.e. a much larger search space has to be explored by evolutionary processes. Within our framework of structure evolution we address this problem with the two introduced neuro-module combination methods, fusion and expansion. Both methods keep the search space as small as possible due to two facts: (1) they start with at least one already existing neuro-module and make use of already established behavioural capabilities; and (2) the *ENS* ^{3}—algorithm enforces development of small structures. Furthermore, one module can be fused with different neuro-modules or can be expanded by adding arbitrary sensor inputs to solve more complex behaviour tasks. In this way, the two methods give the opportunity to generate complex behaviour control in a fast and effective way, in contrast to evolutionary processes that always have to start from scratch.

Module fusion and expansion can be applied using three different techniques to achieve effective combination of several and sometimes contradictory behaviour tasks, which are introduced as restricted, semi-restricted and free combination techniques. The three techniques allow different modes of structure fixation corresponding to different degrees of retention and adaptation of already available behavioural capabilities. Here, the restricted expansion technique was used to extent the MRC, a controller for obstacle avoidance, to a light-seeking module. Although obstacle avoidance and light-seeking are well-studied tasks in robotics, this example is presented as a demonstration of a method. In this case it demonstrates that a neuro-module can extend its behavioural capabilities by plugging in a new structure without any calibration process. From the engineering point of view, this is interesting because complex neuro-modules and robot behaviour can possibly be constructed by directly linking existing neuro-modules.

Furthermore, the results on module expansion show effects of refining capabilities of an original system, instead of simply adding new functionality. Such effects have also been described for biological systems (Arbib and Liaw 1995, Arbib *et al.* 1998). In Arbib and Liaw (1995), investigations on structure and function of biological systems led to general design principles. One of these principles declares that new behavioural features 'often arise as modulators' of existing behavioural capabilities rather than as new independent features. In this way, a successive increase of complex behaviour can be realized by integration of simple sensor qualities. Since the described module fusion and expansion methods effectively utilize those design principles, one may argue that they provide a promising approach to behaviour control. We finally see the presented results as a contribution supporting the view expressed, for example, in Ruppin (2004), that the study of evolved neuro-control for autonomous agents can complement biological findings of neuroscience research.

The benefits of graphical visualization and of plotting neuron activity during robot–environment interaction are demonstrated for each introduced experiment. The data of neuro-modules in action provide first ideas about structure–function relations. In the example of the MRC and its expansion to a light-seeking controller (section 3.1 and 3.2), super-critical self-connections and loops are easily found by the graphical representation of the network. For instance, the odd loop between neuron H_1 and O_2 in the light-seeking controller points to oscillations as applied control technique. The positive self-connections on both output neurons indicate hysteresis effects of the MRC leading to robust obstacle-avoidance behaviour. The relation of neural hysteresis domains and turning angles is demonstrated, likewise the usage of specific period-four attractors for a quasi-halt in front of a light source. Both behaviours can be verified by plotting the neural activities. In addition, in the co-evolution experiment the three specialized behaviours within the five populations are indicated by differences and similarities of the individual neuro-module structures.

Formalization of non-linear control architectures like the MRC demonstrates the advantage of a fast and efficient realization of lesion experiments during robot–environment interaction. Although lesion experiments leading to the MRC were performed on a small structure and rested upon profound knowledge of dynamical systems theory, more efforts have to be undertaken to apply this technique efficiently to larger neuro-modules where complex phenomena are provided by an interplay of many elements. In Husbands *et al.* (1995), an example is given of where a large artificial neural system can be successfully analysed if dynamical features of substructures are clarified. Hence, according to the usage of expansion and fusion methods, large network structures can be understood if the underlying neuro-modules are carefully investigated. Another approach that may give additional guidance to this problem is the functional contribution analysis introduced by Aharonov *et al.* (2003). In this work, the parameters of a fully connected neural network with continuous-time dynamics used for behaviour control are optimized. After optimization these networks undergo a systematically multi-lesion analysis to delineate the 'main functional backbone'. With respect to our lesion tools, those solidified structures may support sufficient formalization of underlying basic control principles.

There is a wide field for applications of formalized neural signal processing principles. For instance, in Manoonpong *et al.* (2004) the neural dynamical Schmitt trigger is used as a filter for auditory signals. An effective neuro-module for the control of a walking machine is presented in Klaassen *et al.* (2004). It is based on non-trivial features of the so-called SO(2) network (Pasemann *et al.* 2003).

The results presented demonstrate that the artificial life approach to evolutionary robotics provides new techniques for a better understanding of the basic mechanisms underlying a behaviour-relevant neural signal processing. Its concentration on the control of systems acting in a sensorimotor loop allows us to find correlations between neural structures with complex dynamical properties and the resulting behaviour of animats acting in a physical or simulated world. Indicating fundamental structures responsible for specific functions may allow us later on to construct relevant control structures by hand. Furthermore, using additional techniques such as lesions of artificial 'brains' and 'electrode' data for the active system may also allow a comparison with neuro-biological findings.

References

Aharonov, R., Segev, L., Meilijson, I., and Ruppin, E., 2003, Localization of function via lesion analysis. *Neural Computation*, **15**: 885–914.

Arbib, W. A., and Liaw, J.-S., 1995, Sensorimotor transformation in the worlds of frogs and robots. *Artificial Intelligence*, **72**: 53–79.

Arbib, W. A., Erdi, P., and Szentagothai, J., 1998, *Neural Organization: Structure, Function, and Dynamics* (Cambridge MA: MIT Press).

Ashby, W. R., 1966, *Design for a Brain*, 2nd edn (London: Chapman & Hall).

Beer, R. D., 1995, An dynamical systems perspective on agent–environment interaction. *Artificial Intelligence*, **72**: 173–215.

Beyer, H. G., 2001, *Theory of Evolution Strategies* (Berlin: Springer).

Bongard, J. C., and Paul, C., 2001, Making evolution an offer it can't refuse: morphology and the extra-dimensional bypass. In J. Kelemen, and P. Sosik (eds) *Advances in Artificial Life, LNAI 2159* (Berlin: Springer), pp. 401–412.

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

Collins, S. H., Wisse, M., and Ruina, A., 2001, A three-dimensional passive-dynamic walking robot with two legs and knees. *International Journal of Robotics Research*, **20**: 607–615.

Dieckmann, U., 1995, The Dynamical Theory of Coevolution, PhD thesis, Leiden University, The Netherlands.

Husbands, P., Harvey, I., and Cliff, D., 1995, Circle in the round: state space attractors for evolved sighted robots. *Robotics and Autonomous Systems*, **15**: 83–106.

Hülse, M., and Pasemann, F., 2002, Dynamical neural Schmitt trigger for robot control. *ICANN 2002, LNCS 2415* (Berlin: Springer), pp. 783–788.

Hülse, M., and Pasemann, F., 2004, Expansion of neuro-modules by structure evolution. In H.-M. Gross, K. Debes, and M.-J. Boehme (eds) *SOAVE2004, Fortschritt-Berichte VDI 10* (VDI Verlag), Germany, pp. 135–145.

Hülse, M., Zahedi, K., and Pasemann, F., 2003, Representing robot–environment interactions by dynamical features of neuro-controllers. In M. Butz, O. Sigaud, and P. Gerard (eds) *Anticipatory Behavior in Adaptive Learning Systems* (Berlin: Springer), pp. 222–242.

Kelso, J. A., 1995, *Dynamic Patterns—The Self-organization of Brain and Behavior* (Cambridge MA: MIT Press).

Klaassen, B., Zahedi, K., and Pasemann, F., 2004, A modular approach to construction and control of walking robots. In *Robotik 2004*, VDI-Berichte 1841 (Düsseldorf: VDI Verlag), Germany, pp. 633–640.

Manoonpong, P., Pasemann, F., and Fischer, J., 2004, Neural processing of auditory–tactile sensor data to perform reactive behavior of walking machines. In *Proceedings of the IEEE MechRob2004*, Aachen, Germany, pp. 189–194.

Mautner, C., and Belew, R. K., 1999, Evolving robot morphology and control. In *Proceedings of the Artificial Life and Robotics Conference (AROB99)*, Oita, Japan.

McGeer, T., 1990, Passive dynamic walking. *International Journal of Robotics Research*, **9**: 62–82.

Michel, O., 1997, Khepera Simulator, Package Version 2.0.

Mondada, F., Franzi, E., and Ienne, P., 1993, Mobile robots miniaturisation: a tool for investigation in control algorithms. In *Proceedings of ISER' 93*, Kyoto, Japan, pp. 501–513.

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

Open Dynamics Engine, 2004, http://www.ode.org

Pasemann, F., 1997, A simple chaotic neuron. *Physica D*, **104**: 205–211.

Pasemann, F., 2002, Complex dynamics and the structure of small neural networks. *Network: Computation in Neural Systems*, **13**: 195–216.

Pasemann, F., Hild, M., and Zahedi, K., 2003, SO(2)-networks as neural oscillators. In *Proceedings of IWANN 2003*, Spain, LNCS 2686 (Berlin: Springer), pp. 144–151.

Pasemann, F., Steinmetz, U., Hülse, M., and Lara, B., 2001, Robot control and the evolution of modular neurodynamics. *Theory in Biosciences*, **120**: 311–326.

Pfeifer, R., 2002, On the role of embodiment in the emergence of cognition: Grey Walter's turtles and beyond. In *EPSRC/BBSRC International Workshop on Biologically-inspired Robotics: The Legacy of W. Grey Walter*, Bristol, pp. 49–57.

Pfeifer, R., and Scheier, C., 2000, *Understanding Intelligence* (Cambridge MA: MIT Press).

Julius Popp, 2003, http://www.sphericalrobots.com

Port, R. F., and van Gelder, T., 1995, *Mind as Motion* (Cambridge MA: MIT Press).

Ruppin, E., 2002, Evolutionary autonomous agents: a neuroscience perspective. *Nature Reviews Neuroscience*, **3**: 132–142.

Walter, G. W., 1953, *The Living Brain* (London: G., Duckworth).

Wiener, N., 1961, *Cybernetics: Or the Control and Communication in the Animal and the Machine*, 2nd edn (Cambridge MA: MIT Press).

Wischmann, S., and Pasemann, F., 2004, From passive to active dynamic 3D bipedal walking—an evolutionary approach. In *Proceedings of 7th International Conference on Climbing and Walking Robots* (CLAWAR2004) (Berlin: Springer), Madrid, Spain, in press.