Expansion of Neuro-Modules by Structure Evolution

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

Abstract

Two methods for the extension of neuro-modules are introduced resulting in a new behavioral functionality. We call them restricted and semi-restricted module expansion. These methods are developed and applied using a modular neuro-dynamics approach to behavior control of autonomous mobile robots. Evolved neuro-controllers which solve an obstacle avoidance task are expanded to show in addition a positive phototropism. All resulting neuro-modules produce a robust light seeking behavior. These neuro-modules use recurrent connectivity structures and non-trivial dynamical features to enable the robot to solve its task. For each neuro-module the structure-function-relation is analyzed. The presented results demonstrate that restricted and semi-restricted expansion are promising methods for generating efficient extensions of recurrent neural networks with additional behavioral functionality.

1 Introduction

Within the framework of behavior based [8, 20] evolutionary robotics [12, 17] and dynamical system theory this article presents a modular neuro-dynamics approach for behavior control of autonomous mobile robots. It is based on the assumption that complex internal dynamics of recurrent neural systems can efficiently solve tasks. The focus lies therefore on behavior control of autonomous robots with recurrent neural networks (RNNs) of general type.

In the following we call such a RNN neuro-controller or neuro-module. Neuro-modules act within a sensory-motor loop, which means they process sensor data to create effective actor signals. Therefore we can define a neuro-module as functional unit. Such modules are described by the number of their input and output neurons, their hidden neurons, and their internal connectivity structure. In our approach the creation of the internal recurrent connectivity structure is established using the evolutionary algorithm ENS^3, which combines structure development and parameter optimization [18].

In this article we demonstrate how already existing neuro-modules can be extended to produce qualitatively new behavior. The extension has to add new behavioral features to a neuro-module, without loosing its essential capabilities. In our context of structure evolution we have developed two methods for this problem: restricted and semi-restricted module expansion [18]. This paper does not only demonstrate the application of these methods. It mainly focuses on methods which allow a clear picture of the underlying internal dynamical mechanisms. The topic of this work is focused on the question: how does a resulting behavior of an autonomous agent emerges from the dynamical features of a given connectivity structure provided by a recurrent neural network. This issue is summarized by the term the structure-function-relation [3]. In evolutionary robotics many techniques are successfully applied creating simple behavior tasks, e.g. obstacle avoidance and light seeking. But less efforts are usually done in clarifying the relation between structure and function, which is here investigated in the domain of the combination of two different behavior tasks. As behavioral tasks we apply: obstacle avoidance and positive phototropism.

We start with the introduction of two obstacle avoidance modules evolved under different boundary conditions (chapter 4). These neuro-modules are used for restricted and semi-restricted module expansion to a light seeking module (chapter 5). Each introduced neuro-module is analyzed according to its structure-function-relation. The conclusions for our module-expansion methods are discussed in chapter 6. First, we briefly introduce the type of neuron, the evolutionary algorithm, the two methods of module expansion and discuss our approach according to related works.

2 The Approach

The neuro-modules created by the evolutionary algorithm are composed of standard neurons with sigmoidal transfer functions. Their time-discrete activation dynamics is given by the equations

$$a_i(t+1) = \Theta_i + \sum_{j=0}^{n} \omega_{ij} \cdot f(a_j(t)) \ , \ i = 1, \ldots, n \ ,$$

where Θ is the bias term, f the sigmoidal transfer function and ω_{ij} the weight of the incoming connection from neuron.

As already mentioned we use the ENS^3 (*evolution of neural systems by stochastic synthesis*) evolutionary algorithm to develop the inner connectivity structure of the neuro-modules [18]. It is primarily designed to develop size and connectivity structure of neuro-controllers and it optimizes parameters like synaptic weights and bias terms at the same time. Starting only with the input and output neurons, nothing else of the network topology is determined, neither the number of hidden neurons nor their connectivity, i.e. self-connections and every kind of recurrences are allowed, as well as excitatory and inhibitory connections. There is one restriction: Because input neurons are only buffering data no backward connections to the input neurons are allowed.

An additional feature of the ENS^3 algorithm is the ability to initialize any evolutionary process with an arbitrary neuro-module. This makes it possible to start a new evolution run with already developed neuro-modules. Furthermore it is possible to set special marks on existing substructures of the module. These marks are taken into account by the variation operator of the evolutionary algorithm. They decide if a structure element (connection or hidden neuron) or a parameter (synaptic weight and bias term) can be deleted and / or modified or not. These features lead to the methods of module expansion: restricted and semi-restricted. *Restricted module expansion*: Structure and parameters of the initial module stay fixed during the evolutionary process. The expanded module will consist of the fixed subnetwork with possible additional connections and additional hidden neurons. *Semi-restricted module expansion*: Only the structure of the initial module stays fixed during evolution. Parameters like synaptic weights and bias terms of the fixed structure can be modified. Therefore the expanded module may consist of the same connectivity structure but with different parameters.

Since behavior of embodied autonomous agents is an interplay of environment, body and control, and stressing embodiment as the interdependence of agent morphology and control, a demand of behavior engineering is that the entire morphology of the robot as well as its control system should be evolved. Progress in this direction is already made [9]. In [7] parameters of a recurrent neural network are optimized by an evolutionary strategy for different morphologies of walking machines.

The results of these experiments identify and qualify constraints and opportunities on the evolution of locomotion. An example of pure morphology evolution by a genetic algorithm is the „box pusher" in [21]. The evolution of morphology plays a major role for bipedal walking [23, 6]. In [23] a minimal control structure for an active bipedal walking device is evolved. The development of this minimal control is realized by utilization of the passive dynamical effects of the morphology. Nevertheless, most of the work is only concentrated on one part: either morphology or control system.

In our approach we develop non-linear control systems for given morphologies and tasks. As mentioned above, the development of control systems involves also the analysis of the structure-function-relation. We use techniques of dynamic systems theory to analyze evolved neuro-modules. This approach is already successfully applied in evolutionary robotics [4, 5, 13, 24]. A complaint to this approach is the common assumption that it will be impossible to understand evolved recurrent systems [14]. Arguing along the line of [5] and [14] there is no evidence to suggest that such will be impenetrable. There are many examples where behavior relevant mechanisms of recurrent neural system can be explicitly described [5, 10].

However, there is no doubt that the number of free parameters of neuro-modules are one crucial issue for the analysis of behavior relevant dynamical effects. The smaller the number of hidden neurons and connections in the neuro-module the simpler the investigation of the structure-function-relation. This problem is handled in our approach by the evolution of minimal neuro-modules. Minimal means a small number of hidden neurons and connections. The structure evolution algorithm ENS^3 enables us to encourage the development of minimal networks by cost terms for neurons and connections. Hence, already during the evolutionary process neuro-modules are developed which consist only of the behavior relevant neural connectivity structure, in contrast to other results [1, 5], where parameter optimization is performed on human defined and fixed neural structures. In [5] the weight parameters of a fully connected and fixed connectivity structure of continuous time recurrent neural networks are optimized. [1] introduces a method called *Functional Contribution Analysis* shrinking a fully connected neural network to its "functional back bone". In [14, 2] one finds examples where the structure of neural control systems is developed without restrictions and successfully applied to several tasks.

3 Experimental Setup

In the following experiments two transfer functions are applied: $tanh(x)$ and $\sigma(x) := \frac{1}{1+e^{-x}}$. The most obvious difference are the upper and lower bounds. The $tanh$ is limited by the open interval $(-1.0, 1.0)$ and σ by the open interval $(0, 1.0)$. For the discussion of the following neuro-modules there are three specific domains important to be distinguished: the (almost) linear domain around zero, and the two saturation domains. Saturation domains are characterized as ranges where large variations in x cause only small changes in the co-domain. These domains approximately begin with $|x| > 3.0$ for $tanh(x)$ and with $|x| > 5.0$ for $\sigma(x)$. In the linear domain small changes in x cause the largest change in the co-domain.

All of the following experiments are done with the Khepera robot [16]. The evaluation of the evolutionary process is done with a 2-dimensional Khepera simulator [15]. All controllers presented in this work are tested with the physical miniature robot Khepera. The scheme of this robot is depicted in Figure 1. Its eight infra-red sensors can be executed in two modes, measuring light intensity (senor values $l_0, l_1, \ldots l_7$) and distances to obstacles (sensor values $d_0, d_1, \ldots, d_7$). Two DC-motors (control signals m_l, m_r) drive the robot. The motors are able to move the left and right wheel forward (positive signals) and backward (negative signals).

In all presented experiments the sensor data are summarized as follows:

$$i_1 := f_{distance\_pre}(\frac{1}{3}(d_0 + d_1 + d2)), \qquad i_2 := f_{distance\_pre}(\frac{1}{3}(d_2 + d_3 + d4)),$$

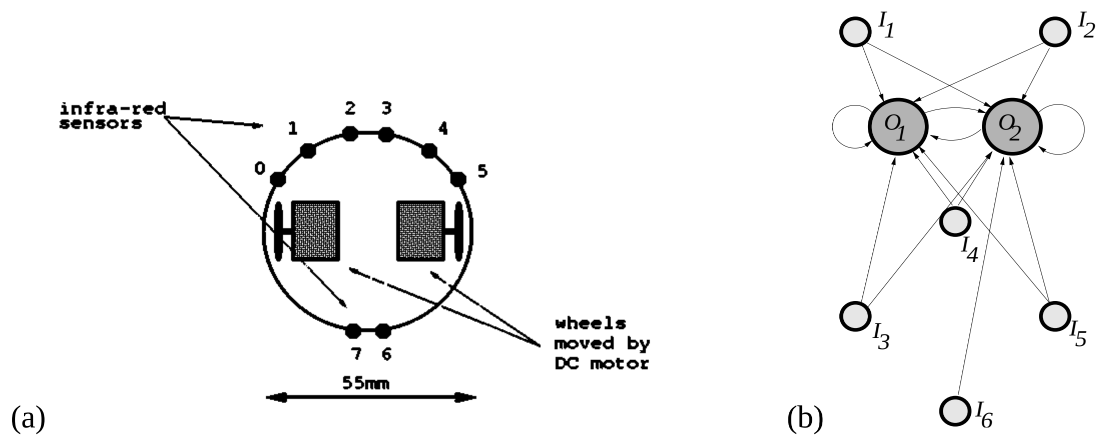

Figure 1. (a) Scheme of the simulated Khepera robot. (b) Sketch of the input-output structure of the neuro-controllers without hidden neurons.

$$i_3 := f_{light\_pre}(\frac{1}{2}(l_0 + l_1)), \qquad i_4 := f_{light\_pre}(\frac{1}{2}(l_2 + l_3)),$$
$$i_5 := f_{light\_pre}(\frac{1}{2}(l_4 + l_5)), \qquad i_6 := f_{light\_pre}(\frac{1}{2}(l_6 + l_7)).$$

The values $i_1, \ldots, i_6$ are the inputs to the corresponding input neurons $I_1, \ldots, I_6$ of the neuro-module. For this reason we call the functions $f_{distance\_pre}(x)$ and $f_{light\_pre}(x)$ the pre-processing for proximity and light input neurons. According to the used transfer function of the neuro-module distinct pre-processings are applied (detailed description in chapter 4 and 5).

A neuro-module without hidden neurons is sketched in Figure 1 to present the general input-output-structure of the following neuro-module. In addition to the six input neurons the module has two output neurons O_1, O_2. These outputs deliver the signals o_1, o_2 used to calculate the control signals m_l and m_r for the left and right wheel. This calculation is called post-processing:

$$m_l := f_{left\_post}(o_1, o_2), \qquad m_r := f_{right\_post}(o_1, o_2).$$

As for the pre-processing mentioned before in the following experiments different post-processings are applied.

4 Neuro-Modules for Obstacle Avoidance

The first neuro-module for obstacle avoidance is a generalization of former evolution experiments [10] (Figure 2). In the following we call this neuro-controller $tanh$-module. This neuro-module contains the essential structural and dynamical features, which were analyzed in [10, 11]. Therefore it is an excellent candidate for module expansion experiments. Here $tanh$ is used as transfer function and therefore a pre-processing is applied, which delivers linear input signals in the range from -1.0 to $+1.0$:

$$f_{distance\_pre}(x) := \frac{2x}{1024} - 1.0 \quad with \quad x \in [0, 1024].$$

The output signals o_1 and o_2 directly drive the left and right motor:

$$f_{left\_post}(o_1, o_2) := v \cdot o_1, \qquad f_{right\_post}(o_1, o_2) := v \cdot o_2,$$

where v ($1.0 \leq v \leq 10.0$) is a speed factor [15] and is for all experiments 5.0.

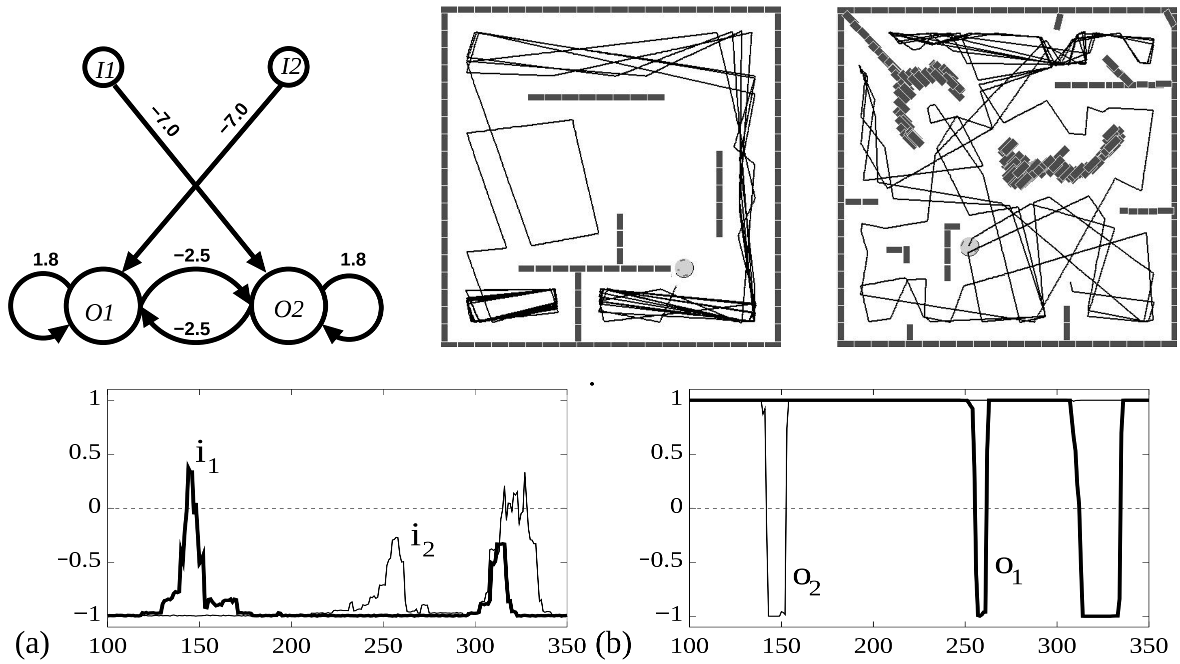

Figure 2. **Upper figures:** *The structure of the $tanh$-module which solves an obstacle avoidance task. The plotted path show the resulting behavior.* **Lower figures:** *Plot of input (a) and output values (b) of the network during the interaction of the robot in its environment. This sample shows a turn to the left and right.*

As it is indicated by figure 2 the robot shows a robust obstacle avoidance. It does not collide with objects and is able to escape from sharp corners and dead-ends.

As it was analyzed in [10] the robust behavior emerges from an interplay of three different hysteresis domains due to the over-critical self-connections at output neuron O_1 and O_2 and the even loop between them. The three hysteresis domains produce in specific situations (obstacle on the left, the right and sharp corners) a robust turning with appropriate angles. During the robot-environment interaction the activation values of both output neurons mainly remain in the two saturation domains (see Figure 2).

The second neuro-controller for obstacle avoidance uses the standard sigmoid $\sigma(x)$ as transfer function. We will call them σ-module in the following. According to the problem of handicapped navigation possibilities with only positive control signals a special post-processing is implemented. Here, the two output neurons are functionally decomposed. The left output neuron O_1 controls the speed and the O_2 the turning angle of robot's movement. This is formalized as follows:

$$f_{left\_post}(o_1, o_2) := v \cdot (o_1 - (2o_2 - 1.0)), \quad f_{right\_post}(o_1, o_2) := v \cdot (o_1 + (2o_2 - 1.0)),$$

where v is the already introduced speed factor. Because of the usage of $\sigma(x)$ as transfer function we change the pre-processing in such a manner that input values are in the interval $[0, +1.0]$: $f_{distance\_pre} := \frac{x}{1024}$ with $x \in [0, 1024]$, where 0 means no obstacle is detected and 1.0 refers to a collision.

Figure 3 shows the network structure and the resulting behavior. Robots controlled by this network are able to leave sharp corners and dead ends. If no obstacle is detected the robots show slightly curved movements. Also seen in Figure 3 is the time evolution of input and output signals during the robot-environment interaction. For the case that no obstacle is detected (e.g. between step 60 and 100) there is an output configuration, refering to maximum speed ($o_1 \approx 1.0$) with slightly curved forward movement ($o_2 \approx 0.53$). If an obstacle on the right side is detected the output signal o_2 increases while

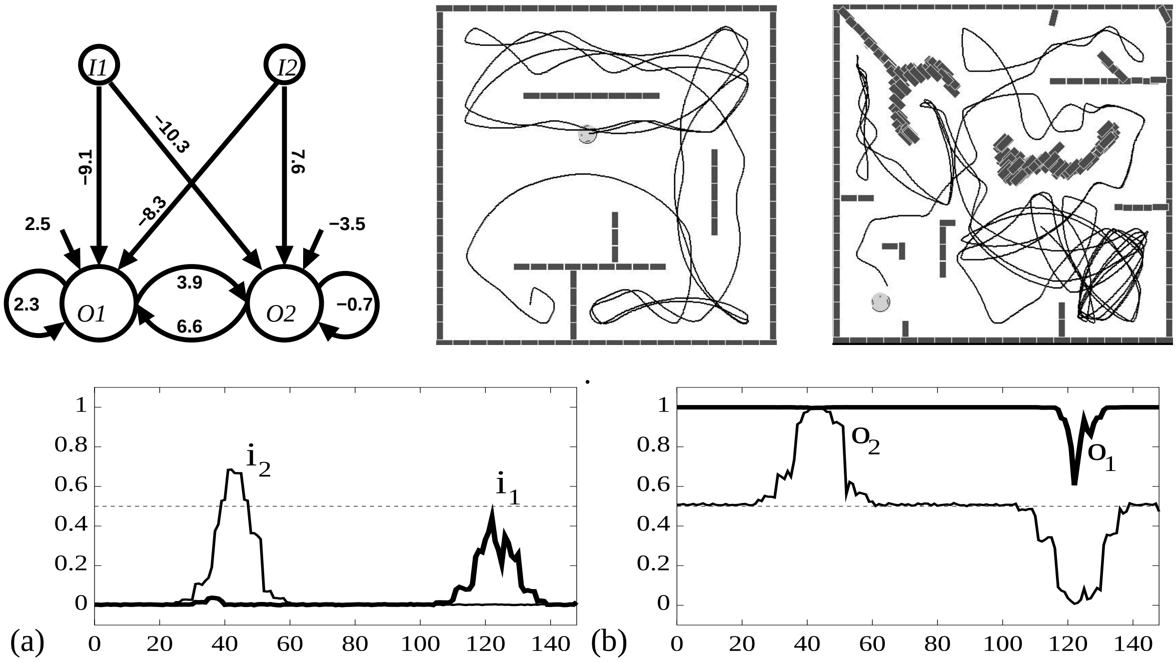

Figure 3. **Upper figures:** *Structure of the σ-module where output neurons serve as controller for turning angle and motor speed.* **Lower Figures:** *Plot of input (a) and output values (b) of the network during the robot-environment interaction. This sample shows a turn to the left and right.*

output signal o_1 remains constant at its maximum. This creates a left turn with maximal speed. In contrast a right turn is done with reduced speed due to the decrease of the output signal o_1, if the activation of the left input neuron increases. The reduced speed is essential for escaping from critical situations, e.g. sharp corners or dead-ends. Critical situations are characterized by a high activation of both input neurons. These produce a decrease of the output signal o_1 and o_2 to zero, which refers to a turn with no forward speed, i.e. a turn on the spot.

5 Expansion of Neuro-Modules for Light Seeking Behavior

In the following the two previous described neuro-modules for obstacle avoidance are extended to solve a light seeking task. Light seeking is defined as searching and following a light source and stay in front of it while avoiding collisions with other objects. Therefore we use the $tanh$- and σ-module as initial configuration and expand them with a positive light tropism. For expansion we apply the restricted method for the $tanh$-module and the semi-restricted method for the $\sigma(x)$-module. To compare the resulting light seeking modules we use the same pre-processing for the light values for each module:

$$f_{light\_pre}(x) := \frac{500 - x}{450} \quad with \quad x \in [50, 500].$$

This mapping delivers linear input signals, where a value of zero refers to darkness and 1.0 represents brightness.

First we introduce the experiment of the $tanh$-module expansion by the restricted method. The evolutionary algorithm is then applied to the initial configuration composed of the $tanh$-module with the four additional light sensor inputs I_3, I_4, I_5 and I_6. Structure and parameters of the $tanh$-module are fixed but additional internal neurons and connections may be added and varied.

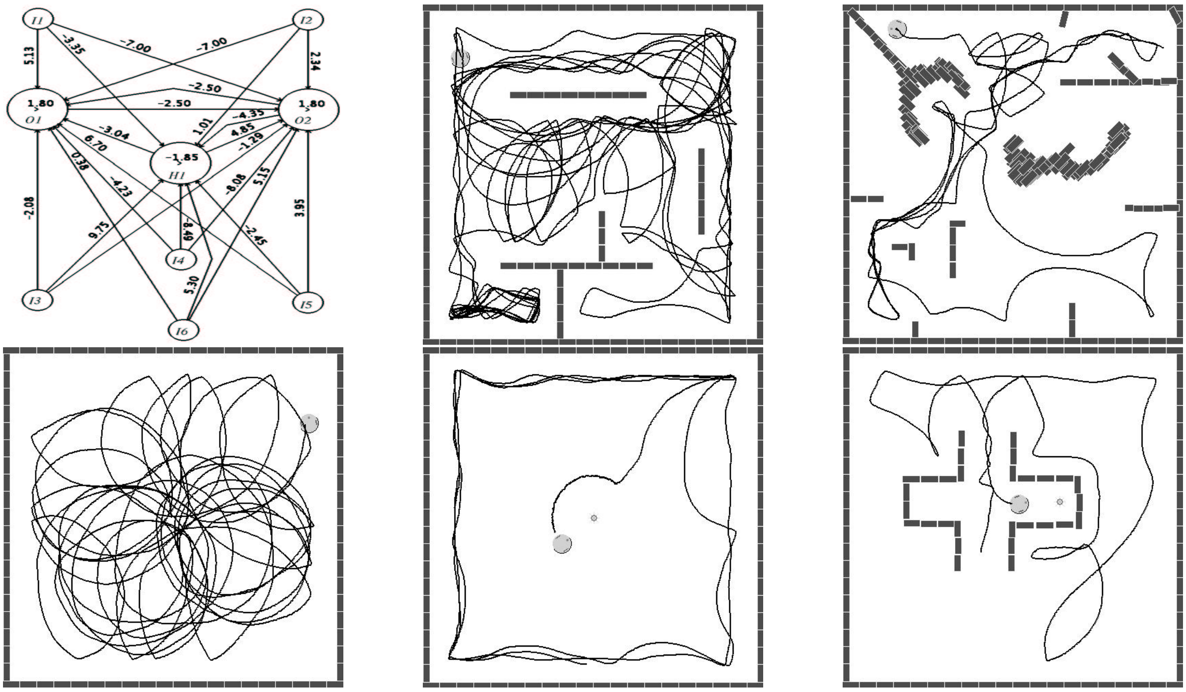

Figure 4. The expanded $tanh$-module for light seeking behavior and its resulting behavior.

During the evolutionary process many solutions to the light seeking problem appear. All of the observed solutions have the property that the robot does not really come to a halt in front of a light source, instead it circles around the light source in an oscillatory motion. One of this resulting modules and its behavior is shown in figure 4. For details of the inner structure we give the weight matrix of this controller:

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

The plotted trajectories show that the robot is still able to avoid collision with objects and is searching and following a light source. But it lost the capability of leaving sharp corners. It can also be observed that the robot moves in circular paths in contrast to the straight movements of the underlying $tanh$-module for pure obstacle avoidance. The expanded $tanh$-module has additional connections to the two output neurons coming from the light sensors as well as from the proximity sensors. An additional hidden neuron H_1 has also connections to the outputs and incoming connections from all input and one output neuron.

A 2-neuron configuration like that of H_1 and O_4 is denoted a *chaotic 2-module* [19] because it allows all kinds of complex dynamics like oscillations and chaos. The plots in Figure 5 of the neuron output signals give an impression how different oscillations become active under specific input value configurations. For instance if the robot detects no light and no obstacle a period-5 oscillation can be observed. This oscillation produces a mean output of 0.6 for O_1 and 0.7 for O_2 (Table 1) which is related to the slightly curved movement. In the case that a light source is in front of the robot and no obstacle is detected the module produces a period-4 oscillation which generates the quasi-halt in front of the light source.

In the second experiment the σ-module is expand by the semi-restricted method. Like in the experiment before: the evolutionary algorithm is applied to the initial configuration composed of the σ-module with four additional light sensors. But here only the structure of the σ-module is fixed, i.e. all parameter are open for variation while already existing hidden neurons and connections can not

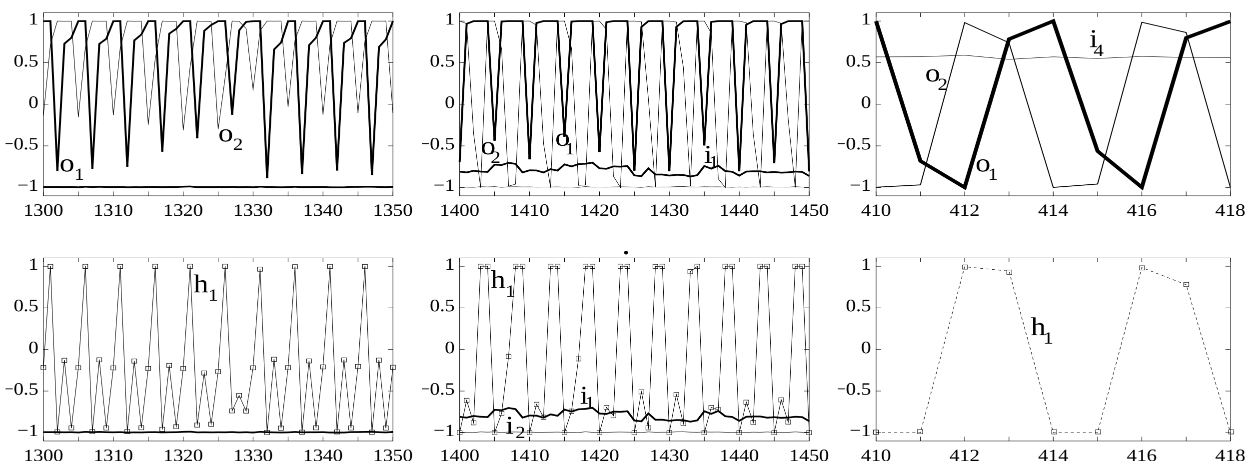

Figure 5. Output signals o_1 and o_2 of the two output neurons (upper plots) and the hidden neuron H_1 (lower plots). **Left:** *all sensor input values are minimal, e.g. no light source and no obstacle is detected.* **Middle:** *The robot moves in the dark while it detects an obstacle on the left, which is indicated by the high input value i_1 (thick line).* **Right:** *The robot is in front of a light source ($i_4 \approx 0.55$). The mean output of each output neuron is zero which produces the „standing" in front of a light.*

Table 1. The average value of neuron outputs for specific input configurations. The resulting behavior 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.

env. situation	i_1	i_2	i_3	i_4	i_5	i_6	o_1	o_2
empty dark space	-1.0	-1.0	0.0	0.0	0.0	0.0	0.6	0.7
obstacle left	0.2	-1.0	0.0	0.0	0.0	0.0	1.0	-1.0
obstacle right	-0.9	0.7	0.0	0.0	0.0	0.0	-1.0	1.0
light in front	-1.0	-0.9	0.1	0.6	0.0	0.0	0.0	0.0

be deleted. One example of a resulting neuro-controller with minimal structure and its behavior in different environments is shown Figure 6. A robust light seeking behavior can be seen. The robot gets stuck in sharp corners under certain circumstances. All synaptic weights and bias terms of the initial σ-module are modified, no hidden neurons are added and each light sensor input neuron has only one connection to the output neurons. An interesting fact is that the resulting structure is symmetric according to the synaptic weight parameters.

For the case that no obstacle and no light is detected: the output neuron O_1 generates output signals close to its maximum $+1.0$ (saturation domain) and output neuron O_2 has output values around $+0.5$ (linear domain). This is related to forward movement with maximum speed. If the robot detects a light source e.g. on the left side: the increasing output value of input neuron I_3 forces a turning to the left. The same holds for the light sensor on the left and at the rear. The robot turns to the light as long left and right light values i_3 and i_5 are equal. If i_3 and i_5 are equal: their influence on the turn neuron O_2 is compensate and the robot moves toward the light source. But the smaller the distance between light source and robot the larger the input value i_4. At a specific distance the output of O_1 becomes zero and the robot comes to a halt in front of the light. Conflicts between approaching a light and avoiding collisions are solved by the additional connections from the proximity sensors to the output

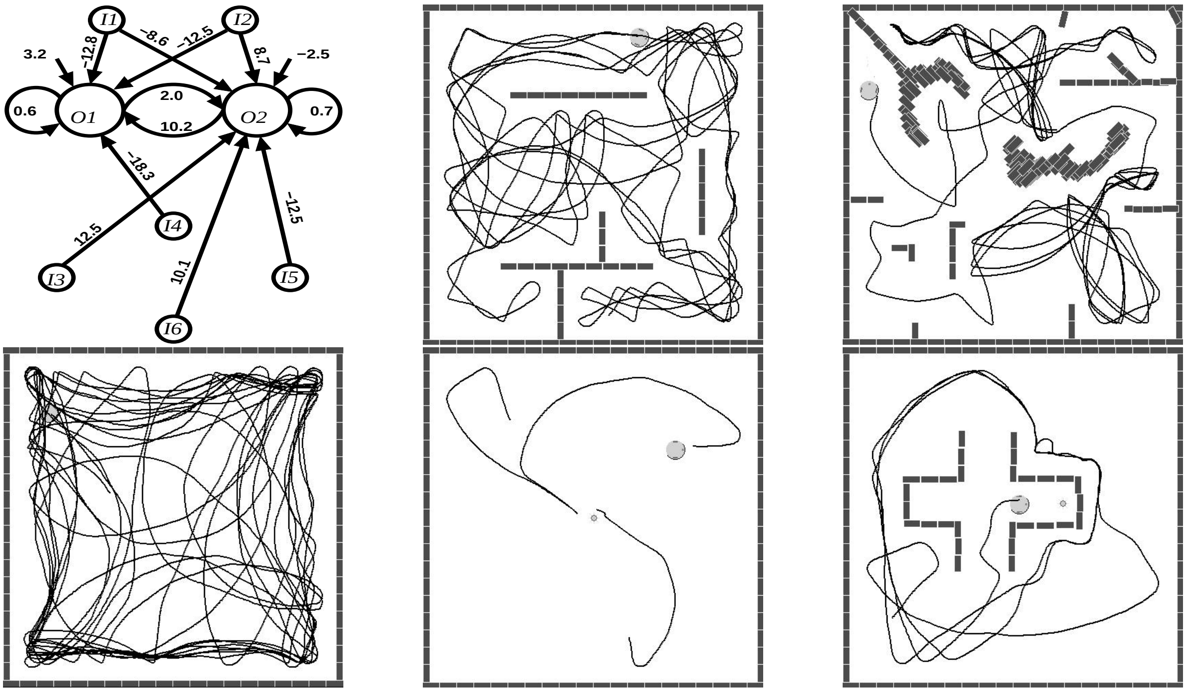

Figure 6. The expanded σ-module and its resulting behavior.

neuron O_1. All proximity sensors, but only one light sensor, have influence on the speed neuron.

6 Conclusion

Within the framework of the modular neuro-dynamics approach we introduced two methods for module expansion: restricted and semi-restricted expansion. Both methods show the capability for optimal extending or combining features of already evolved modules with new behavioral requirements.

The extended σ-module is an example where already established control techniques are used, and new sensors and requirements are added. The structure has only four new connections coming from the four light sensor input neurons. Its output neurons mainly work in the same domains like the underlying σ-module for pure obstacle avoidance. Output neuron O_1 works in its two saturation domains and output neuron O_2 in its linear domain. The extended σ-module shows qualitatively no new control technique. The orientation to the light is realized in the same way as the orientation away from obstacles.

The extended $tanh$-module is an example for a suitable modification of dynamical properties, if the already existing features are inapplicable to include new behavioral functionality. Looking at the dynamical features of neuro-modules the extended $tanh$-module shows an increase of complexity. The new hidden neuron is connected to output neuron O_2 in such a way that it can generate different oscillations. It is still an open question how and which oscillation becomes active under certain circumstances but we could already show that at least a period 4 and 5 attractor are behavior relevant. Oscillations compensate the maximal or minimal output signals of the output neurons O_1 and O_2. These values, generated by the hysteresis domains in the underlying $tanh$-module, realize a straight forward movement and robust obstacle avoidance behavior, but they are unfavorable for a smooth orientation to a light source and totally unfit for coming to a halt in front of it. The emergent oscillations of the extended $tanh$-module generate: (1) curved paths for exploration, (2) smooth orientation to a light and (3) a quasi-halt in front of a light source. These are qualitative new behavioral capabilities which are generated by new dynamical features - oscillations instead of hysteresis effects.

The extended $tanh$-module is also an example of how an original behavior can be modified although the original structure of the neuro-module is still present. The obstacle avoidance behavior of the pure $tanh$-module is obviously different from the obstacle avoidance behavior of the extended $tanh$-module (see Figure 2 and 4). The robot controlled by the extended module moves in irregular curved paths, which is an advantage for light seeking. In contrast to that, a robot controlled by the pure $tanh$-module moves straight and has constant turning angles. Such effects of refinement the capabilities of the original system (instead of simply adding new functionality) are described in [3] for biological system. So we finally see the presented results as an additional contribution to the view expressed e.g. in [22], where it is argued that the study of neurally-driven evolved autonomous agents is a promising methodology in neuroscience research complementing the biological findings.

References

[1] Aharonov, R., Segev, L., Meilijson, I., and Ruppin, E. (2003), Localization of Function Via Lesion Analysis, *Neural Computation*,15, pp. 885–914.

[2] Angeline, P.J., Saunders, G.B., and Pollack J.B. (1994), An Evolutionary Algorithm that Evolves Recurrent Neural Networks, *IEEE Transactions on Neural Networks*,5, pp. 54–65.

[3] Arbib, M. A. (1998), A Functional Overview, In: Arbib, M. A., Erdi, P., Szentagothai, J., *Neural Organization: Structure, Function, and Dynamics*, Cambridge, MA: The MIT Press.

[4] Beer, R., and Gallagher, J. C. (1992), Evolving dynamic neural networks for adaptative behavior, *Adaptive Behavior*, 1(1), pp. 91 – 122.

[5] Beer, R. (2003), The Dynamics of Active Categorical Percetion in an Evolved Model Agent, *Adaptive Behavior*, 11 (4), pp. 209 – 244.

[6] Bongard, J.C., and Paul, C. (2001), Making Evolution an Offer It Can't Refuse: Morphology and the Extradimensional Bypass, *Proceedings of the Sixth European Conference on Artificial Life*, pp. 401–411.

[7] Bongard, J.C., and Pfeifer, R. (2002), A Method for Isolating Morphological Effects on Evolved Behavior, *Proceedings of 7th International Conference of Adaptive Behavior, SAB'02*, Cambridge, MA: MIT Press/Bradford Books, pp. 305–311.

[8] Brooks, R. (1991), Intelligence Without Reason, *Proceedings of 12th Int. Joint Conf. on Artificial Intelligence*, Sydney, Australia, August 1991, pp. 569–595.

[9] Harvey, I., Husbands, P., and Cliff, D. (1994), Seeing the light: Artifical Evolution, real vision., *Proceedings of 3rd International Conference of Adaptive Behavior, SAB'94*, Cambridge, MA: MIT Press/Bradford Books, pp. 392 – 401.

[10] Hülse, M., and Pasemann, F. (2002) Dynamical Neural Schmitt Trigger for Robot Control, J. R. Dorronsoro(Ed.): ICANN 2002, LNCS 2415, Springer, pp. 783–788.

[11] Hülse, M., Pasemann, F., and Zahedi, K. (2002) Representing Robot-Environment Interactions by Dynamical Features of Neuro-Controllers, In: M. Butz et al. (Eds.) *Anticipatory Behavior in Adaptive Learning Systems*, LNAI 2684, pp. 222 – 242.

[12] Husbands, P., Harvey, I. and Cliff, D. (1993), Analysing recurrent dynamical networks evolved for robot control, Proceedings of the Third IEE International Conference on Artificial Neural Networks (ANN93), IEE Press, pp.158–162.

[13] Husbands, P., Harvey, I., and Cliff, D. (1995), Circle in the round: State space attractors for evolved sighted robots, *Robotics and Autonomous Systems*, 15, pp. 83 – 106.

[14] Husbands, P., Harvey, I., Cliff, D., Miller, G. (1997), Artificial Evolution: A New Path for Artifical Intelligence?, *BRAIN AND COGNITION*, 34, pp. 130 – 159.

[15] Michel, O., *Khepera Simulator* Package version 2.0: Freeware mobile robot simulator written at the University of Nice Sophia-Antipolis by Oliver Michel. Downloadable from the World Wide Web at `http://wwwi3s.unice.fr/∼om/khep-sim.html`

[16] Mondada, F., Franzi, E., and Ienne, P. (1993), Mobile robots miniturization: a tool for investigation in control algorithms, in: *Proceedings of ISER' 93*, Kyoto.

[17] Nolfi, S., and Floreano, D. (2000) *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines* MIT Press, Cambridge.

[18] Pasemann, F., Steinmetz, U., Hülse, M., and Lara, B. (2001), Robot Control and the Evolution of Modular Neurodynamics, *Theory in Biosciences*, 120, 311–326.

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

[20] Pfeifer, R., and Scheier, C. (2000), *Understanding Intelligence*, MIT Press, Cambridge.

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

[22] Ruppin, E., (2002), Evolutionary Autonomous Agents: A Neuroscience Perspective, *Nature Reviews Neuroscience*, 3(2), (2002), 132–142.

[23] Wischmann, S., and Pasemann, F. (2004), From Passive to Active Dynamic 3D Bipedal Walking - An Evolutionary Approach, , To appear in *Proceedings of the 7th International Conference on Climbing and Walking Robots*.

[24] Yamauchi, B., and Beer, R. (1994), Integrating reactive, sequential, and learning behavior using dynamical neural networks, *Proceedings of 3rd International Conference of Adaptive Behavior, SAB'94*, Cambridge, MA: MIT Press/Bradford Books, pp. 382 – 391.