Dynamic Regulation of Hebb Learning by Artificial Neuromodulators in Mobile Autonomous Robots

Helmut A. Mayer
Department of Computer Science
University of Salzburg
A–5020 Salzburg, Austria
helmut@cosy.sbg.ac.at

Gerald Wiesbauer
Department of Computer Science
University of Salzburg
A–5020 Salzburg, Austria
wiesi@cosy.sbg.ac.at

Abstract – *We investigate key components of a dynamic neurocontroller changing its internal structure enabling "lifetime" learning of a mobile autonomous robot. The behavioral change of the robot is linked to inputs from the environment that cause the emission of artificial neuromodulators (ANMs) in the robot's neurocontroller. In its simplest form an outside teacher (human or machine) constantly evaluates the robot's actions by transmitting positive or negative feedback signals to the robot initiating the internal changes. The focus of investigations is put on the mechanisms of the interaction of teaching input and structural changes. A well–known concept for this interaction is Hebbian learning, which is regulated by ANMs in the presented approach. In extension to related work in evolutionary robotics (ER), we analyze important details of robotic (ontogenetic) learning by experiments measuring the ability of robots to learn simple tasks in a simulated environment without employing evolution. Specifically, we are interested in the comparison of Hebb learning variants, and the crucial question of the correct interpretation of reward or punishment signals by the robot.*

Keywords: Mobile Autonomous Robots, Dynamic Neurocontrollers, Artificial Neuromodulators, Self-learning.

1 Introduction

Today, a variety of control techniques are implemented for mobile autonomous robots, e.g., fuzzy control, machine learning systems, or artificial neural networks. Some of these systems have reached an impressive level of performance, however, most of these systems are not adaptive in the sense that they cannot change their behavior by feedback from the environment. Although, a human observer of these systems might get the impression that the robots can adapt to different situations, they can only adapt to situations the control program is aware of in advance.

State of the art is that control programs for the robots are no longer constructed by humans but by computers employing evolutionary methods [1]. Still, in order to evolve working "robot brains" all possible scenarios have to be presented to the system in advance. Hence, the control system will deal sufficiently with most situations it has been trained with, but it is mostly not able to deal with new, unknown situations, and it also cannot adapt itself to these during the "lifetime" of the robot. When evolving robotic neurocontrollers, learning is taking place in a generational time frame (*phylogenetic* learning).

Obviously, the main problem of (most) current control systems is that they cannot "reprogram" themselves during their exploration of the environment. This static behavior of an artificial structure is the most fundamental difference to *Biological Neural Networks* (BNNs) exhibiting highly dynamic properties not only throughout their lifetime, but also within very short time spans of activity [2].

Dynamic changes in a neurocontroller may be induced employing *Reinforcement Learning* (RL) techniques [3] enabling *ontogenetic* learning, i.e., the robot's brain (consequently, its behavior) is shaped during exploration of the environment. A popular RL method applicable to neurocontrollers is *Temporal Difference* (TD) learning [4]. With this method the neurocontroller is not generating motor signals driven by sensor input, but evaluates potential (motor) actions. Actions are presented as an additional input, and a single output neuron predicts the value of a potential action. Learning is driven by the difference of predictions in consecutive time steps (temporal difference) and scalar feedback signals (reward or punishment) from the environment or a teacher.

Technically, learning in TD neurocontrollers is implemented by the common *Back–propagation* method. TD learning is purely ontogenetic and does not alter the structure of the neurocontroller. A biologically more plausible method to achieve a combination of phylogenetic and ontogenetic learning (as seen in nature) are evolved network structures, whose parameters are altered by *Artificial Neuromodulators* (ANMs) [5, 6]. The ANMs influence learning by defining the type of *Hebb* learning based on the combination of modulators received by each neuron [5], or by specifically changing neurons' activation functions [6].

As a consequence, very complex interactions can be observed in ANM neurocontrollers that make interpretations of the internal mechanisms nearly impossible. Hence, in this work we are concerned with neurocontrollers with pre–defined, simple modulator diffusion models and single learning rules for the whole network. The biologically plausible learning mechanism is based on the *Hebbian* learn rule promoting self–organization of the neurocontroller.

Especially, in feed–forward networks Hebb learning has the properties of implicit calculation of the *Principal Components* of the input data [7]. The *Principal Component Analysis* (PCA) is a statistical method linearly transforming a sample of points in an n–dimensional space such that the variance of the components in the new coordinate system are extremal. Components (or features) with low variance contribute little to the information content of the sample, hence they may be neglected in order to compress the input data. PCA networks can be used in signal classification, feature extraction, and data compression [7]. In the context of this work the feature extraction property could be useful in order to detect structures in the sensor signals of the robot, which might convey relevant information at the current time.

As pointed out above this work is concerned with a deeper analysis of the prerequisites of successful learning in mobile autonomous robot based on the ANM paradigm. A better understanding of the processes and model parameters leading to a well–performing robot could not only improve the speed and the quality of ontogenetic learning, but could also be beneficial in evolutionary robotics by reducing the search space of all possible arrangements.

2 Ontogenetic Learning

In the works referred to above the parameters for the dynamic changes in the robot's neurocontroller have been evolved. Hence, the final neurocontroller intrinsically deployed the correct types and doses of ANMs so as to achieve the desired behavior of th robot. If we want to teach the robot during its lifetime (on–line), we have to know which ANMs cause the robot to change or enforce its behavior. More specifically, the reaction of a neuron receiving a modulator must be correctly implemented. E.g., in BNNs *Dopamine* acts as a "reward" hormone, which is emitted as a consequence to positive feedback [2]. Though, we can easily define such a reward ANM in the artificial brain, it is not clear which reaction (in our system Hebb learning variants) has to be chosen in order to link the rewarded behavior with the future behavior of the robot.

We want to emphasize that the neurocontroller we are going to present is enabling ontogenetic learning by feedback signals from the environment (mediated by ANMs). Though, being a classical reinforcement learning approach, the neurocontroller's architecture is different from RL methods, as it does not evaluate policies (potential actions), but represents the basic architecture of neurocontrollers employed in ER approaches. The robot's sensor signals at the input layer of the network generate motor signals at the output layer. In addition to pure phylogenetic learning achieved by evolving the structure of the robotic brain, ER researchers also suggested evolution of learning rules enabling lifetime learning [8]. The latter system is learning constantly, while in our approach learning is triggered by pre–defined events or an outside teacher, i.e., there may be only short time periods, where learning is activated or deactivated. Evidently, this should assist the robot in finding interesting subspaces of the input signal space, where it can extract the most useful information to learn the given task. The basic architecture of the neurocontroller employed in the following experiments is shown in Figure 1.

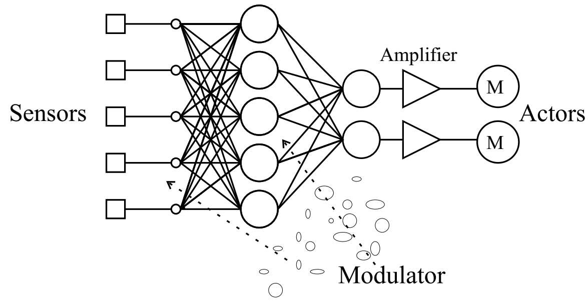

Figure 1: Basic architecture of a dynamic neurocontroller with artificial neuromodulators.

The plasticity of the robotic brain (induced by the ANMs) also allows for adaptions even when the environment or the physical appearance of the robot (e.g., sensor loss) changes after it has successfully learned a task. In order to investigate the prerequisites for successful ontogenetic learning employing a dynamic neurocontroller, we set up two simple tasks: i) the robot should learn to avoid the walls of a rectangular arena (wall avoidance) ii) the robot is taught to move to a spot in the arena that has a specific odor (spot finding).

In the wall avoidance task the environmental feedback is given by a bumper sensor, which is activated, when the robot touches the wall of the arena. The sensor signal is fed into an input neuron, which emits an ANM signalling "pain" inside the robotic brain. With spot searching the smell of the spot is proportional to the distance of the robot to the spot. The nose of the robot is connected to an input neuron emitting a "joy" modulator, which should enforce the robot to move towards the spot. The presented experiments have two fundamental differences: wall avoidance should be learned by giving negative feedback for short periods of time (wall contact), while spot finding should be achieved by positive feedback given over long time periods (robot smelling the spot).

Results of various experiments should assist to resolve a number of design questions, namely, the rates of emitted modulators, and the reaction to reception of a modulator (actually changing network parameters via Hebbian learning).

3 Experimental Setup

All experiments are conducted in a Java simulator designed and constructed by the authors allowing real time and soft time simulation. The latter enables to perform experiments, where many hours of robot action have to be simulated, in a few seconds or minutes. The cylindrical robot shown in Figure 2 is equipped with four distance sensors (front, back, left, right), and a contact sensor (wall avoidance), or a nose sensor (spot finding).

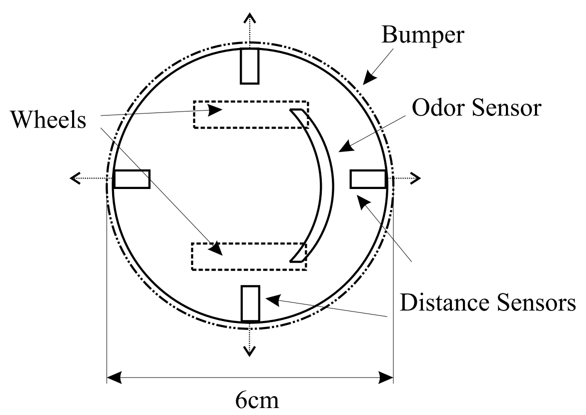

Figure 2: The cylindrical robot.

The software sensor simulates a nonlinear, noise–free, real device measuring the reflection of a physical signal emitted exactly in direction of the line from robot center to the sensor positioned at the perimeter of the robot.

The neurocontroller is a standard *One–Hidden Layer* network (five hidden neurons) composed of neurons with logistic activation function. Each sensor is associated with an input neuron, whose activation determine the signals at the two output neurons (left and right motor). Each neuron is capable of receiving and reacting to the emitted ANM. In case of the wall avoidance experiment, a single type of ANM (pain) is diffused by the contact sensor neuron, when the robot touches the wall. If the ANM is emitted all neurons immediately are able to receive the modulator (in the next time step) by a given reception rate. The reception of a modulator triggers the unsupervised learning process. The dose (measured in mole) of the receipted modulator is directly proportional to the learning parameter η for basic Hebbian learning:

$$\Delta w_{j,i} = \eta a_i a_j, \tag{1}$$

where a_i, a_j are the pre–, and postsynaptic activations, respectively, of the neurons connected by the weighted link. Note that the weight change $\Delta w_{j,i}$ only takes place, when an ANM is received by a neuron. By setting the emission rate larger than the (summed) reception rate of all neurons, it is possible to easily introduce a kind of short–term memory, as it takes a number of time steps (simulation cycles) to fully absorb the modulator.

3.1 Hebbian Learn Rules

Table 1 gives the definition of the four Hebbian learning rules that are used in the experiments.

Hebb (H)	$\Delta w_{j,i} = \eta\, a_i\, a_j$
Anti–Hebb (AH)	$\Delta w_{j,i} = -\eta\, a_i\, a_j$
Covariance Hebb (CH)	$\Delta w_{j,i} = \eta\,(a_i - \overline{a_i})(a_j - \overline{a_j})$
Covariance Anti-Hebb (CAH)	$\Delta w_{j,i} = -\eta\,(a_i - \overline{a_i})(a_j - \overline{a_j})$

Table 1: Variants of Hebbian learn rules.

The parameters $\overline{a_i}$ and $\overline{a_j}$ are the pre–, and postsynaptic mean activations, respectively, being defined as the running mean of the neuron's activation from $t = 0$ ("birth") to the current time t.

3.2 The Wall Avoidance Experiment

In this experiment the robot is placed in a rectangular arena (1.05×0.70 m) and should learn to avoid wall contact. We performed experiments with 500 simulated robots initialized with different random weights and biases from the interval $[-1.0, 1.0]$. The learning behavior is evaluated by a *Learn Ability* calculated in the following way:

1. Every robot is placed into each of the four corners (in a distance of ten cm to the walls) and then moves freely (without learning) for ten minutes. Throughout these 40 minutes we measure the time t_{pre} it is in contact with the wall.

2. The robot is placed in the upper left corner with activated learning (modulators are diffused by the

contact sensor neuron). From now on the robot has two hours to learn the task.

3. After the learning procedure the robot is tested in the same way as described in 1 measuring the wall contact time t_{post}.

The learn ability L_{WA} is defined as

$$L_{WA} = \frac{t_{pre} - t_{post}}{t_{pre} + t_{post}}. \quad (2)$$

We study the impact of different learning rules on the learning behavior of the robot using the mean learn ability $\overline{L}$ of the robots. Note that a number of robots avoid the wall without any learning, which we labelled *Genius*, as they perfectly master the task right from the time of "birth". Genius robots are not considered for calculation of the mean learn ability. The learn ability L is 1.0, if the robot has learned the task perfectly, e.g., never touches the wall after training. An $L > 0.0$ indicates an improvement after learning, while an $L < 0.0$ is the sign of a negative effect of training, i.e., the robot exhibits a worse behavior.

The learn ability is influenced by the dose d of modulator, which is emitted by the contact sensor neuron at wall contact. The emission rate is set to 12 mole per second. Each neuron in the network is able to receipt this modulator. The reception rate is set such that the complete amount of modulator diffused in one time step is consumed by all neurons at equal parts in the next time step. The consumed dose is directly mapped to the learn rate η of the neurons' pre–synaptic links, e.g., if a neuron consumes 1.0 mole of the modulator $\eta = 1.0$. Note that in this setting the ANM concept only mediates start and stop of Hebbian learning with a specific learn rate. While this procedure has appealing biological analogies, it could be equally implemented in a simple algorithmic way. However, changes in the emission and/or reception rate would immediately introduce complex temporal interactions of feedback signals and weight changes.

3.3 The Spot Finding Experiment

Again, the robot is placed into the rectangular arena, but this time it should learn to move towards a circular spot in the arena, which can be smelled by the robot. The contact sensor is replaced by a nose sensor detecting odors in an angular range of 90 degrees around the front distance sensor. The (virtual) odorous spot is a circle with a diameter of twelve cm. The nose sensor delivers a signal proportional to the distance, and the associated neuron emits the *Joy* modulator in a binary manner (odor yes/no). The emission rate of the nose neuron is set to 0.6 mole per second.

The evaluation of the robots is performed as follows:

1. Each robot is placed in the upper left corner (ten cm to the walls), while the spot is placed in the upper right corner. During the next 30 minutes we measure the time t_{pre} the robot is inside the spot.

2. The robot is placed in the left upper corner, now with learning activated for two hours.

3. Finally, the robot is tested in the same way as described in 1 measuring the time inside the spot t_{post}.

As the measured times are now indicating wanted behavior, the learn ability $L_{SF} = -L_{WA}$ (Equation 2).

We also measure the mean distance to the spot before ($\overline{s}_{pre}$) and after ($\overline{s}_{post}$) training. Only robots having ever been inside the spot are taken into account for the calculation of the mean values. A specific problem with the mean distance is a robot positioning itself near the spot right at the wall with a tendency to "run away" from the spot. Though, the robot actually is not learning the intended behavior, in this case the mean distance would indicate a robot attracted by the spot.

4 Results

Employing negative Hebb learning as the reaction to the received modulator in the wall avoidance experiments a number of robots is able to avoid the wall after a few collisions. Other robots (each "born" with a different random brain) take some minutes (real–time simulation) to learn the task, while a few never learn it, and sometimes always remain in contact with the wall. All robots learning the task develop an intuitively expected behavior of slowing down, when approaching a wall, and starting to turn away from the wall, then accelerating into "open terrain". A typical motion trail of a robot having quickly learned the task can be seen in Figure 3.

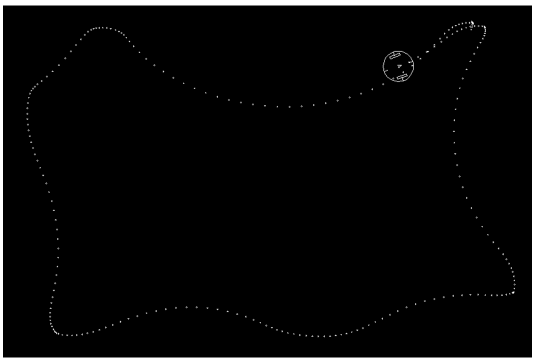

Figure 3: A typical motion trail of a wall avoiding robot after ontogenetic learning.

In a number of experiments (Table 2) we expectedly saw that the type of learning reaction has a dramatic influence on the robot's learning ability. We also noticed that successful learning is greatly influenced by a detail neglected in most previous work on Hebb learning. Usually, the learn ability of the robot is considerably improved, when the bias values of a neuron are not subjected to Hebb learning, i.e., they remain constant. Consequently, we also present results comparing fixed with learned bias values in Table 2.

	AH, yes	AH, no	H, no	CAH, no	CH, no
L = 1	40	52	61	52	73
L > 0	27	161	55	180	138
L = 0	0	0	0	8	4
L < 0	235	83	205	62	97
Genius	198	204	179	198	188
$\overline{L}$	-0.222	0.417	0.028	0.457	0.451
$\overline{d}$	64880	12177	54260	15984	23219

Table 2: Learn abilities L of 500 wall avoiding robots using different learn rules with (yes) or without (no) bias learning. The contact sensor neuron is synaptically connected to the hidden layer.

Interestingly, there is a clear difference between changing the bias values according to Hebb's rule, or keeping them fixed. Training the initially random bias values results in a much worse learning ability $\overline{L}$ (averaged on all trained robots) than excluding the bias from training (genius robots never touch a wall, hence, they are never trained, and do note contribute to $\overline{L}$).

When employing AH learning, the weights are decreased in each learn step being triggered by wall contact. As a consequence, a trained robot has mostly (large) negative weights. If it approaches the wall of the arena, a strong signal is generated by one of the distance sensors, which leads to low activation (close to zero) of the hidden neurons. Then, the activity of the motor neurons is only determined by its bias values. If the bias values are fixed and different for the two motor neurons, the robot will turn, which is what it should learn to do near the wall. However, if the bias values are subjected to AH learning as well, they will mostly become negative resulting in zero activation of the motor neurons, actually moving the robot straight with full reverse speed.

The essence of these considerations is that in this case the learn ability of the robot is only dependent on its fixed bias values given at "birth". AH learning more and more reveals the basic "character" of the robot, but it does not change this character. Thus, learn ability is only determined by traits already existing at the time of the robot's "birth". The results in Table 3 confirm this observation, but they also show that CAH Learning does not depend on the initial bias values.

With bias values fixed to 0.0 AH learning achieves a much smaller learn ability than with fixed random values, as the key to successful learning in this setting is a difference in the bias values of the output neurons

	AH	CAH
L = 1	48	72
L > 0	77	176
L = 0	0	3
L < 0	184	67
genius	191	184
$\overline{L}$	0.022	0.485
$\overline{d}$	47554	15784

Table 3: Learn abilities L of 500 wall avoiding robots with bias values fixed to 0.0.

(enabling turning behavior).

CH learning does not only not exhibit this dependency, but also is successful regardless of the positive or negative variety. There are a number of possible explanations to this behavior. This Hebb variant allows weight changes in both directions even in the same learn step (simulation cycle). The mean activations represent a very basic form of memory, which makes learning dependent on time, or in other words on the robot's age. Learning is also dependent on the mobility of the robot. A robot mostly staying in a certain area of the arena, will process similar input signals most of the time leading to a convergence of the mean activations. If the same robot moves to another area, the difference of input signals to the mean activations commanding the actual weight change will be larger (stronger learning) than for a more mobile robot. Putting all together and considering that learning only takes place at certain points in time (wall contact) the complexity of this still simple Hebb variant becomes obvious.

Naturally, the dose of the emitted modulator contributes to the learning process of the robot. Thus, we measured how the learn ability of the robots is influenced by the modulator dose. Comparing AH and CAH learning (fixed random bias) in Figure 4 reveals interesting properties.

The increase of the rate of modulator emission is balanced with a proportional increase of the reception rate. Hence, the complete dose of modulator emitted in a time step is consumed in the next triggering the given type of learning. In case of the CH rule the weight changes are in both directions. Increasingly strong learn signals lead to a complete perturbation of the network weights, i.e., a new random network. With the simple wall avoidance task it might only take a few wall contacts until a genius contributing to improved learn ability is found. High modulator doses in combination with AH learning make the robots more and more insensitive to the input signals, as even weak sensor signals lead to deactivation of the hidden layer. Hence, the robot no longer switches between a behavior close to the wall and a different one in "open terrain". If the robot moves in a rather straight manner, wall contacts are inevitable, and

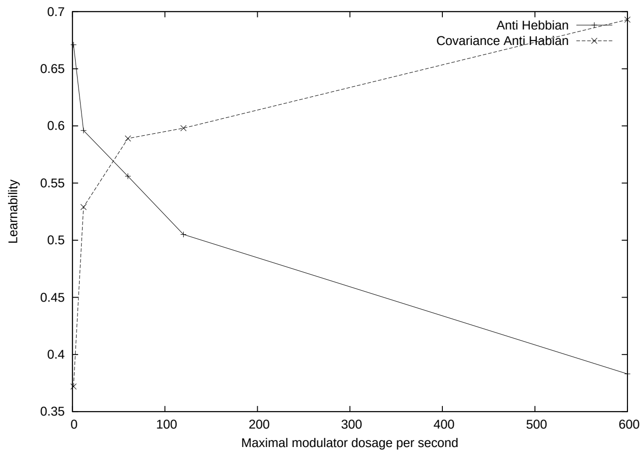

Figure 4: The dependence of the learn ability on the modulator rate for wall avoidance.

the strong learning signals further enforce the singular behavior.

The results of the spot finding experiment are summarized in Table 4 showing that no learn rule is able to enforce the wanted behavior. Still the best is AH learning, but on average it also makes the robots to avoid the spot instead of approaching the target, which can also be seen with the mean distances $\overline{s}$. If a robot successfully finds the spot, it has a rotating behavior, when it does not smell the spot, and a more straight motion towards the spot, if the nose detects the odor. In case of AH learning similar arguments as for the wall avoiding experiments apply. In case of an activated nose neuron, the hidden layer is shut down, which makes the motor action dependent on the bias values. Only, if these are approximately equal and cause a motion towards the spot, a robot is able to potentially learn the spot finding behavior. Again, AH learning only reveals the innate character of a robot.

	AH, Yes	AH, No	H, Yes	H, No	CH, No
$L = 1$	47	95	4	6	5
$L > 0$	1	3	1	0	1
$L = 0$	0	1	0	1	0
$L < 0$	224	187	194	218	166
$\overline{L}$	-0.639	-0.310	-0.951	-0.942	-0.929
$\overline{s}_{pre}$	0.203	0.283	0.109	0.109	0.122
$\overline{s}_{post}$	0.484	0.363	0.472	0.249	0.539
Inside	272	286	199	225	172

Table 4: Learn abilities L and mean distances $\overline{s}$ of 500 spot finding robots using different learn rules with (yes) or without (no) bias learning. "Inside" indicates the number of robots ever within the odorous spot.

5 Summary

The results show that ontogenetic learning of mobile autonomous robots with neurocontrollers regulated by external feedback mediated by ANMs is sufficient to teach robots simple tasks. However, we have found that the learning ability of the robots is dependent on parameters that are randomly assigned at the "birth" of the robot. The crucial question to be addressed in future research is, if there exists an unsupervised learning method allowing the robot to correctly interpret the feedback signals so as to learn the appropriate behavior. In conventional reinforcement learning the problem of interpretation is solved by assigning values to actions, while in this work we investigate the classical neural mapping of sensor to motor (action) signals. Assuming that Hebbian learning plays an important role in BNNs, the finding that a sensor–motor neurocontroller cannot be generally trained by unsupervised learning, would possibly imply that biological systems rely on action–value networks as suggested by various researchers.

References

[1] Stefano Nolfi and Dario Floreano. *Evolutionary Robotics – The Biology, Intelligence, and Technology of Self–Organizing Machines.* MIT Press, 2000.

[2] G. M. Shepherd. *Neurobiology.* Oxford University Press, 3rd edition, 1994.

[3] Richad S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction.* MIT Press, Cambridge MA, 1998.

[4] William D. Smart and Leslie Pack Kaelbling. Reinforcement learning for robot control. In *Mobile Robots XVI (Proc. SPIE 4573)*, 2001.

[5] Akio Ishiguro, Siji Tokura, Toshiyuki Kondo, Yoshiki Uchikawa, and Peter Eggenberger. Reduction of the Gap between Simulated and Real Environments in Evolutionary Robotics: A Dynamically–Rearranging Neural Network Approach. In *IEEE Systems, Man, and Cybernetics Conference*, pages III – 239–244. IEEE, October 1999.

[6] Tom M. C. Smith and A. Philippides. Nitric Oxide Signalling in Real and Artificial Neural Networks. *British Telecom Technology Journal*, 18(4):140–149, October 2000.

[7] E. Oja, J. Karhunen, L. Wang, and R. Vigario. Principal and independent components in neural networks - recent developments. Technical report, Helsinki University of Technology, 1996.

[8] Dario Floreano and Francesco Mondada. Evolutionary neurocontrollers for autonomous mobile robots. *Neural Networks*, 11(1998):1461–1478, 1998.