
Evolution of Adaptive Synapses: Robots with Fast Adaptive Behavior in New Environments

Joseba Urzelai

joseba.urzelai@elca.ch

ELCA Informatique SA, Av. de la Harpe 22-24, CH-1000 Lausanne 13, Switzerland

Dario Floreano

Dario.Floreano@epfl.ch

Evolutionary and Adaptive Systems, Institute of Robotics, Swiss Federal Institute of Technology (EPFL), CH-1015 Lausanne, Switzerland

Abstract

This paper is concerned with adaptation capabilities of evolved neural controllers. We propose to evolve mechanisms for parameter self-organization instead of evolving the parameters themselves. The method consists of encoding a set of local adaptation rules that synapses follow while the robot freely moves in the environment. In the experiments presented here, the performance of the robot is measured in environments that are different in significant ways from those used during evolution. The results show that evolutionary adaptive controllers solve the task much faster and better than evolutionary standard fixed-weight controllers, that the method scales up well to large architectures, and that evolutionary adaptive controllers can adapt to environmental changes that involve new sensory characteristics (including transfer from simulation to reality and across different robotic platforms) and new spatial relationships.

Keywords

Evolutionary neural networks, adaptive synapses, compact genetic encoding, new environments, adaptive behavior, cross-platform adaptation.

1 Evolution and Adaptation

The situated nature of Evolutionary Robotics (Nolfi and Floreano, 2000) is such that often evolved controllers find surprisingly simple, yet efficient, solutions that capitalize upon unexpected invariants of the interaction between the robot and its environment. For example, a robot evolved for the ability to discriminate between shapes can do so without resorting to expensive image processing techniques by simply checking the correlated activity of two receptors located in strategic positions on the retinal surface (Harvey et al., 1994). Analogously, a robot evolved for finding a hidden location can display performances similar to those obtained by rats trained under the same conditions, without resorting to complex environmental representations, by using simple sensory-motor sequences that exploit geometric invariants of the environment (Lund and Miglino, 1998). The remarkable simplicity¹ and efficiency of these solutions is a clear advantage for fast and real-time operation required from autonomous robots, but it raises the issue of robustness when environmental conditions change after evolutionary training. Environmental changes can also be a problem for other approaches (e.g., conventional programming, learning) to the extent that the sources of change have not

¹This does not imply that evolutionary approaches are restricted to forms of reactive intelligence; see for example Floreano and Mondada (1996a).

been considered during system design, but they are even more so for evolved systems because these systems often rely on environmental aspects that are not predictable by an external observer. Therefore, it is difficult to predict whether an evolved system will withstand certain types of change.

Environmental change can be induced by several factors such as modifications of the sensory appearance of objects (e.g., different light conditions), changes in sensor response, re-arrangement of environment layout, transfer from simulated to physical robots, and transfer across different robotic platforms.

Some authors have suggested improving the robustness of evolved systems by adding noise (Miglino et al., 1996; Jakobi, 1997) and by evaluating individuals in several different environments (Thompson, 1998). However, both techniques imply that one knows in advance what makes the evolved solution brittle in the face of future changes in order to choose a suitable type of noise and of environmental variability during evolutionary training. In Grefenstette and Ramsey (1992), the authors proposed an approach to continuous learning in changing environments known as *Anytime Learning*. The approach consists of continuously testing new strategies against a simulation model of the task environment and includes a monitor that can dynamically modify the simulation model based on its observations of the environment. The learning system is composed of a genetically represented rule base, which can be updated on the basis of the performance results. However, this method requires considerable domain knowledge in order to identify those aspects of the environment that are initially uncertain or subject to change and to design the policies for updating the learning model.

Another approach consists of combining standard evolution and lifetime learning algorithms (Hinton and Nowlan, 1987; Ackley and Littman, 1992; Todd and Miller, 1991; Gruau and Whitley, 1993; Nolfi and Parisi, 1997; Nolfi, 1999). This strategy not only can improve the search properties of artificial evolution but can also make the controller more robust to changes that occur faster than the evolutionary time scale (i.e., changes that occur during the life of an individual) (Nolfi and Floreano, 1999). This is typically achieved by evolving neural controllers that learn with an off-the-shelf algorithm, such as reinforcement learning or back-propagation, starting from synaptic weights specified on the genetic string of the individual (Ackley and Littman, 1992; Nolfi et al., 1994). Only initial synaptic weights are evolved. A limitation of this approach is the Baldwin Effect, whereby the evolutionary costs associated with learning give a selective advantage to the genetic assimilation of learned properties and, consequently, reduce the plasticity of the system over time (Mayley, 1996). Another limitation is the fact that the adaptation process is constrained by the type of learning algorithm chosen by the experimenter, which may not be the most suitable for the actual situation.

In a pioneering work, Chalmers (1990) proposed evolving a function capable of changing the synaptic weights of a randomly initialized feed-forward neural network using only information local to each synapse and a set of training signals. He showed that evolution of such self-organizing mechanisms was capable of rediscovering the Delta Rule (Widrow and Hoff, 1999) and of coping with a variety of different environmental features as they arise. Following this direction, in previous work we have suggested evolving *the adaptive characteristics* of a controller instead of combining evolution with off-the-shelf learning algorithms (Floreano and Mondada, 1996b; Floreano and Mondada, 1998; Floreano and Nolfi, 1997). The method consists of encoding on the genotype a set of four local Hebb rules for each synapse, but *not the synaptic weights*, and letting these synapses use these rules to adapt their weights online starting always from random values at the beginning of the life. Since the synaptic weights are not en-

coded on the genetic string, there cannot be genetic assimilation of abilities developed during life (i.e., there cannot be Baldwin effect). In other words, these controllers can rely less on genetically-inherited invariants and must develop on-the-fly the connection weights necessary to achieve the task. At the same time, the evolutionary cost of adaptation (i.e., the time and energy spent to adapt reduced fitness of the individual) implicitly puts pressure for the generation of fast-adaptive controllers.

In this paper, we extend previous work by using a much more compact genetic representation of adaptive neurocontrollers and systematically compare its performance to direct encoding of synaptic weights and to encoding of noisy synapses. Through a set of new experiments we show that: (i) evolutionary adaptive controllers solve a robotic task much faster and better than evolutionary standard (non-adaptive) controllers; (ii) the method scales up well to large architectures; (iii) the evolved adaptive characteristics affect the behavior of the robot in several ways; (iv) evolved adaptive controllers can adapt to sensory, motor, and environmental changes that take place after evolutionary training.

2 Encoding Mechanisms of Adaptation

The method proposed here consists of evolving mechanisms for parameter self-organization, instead of the parameters themselves as in conventional approaches. The artificial chromosome encodes a set of four modification rules for each component (parameter) of the neural network (components can be individual synapses or groups of synapses that converge towards the same neuron), but not values of components (synaptic strengths of the network). Whenever an artificial chromosome is decoded into a neural controller, the synaptic strengths are always set to small random values. This means that robots will display random actions at the beginning of their “lives” both at the first and last generation. While the robot moves, synapses are allowed to change their values every 100 ms (the time necessary for a full sensory-motor loop on the physical robot) using the genetically specified rules. Synaptic change occurs online and without external supervision and reinforcement signals during the whole life² of the individual. The fitness function is evaluated along the whole duration of each individual’s life. This introduces an implicit learning cost (Mayley, 1996) that gives selective advantage to individuals that can adapt faster and therefore gain more fitness points. At the end of the life, the final synaptic strengths are not “written back” into the artificial chromosome.³

We have selected four types of adaptation rules (Figure 1) to be encoded on the artificial chromosome. The choice has been based on neurophysiological findings and on computational constraints of local adaptation. In other words, these rules capture some of the most common mechanisms of local synaptic adaptation found in the nervous systems of mammals (Willshaw and Dayan, 1990). These rules have been mathematically formulated in order to satisfy the following constraints. Synaptic strength cannot grow indefinitely but is kept in the range $[0, 1]$ by means of a self-limiting mechanism depending on synaptic strength itself. Because of this self-limiting factor, a synapse cannot change sign, which is genetically specified, but only strength. Each synaptic weight w_{ij} is randomly initialized (based on a uniform distribution in the interval $[0, 0.1]$) at the beginning of the individual’s life and can be updated after each sensory-

²The number of sensory-motor cycles allowed to evaluate each individual chromosome.

³In other words, we use Darwinian evolution instead of Lamarckian evolution where the effects of learning are encoded in the artificial chromosome. See Yamamoto et al. (1999) for an experimental comparison between these two types of evolution in changing environments.

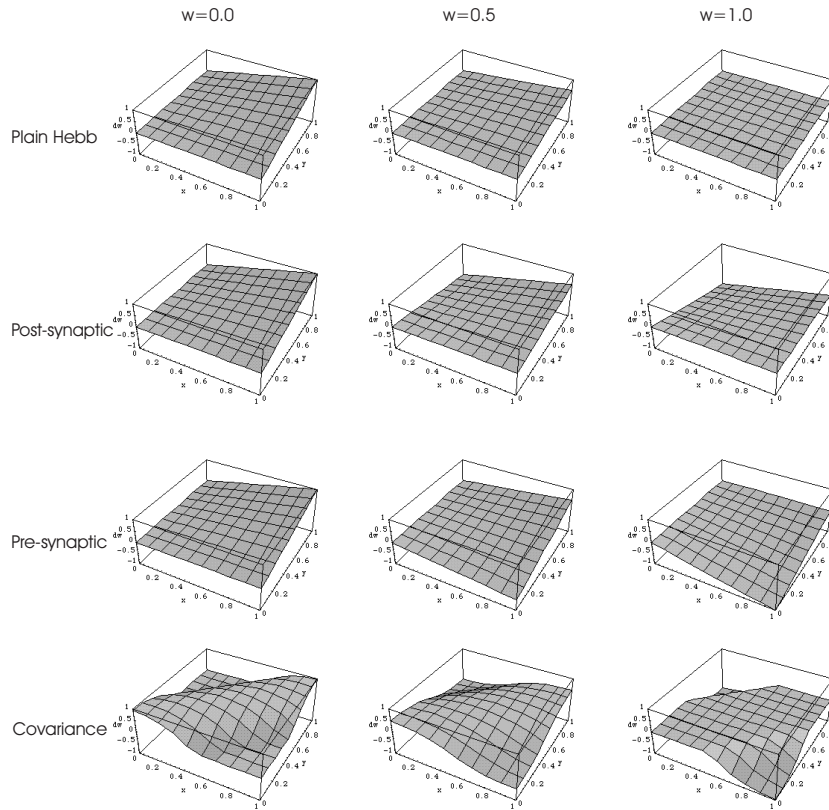


Figure 1: Synaptic change for each of the four Hebb rules. Notice that this is the *amount of change* Δw added to the synapses, not the synaptic strength. Each graph indicates the amount of change as a function of pre-synaptic x and post-synaptic y activity. The amount of change also depends on the current strength w of the synapse so that synapses are always bound between 0 and 1. Three graphs are shown for each rule, in the case of current strength, 0.0, 0.5, and 1.0, respectively.

motor cycle (100 ms),

$$w_{ij}^t = w_{ij}^{t-1} + \eta \Delta w_{ij},$$

where $0.0 < \eta < 1.0$ is the learning rate and Δw_{ij} is one of the four adaptation rules specified in the genotype:⁴

1. *Plain Hebb rule* can only strengthen the synapse proportionally to the correlated activity of the pre-synaptic neuron x_j and of the post-synaptic neuron y_i (Hebb, 1949; Kelso et al., 1986).

$$\Delta w_{ij} = (1 - w_{ij}) x_j y_i \tag{1}$$

2. *Post-synaptic rule* behaves as the plain Hebb rule, but it also weakens the synapse when the post-synaptic node is active but the pre-synaptic is not (Stent, 1973; Singer, 1987).

$$\Delta w_{ij} = w_{ij} (-1 + x_j) y_i + (1 - w_{ij}) x_j y_i \tag{2}$$

⁴These four rules can be applied to different synapses within the same network.

3. *Pre-synaptic rule* is complementary to the post-synaptic rule: weakening occurs when the pre-synaptic unit is active but the post-synaptic is not.

$$\Delta w_{ij} = w_{ij}x_j(-1 + y_i) + (1 - w_{ij})x_jy_i \quad (3)$$

4. *Covariance rule* strengthens the synapse whenever the difference between the activations of the two neurons is less than half their maximum activity, otherwise the synapse is weakened. In other words, this rule makes the synapse stronger when the two neurons have similar activity levels, otherwise it makes the synapse weaker (Stanton and Sejnowski, 1989).

$$\Delta w_{ij} = \begin{cases} (1 - w_{ij})\mathcal{F}(x_j, y_i) & \text{if } \mathcal{F}(x_j, y_i) > 0 \\ (w_{ij})\mathcal{F}(x_j, y_i) & \text{otherwise} \end{cases} \quad (4)$$

where $\mathcal{F}(x_j, y_i) = \tanh(4(1 - |x_j - y_i|) - 2)$ is a measure of the difference between the pre-synaptic and post-synaptic activity. $\mathcal{F}(x_j, y_i) > 0$ if the difference is bigger or equal to 0.5 (half the maximum node activation) and $\mathcal{F}(x_j, y_i) < 0$ if the difference is smaller than 0.5.

The adaptation rules, but not the synaptic strengths, are encoded in the artificial chromosome. Genetic encoding refers to the way in which a neural controller is mapped onto a bit string representing the artificial chromosome of an individual. A chromosome is composed of a series of genes. A gene is a set of bits that encodes a given feature of the neural controller. We consider two aspects of genetic encoding: the *feature* level and the *properties* of that feature.

Features We consider two levels of feature encoding: synapses and nodes (see top of Figure 2). *Synapse encoding* refers to the case where a gene encodes the properties of all individual synapses. In this case, the artificial chromosome will have as many genes as synapses in the network. This type of encoding is also known as *direct encoding* and is rather common in works that combine evolutionary computation and neural networks (Yao, 1993). *Node encoding* instead refers to the case where a gene encodes the properties of individual nodes. In that case, all the incoming synapses to that node will share the same properties specified for that node (except for the sign of the traversing signal, which is a property of the pre-synaptic node). In this case the artificial chromosome will have as many genes as nodes in the network. Synapse encoding allows a detailed definition of the neural network, but for a fully connected network of N neurons, the genetic length is proportional to N^2 . On the contrary, node encoding requires a much shorter genetic length (proportional to N), but it allows only a rough definition of the network because all incoming synapses to a node share the same properties.

Properties Regardless of the feature level chosen (synapses or nodes), each gene is composed of five bits that represent the properties of the corresponding feature. We consider three types of properties (see bottom of Figure 2). For all three types, the first bit always represents the sign of the signal traveling outward (either through the synapse in the case of synapse encoding or through the outgoing axon in the case of node encoding). The remaining four bits can encode the following properties:

1. *Genetically determined synapses*: 4 bits encode the synaptic strength. This value is constant during the life of the individual. This is the conventional way of evolving neural networks (Yao, 1993).

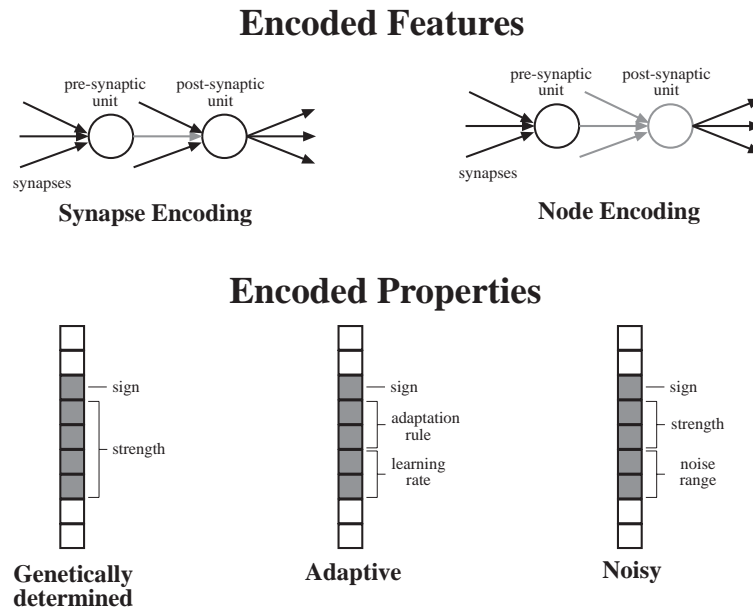


Figure 2: Different types of genetic encoding. (Top) Two feature levels. The genetic string can either encode the properties of each individual synapse in the network (synapse encoding) or encode the properties of an entire node and its synapses (node encoding). In the latter case, the encoded properties are applied to all incoming synapses to that node. Node encoding results in shorter genetic strings. (Bottom) Three types of properties. Genetically determined properties specify the connection sign and strengths of synapses. Adaptive properties specify the sign, the adaptation rule, and the adaptation rule of the synapses. Noisy properties specify the sign, weight strength, and a noise range that is continuously applied to the synapse. Properties are applicable to both synapse and node encoding, but in the latter case, all incoming synapses will have the same properties.

2. *Adaptive synapses*: 2 bits encode the 4 adaptation rules described above and 2 bits the corresponding learning rate (0.0, 0.3, 0.6, 0.9). Synaptic weights are always randomly initialized at the beginning of an individual's life and then updated on-line every 100 ms according to their own modification rule while the individual interacts with the environment. This is the core of the methodology proposed in this paper, that is, evolving the mechanisms of on-line self-organization of a neural controller.
3. *Noisy synapses*: 2 bits encode the weight strength and 2 bits a noise range (± 0.0 , ± 0.3 , ± 0.6 , ± 0.9). The synaptic strength is genetically determined at birth, but a random value extracted from the noise range is freshly computed and added every 100 ms while the individual interacts with the environment. This is a control condition to check whether the effects of random variations are equal or different from the effects induced by the adaptation rules.

In previous work, Floreano and Mondada (1998) used only synapse encoding and showed that evolution of adaptive synapses produced in fewer generations better con-



Figure 3: A mobile robot Khepera equipped with a vision module can gain fitness points by staying on the gray area only when the light is on. The light is normally off, but it can be switched on if the robot passes over the black area positioned on the other side of the arena. The robot can detect ambient light and wall color, but not the color of the floor.

trollers than evolution of genetically determined synapses for simple reactive navigation. Here, we wish to go one step further and investigate whether adaptive synapses can use node encoding, which is a much more compact representation. In the set of experiments presented in the following sections, we will compare evolution of adaptive synapses with node encoding to other types of genetic encoding for a sequential task that is complex enough to require non-trivial solutions. The results show that node encoding of adaptive synapses can develop more complex abilities, scales up to larger neural networks, and produces neural controllers that remain adaptive to several sources of change after evolutionary training.

3 A Sequential Task: The “Light-Switching” Problem

A mobile robot Khepera equipped with a vision module is positioned in the rectangular environment shown in Figure 3. A light bulb is attached on one side of the environment. This light is normally off, but it can be switched on when the robot passes over a black-painted area on the opposite side of the environment. A black stripe is painted on the wall over the light-switch area. Each individual of the population is tested on the same robot, one at a time, for 500 sensory motor cycles, each cycle lasting 100 ms. At the beginning of an individual’s life, the robot is positioned at a random position and orientation and the light is off.

The fitness function is given by the number of sensory motor cycles spent by the robot on the gray area beneath the light bulb *when the light is on* and divided by the total number of cycles available (500). In order to maximize this fitness function, the robot should find the light-switch area, go there in order to switch the light on, and then move towards the light as soon as possible, and stand on the gray area. Since this sequence of actions takes time (several sensory motor cycles), the fitness of a robot will never be 1.0. Also, a robot that cannot manage to complete the entire sequence will be scored with 0.0 fitness. A light sensor placed under the robot is used to detect the color of the floor—white, gray, or black—and passed to a host computer in order to switch

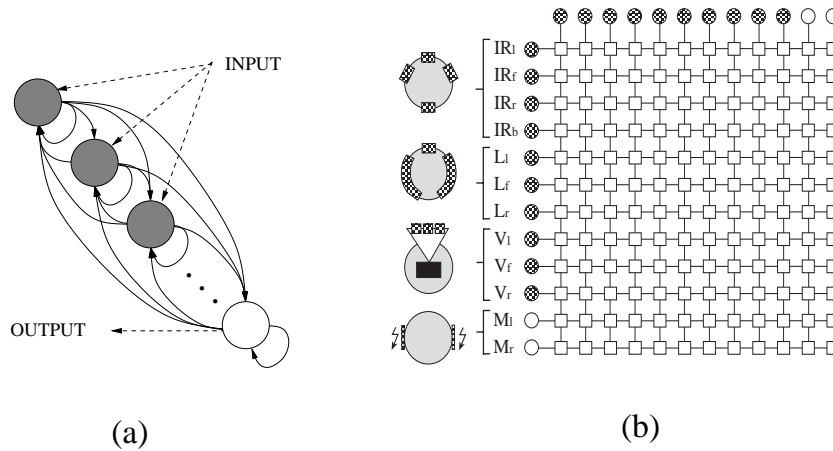


Figure 4: (a) The neural controller is a fully recurrent, discrete-time neural network. (b) The controller can be represented as an unfolded network where the top row shows the neurons at time $t - \Delta t$. It is composed of 12 neurons giving a total of $12 \times 12 = 144$ synapses (here represented as small squares of the unfolded network). 10 sensory neurons receive additional input from one corresponding pool of sensors positioned around the body of the robot shown on the left (l=left; r=right; f=front; b=back). $\vec{I}R$ =infrared proximity sensors; \vec{L} =ambient light sensors; \vec{V} =vision photoreceptors. Two motor neurons \vec{M} do not receive sensory input; their activation sets the speed of the wheels ($M_i > 0.5$ forward rotation; $M_i < 0.5$ backward rotation).

on the light bulb and compute fitness values. The output of this sensor is not given as input to the neural controller because we wish the robot use only infrared and vision sensors to know its own location. After 500 sensory motor cycles, the light is switched off and the robot is displaced by applying random speeds to the wheels for 5 seconds.

Notice that the fitness function does not explicitly reward this sequence of actions (which is based on our external perspective of the task), but only the final outcome of the sequence of behaviors chosen by the robot. This function is behavioral, internal (the computation is based on variables read through the sensors of the robot), and almost implicit (only one constraint is used – time spent under light) (Floreano and Urzelai, 2000).

The controller is a fully recurrent, discrete-time neural network. There is rich literature on the evolution of recurrent neural networks (e.g., Angeline et al. (1993)) and almost all researchers evolving neural controllers use recurrent connections (Nolfi and Floreano, 2000). In this paper, we use a simple recurrent architecture where each node is connected to all other nodes (Figure 4), and we do not evolve its architecture. The network has access to three types of sensory information (Figures 4 and 5):

1. *Infrared light (IR)*: the active infrared sensors positioned around the robot (Figure 5(a) and (c)) measure the distance from objects. Their values are pooled into four adjacent pairs, and the average reading of each pair is passed to a corresponding neuron.
2. *Ambient light (L)*: the same sensors are used to measure ambient light too. These readings are pooled into three groups and the average values are passed to the

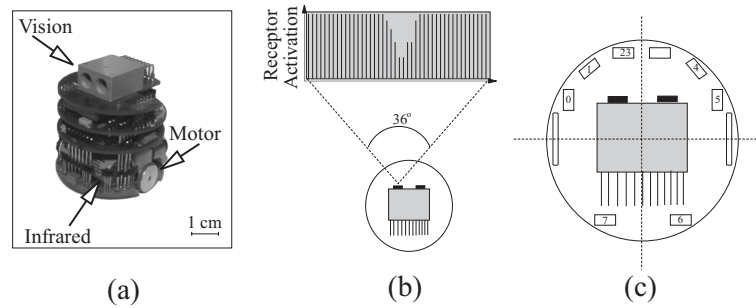


Figure 5: (a) The Khepera robot used in the experiments. The controller receives the activation of the infrared sensors and of the linear vision camera and generates motor commands for the robot. (b) The vision module is composed of 64 photoreceptors covering a visual field of 36° . (c) Configuration of eight infrared sensors.

corresponding three light neurons.

3. *Vision (V)*: the vision module (Figure 5(b)) consists of an array of 64 photoreceptors covering a visual field of 36° . The visual field is divided into three sectors, and the average value of the photoreceptors (256 gray levels) within each sector is passed to the corresponding vision neuron. In addition, the camera accommodates a single light sensor used to detect overall light intensity and adjust online the sensitivity of the photoreceptors.

Two motor neurons (M) are used to set the rotation speed of the wheels (Figure 5(a)), by mapping the activation of each neuron, normalized between 0 and 1, to a discrete speed between -20 and 20 (negative values for backward rotation, and positive values for forward rotation). Neurons are synchronously updated every 100 ms according to the following equation

$$y_i \leftarrow \sigma \left(\sum_{j=0}^N w_{ij} y_j \right) + I_i,$$

where y_i is the activation of the i th neuron, w_{ij} is the strength of the synapse between pre-synaptic neuron j and post-synaptic neuron i , N is the number of neurons in the network, $0 \leq I_i < 1$ is the corresponding external sensory input, and $\sigma(x) = (1 + e^x)^{-1}$ is the sigmoidal activation function. $I_i = 0$ for the motor neurons.

Each synaptic weight w_{ij} is randomly initialized at the beginning of the individual's life and can be updated after every sensory-motor cycle (100 ms),

$$w_{ij}^t = w_{ij}^{t-1} + \eta \Delta w_{ij},$$

where $0.0 < \eta < 1.0$ is the learning rate and Δw_{ij} is one of the four adaptation rules specified in the genotype.

4 Experiments

The experiments have been carried out first in simulations sampling sensor activation and adding 5% uniform noise to these values (see Miglino et al. (1996) for a similar strategy) and later using physical robots. Every replication is composed of a population of 100 individuals, which is evolved for 200 generations. Each individual is tested three times, and the average fitness is calculated. The best 20 individuals are selected for reproduction at each generation, and a new population is obtained by making 5 copies

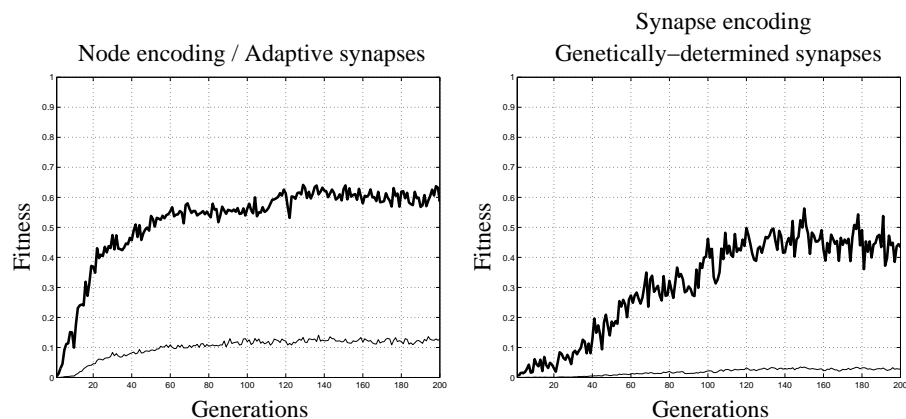


Figure 6: Comparison of different genetic encoding methods in the light-switching problem in simulation. (Left) Node encoding of adaptive synapses. (Right) Synapse encoding of genetically determined synapses. Thick line=best individual; thin line=population average. Each data point is the average over 10 replications with different random initializations. As a reference, the fitness value of a robot driven by hand first towards the black strip and then towards the fitness area is approximately 0.75 (some variability is given by the initial position of the robot in the arena). The slightly lower best fitness of individuals evolved with node encoding of adaptive synapses is due to the adaptive process.

each. We use one-point crossover with crossover probability 0.2. Mutation probability is 0.05 per bit. Each experimental condition consists of 10 evolutionary runs with different random initializations of the population.

The fitness data recorded during evolution and reported in Figure 6 show two main results. The first is that evolved individuals with node encoding of adaptive synapses are better than individuals with synapse encoding of genetically determined synapses in that:

1. Both the fitness of the best individuals and of the population average report higher values (0.6 against 0.5 and 0.12 against 0.03, respectively). The performance difference measured on best individuals of the last generation is statistically significant ($p < 0.05$ for a two-tailed t-test of the difference of the means⁵).
2. They reach the best value obtained by genetically determined individuals in fewer than half generations (40 against more than 100).

The second result is that node encoding of genetically determined synapses (graph not shown) reports zero fitness. This is due to the fact that all incoming synapses to a node always have the same strength and cannot change. Therefore, node encoding of adaptive synapses (Figure 6, left) must be compared to synapse encoding of genetically determined synapses (Figure 6, right).

In addition, we have repeated the evolutionary experiments for node encoding of adaptive synapses and synapse encoding of genetically determined synapses on the

⁵For all two-tailed t-tests presented in this paper, the observation data consists of the fitnesses of the best individual of the last generation corresponding to each replication.

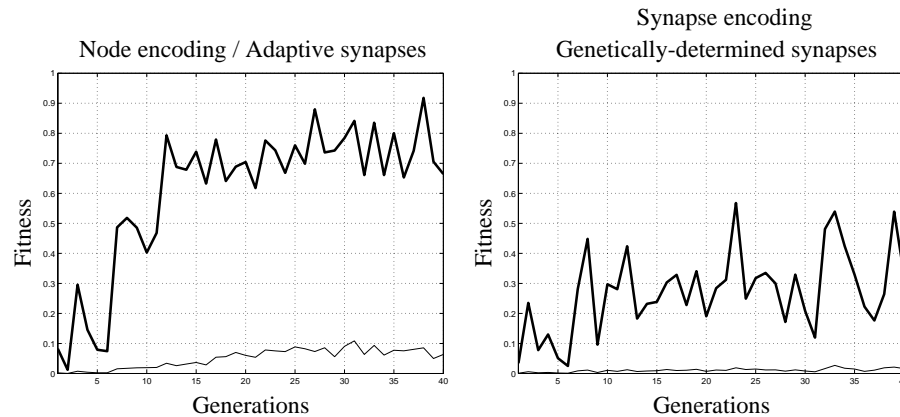


Figure 7: Comparison of different genetic encoding methods for the experiments carried out on the physical robot. (Left) Node encoding of adaptive synapses. (Right) Synapse encoding of genetically determined synapses. Thick line=best individual; thin line=population average. Each data point corresponds to a single replication.

physical robot. Figure 7 shows that the results on the physical robot do not differ significantly from those obtained in simulation. The experimental setup differs from that used in simulated experiments in that: the population size is 80, each run lasts 40 generations, only one run has been carried out for each condition, and each individual is tested only once in the environment. The last restriction means that the effects of chance are more marked on the performance and are the cause of the higher oscillations observed, especially for the individual with genetically determined synapses. These data should be compared to those shown in the graphs of Figure 6. The performances obtained with the physical robots are better than those obtained in simulation because the latter include severe constraints. For example, when a simulated robot pushes against a wall, it cannot move unless it backs away; instead, real robots can often get away by sliding against the walls.

The large difference between best and average fitness values is caused by the selection criterion that requires a robot to both switch the light on *and* go to the fitness area in order to receive fitness points. In all experimental conditions (adaptive and genetically determined), most individuals in the population cannot manage to do both things within their life span. In previous experiments with adaptive and genetically determined synapses where the fitness function explicitly rewarded/punished intermediate states (navigation with obstacle avoidance (Floreano and Mondada, 1996b) and coevolutionary predator-prey robots (Floreano et al., 2001)), we did not observe these large differences. Furthermore, we have observed that the genetic pool converges to a subset of genotypes as long as the fitness of the best individuals keeps increasing (measures of genetic convergence and convergence plots for this experiment can be found in an earlier conference paper (Floreano and Urzelai, 1999)). Therefore, the difference between average and best values is not due to a poor choice of evolutionary method or random search, as one may guess by looking at the graphs only.

Finally, two sets of control experiments, one using node encoding (Figure 8, left) and the other synapse encoding (Figure 8, right), have been carried out in simulation using noisy synapses in order to check whether the improvements obtained by

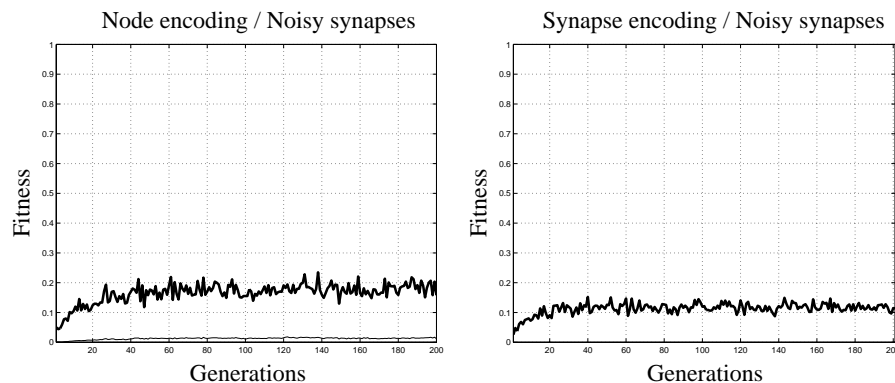


Figure 8: Evolution of noisy synapses. (Left) Node encoding. (Right) Synapse encoding. Thick line=best individual; thin line=population average. Each data point is the average over 10 replications with different random initializations.

evolving adaptive synapses are simply due to a random sampling of the fitness surface surrounding each individual. In both cases, the results are considerably worse than those obtained with adaptive synapses (Figure 6, left) and with genetically determined synapse encoding (Figure 6, right).

4.1 Scaling Up

The choice of a neural architecture is often difficult and may affect the outcome of an experiment. A large architecture may be computationally more powerful, but it may also entail a larger genotype and stronger epistatic effects. Unless one knows that a larger search space for the genotype/phenotype mapping considered has the same proportion of solutions as a smaller one, shorter genotypes may be preferable because evolutionary search could be faster and more effective.

We have performed a new series of experiments using a larger neural network. The architecture shown in Figure 4 was extended by adding 20 hidden neurons. These neurons were fully connected to themselves and to other neurons in the network but did not receive sensory input and were not used to set the speeds of the wheels. The length of the genetic string grows from 60 (5 bits times 12 nodes) to 160 (5 bits times 32 nodes) bits for node encoding and from 720 (5 bits times 144 synapses) to 5120 (5 bits times 1024 synapses) bits for synapse encoding. The results shown in Figure 9 indicate that evolution of node encoding of adaptive synapses reports fitness values still comparable to the case of a smaller network; instead, evolution of synapse encoding of genetically determined controllers is badly affected in this condition. Evolution of node encoding of genetically determined synapses (data not shown) remained close to zero fitness, whereas evolution of synaptic strength and noise range with both node encoding and synapse encoding reported results similar to those shown in Figure 8 (data not shown).

The fact that synapse encoding of genetically determined controllers performs badly may indicate that the search space here contains proportionally fewer solutions than the smaller search space of the network pictured in Figure 4. The slower convergence and slightly lower fitness values of the controller with node encoding of adaptive synapses (compare with Figure 6, left) may be explained by the increased length of the

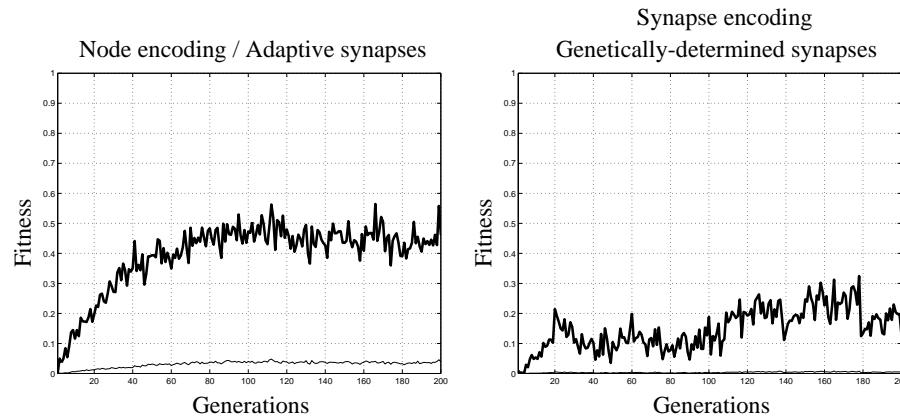


Figure 9: Evolution of large controller with 20 hidden neurons. (Left) Node encoding of adaptive synapses. (Right) Synapse encoding of genetically determined synapses. Thick line=best individual. Each data point is the average over 10 replications with different random initializations.

genetic string, but also by the fact that the architecture is fixed and fully connected. Since in node encoding the properties of a node propagate to all incoming synapses, there might be a high number of “parasitic” connections that cannot be individually eliminated.

5 Functional and Behavioral Analysis

In this section, we analyze the evolved mechanisms and behaviors of evolutionary adaptive individuals. For the sake of clarity and consistency, we shall always refer to the same evolved individual, but the results hold for all the best individuals of the ten evolutionary runs. Figure 10 shows the behaviors of two typical individuals evolved with node encoding of adaptive synapses (left) and with synapse encoding of genetically determined synapses (right). Notice that synapses of adaptive individuals are allowed to change during the behavioral tests. The adaptive individual aims at the area with the light switch⁶, and once the light is turned on, it moves towards the light and remains there. Instead, the genetically determined individual displays always the same looping trajectory around the environment with some attraction towards the stripe and the light (some genetically determined individuals are not even capable of standing still on the fitness area, result not shown). The “minimalist” behavior of genetically determined robots, which depends on invariant geometrical relations of the environment, gives them a chance to accomplish the task but with a lower performance. The better fitness of the adaptive controllers (shown under each figure) is given by straight and faster trajectories showing a clear behavioral change between the first phase where the robot goes towards the switching area and the second phase where it becomes attracted by the light.

Why do individuals with adaptive synapses perform better and differently? For example, one might argue that what matters is the sign of the synapse and not its strength as long as it is non-zero, or that adaptive synapses may have the same effect

⁶Its performance is badly affected if the vision input is disabled, indicating that it does not use random search to locate the switch (data not shown).

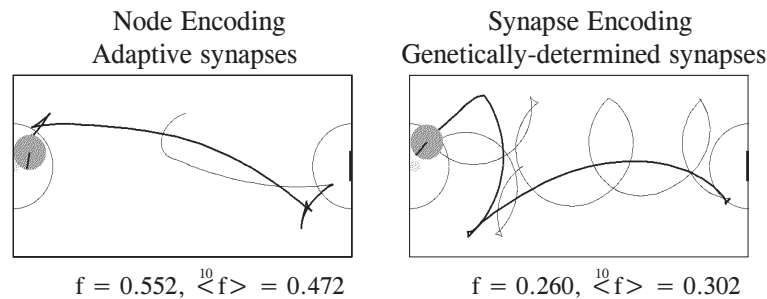


Figure 10: Typical behavior of best individuals of the last generation. (Left) Node encoding of adaptive synapses. (Right) Synapse encoding of genetically determined synapses. When the light is turned on, the trajectory line becomes thick. The corresponding fitness value is printed at the bottom of each box along with the average fitness of the same individual tested ten times from different random positions and orientations.

of fixed synapses with strengths set to their average values.⁷ In order to evaluate this hypothesis, the same best individual with adaptive synapses shown in Figure 10 (left) is tested again without allowing synaptic change and by initializing synapses in three different ways (Figure 11). In the first condition, the synapses are initialized to small random values in the range $[0.0, 0.1]$ (Figure 11, left), as during evolution. In the second condition, the weights are all set to their maximum strength 1.0 (Figure 11, center). In the third condition, the weights are set to their average value (the average values were obtained while testing the robot in adaptive mode and recording the synaptic strength of each connection after every update) (Figure 11, right). The evolved signs are maintained. For each condition, the individual is tested ten times from different positions and orientations, and adaptation is not allowed. None of the individuals manages to complete the task in any of the three conditions. The results reported by these control experiments suggest that the behavior displayed by evolved adaptive individuals is functionally related to online changes of synaptic strengths.

When we observe the synaptic activity of the evolved controller while the robot moves in the environment, we notice that several synapses keep changing along the whole duration of the behavioral sequence (Figure 12). In other words, it is not possible to find an initial learning phase as in most conventional connectionist algorithms.

Nonetheless, this pattern of change can be characterized by three major events, where most synapses transit into new states, labelled as A, B, and C in Figure 12. These events correspond to three clearly different behavioral stages. Event A marks the point when most changing synapses reach a temporarily stable state from their initial random values, and the robot displays a smooth trajectory towards the black stripe on the wall corresponding to the light switch. Event B is in between the end of a minor transition and the beginning of a major transition and corresponds to the point when the robot switches the light on. Finally, event C marks the end of another period of relatively long change and the beginning of another major temporary transition and corresponds to the moment when the robot enters the fitness area for the first time (remember that the neural controller does not receive information from the floor sensor).

⁷This latter suggestion was made by Flotzinger (1996) who replicated our previous experiments on synapse encoding of adaptive synapses (Floreano and Mondada, 1996b).

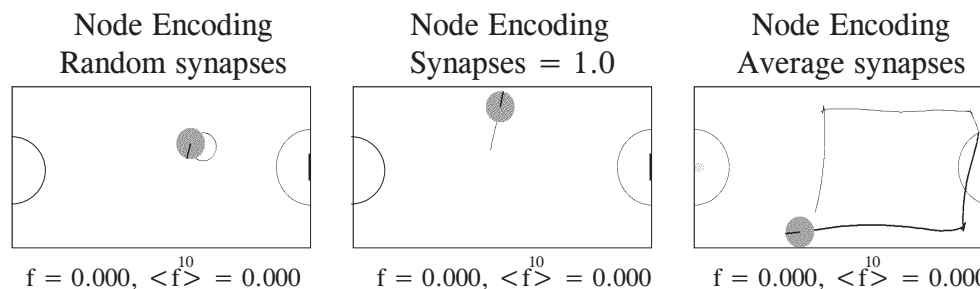


Figure 11: Disabling adaptation on the individual evolved with node encoding of adaptive synapses (Figure 10, left). The synapses are initialized in 3 different ways and are not allowed to change during the tests. (Left) Synapses are initialized to random values in the range $[0.0, 0.1]$, as during evolution. (Center) Synapses are all initialized to 1.0. (Right) Synapses are set to their average value recorded during a full test of the individual. In all cases, the evolved signs (positive or negative) are maintained. The fitness of the individual is shown under each box along with the average fitness of the same individual tested ten times from different positions and orientations. The values are always 0.0 because none of these individuals ever manages to complete the task under these test conditions. The trajectory line is thin when the light is off and becomes thick when the light is turned on.

These data suggest that synaptic change corresponds to the acquisition of and switching between different sub-behaviors. To further explore this hypothesis, we studied the behavior corresponding to the values of the synapses recorded at the moment of the three events. To do so, we *froze* the weights of the synapses after each event and observed the corresponding behaviors of the robot for a few seconds starting from 6 initial locations and orientations in the center of the arena both when the light was on and off. Figure 13 shows the results of these tests where the synapses are not allowed to change. The boxes show the *initial* positions of the robot and its trajectories using the values of the synapses recorded at event A, B, and C, respectively. The top row shows the tests performed with the light turned off and the bottom row those performed with the light turned on. Although the robot could not display its complete behavior using a set of fixed weights (as we already pointed out with the tests reported in Figure 11 above), its trajectories displayed functionally different abilities corresponding to the three sets of synaptic values. After event A, when the light is off the robot rotates on itself and moves towards the black stripe on the wall (light switching area); however, when the test is repeated with the light on, it tends to be attracted by the light. After event B, when the light is off, the robot is no longer attracted by the black stripe on the wall; when the test is performed with the light on, the robot clearly moves towards the light area. After event C, when the light is off, it turns in place, but when the light is on, it still tends to go towards the light area.

This series of tests indicates that evolved adaptive controllers can develop and/or modify on-line new behaviors according to the actual environmental conditions. These behaviors are not learned in the classic meaning of the term because they are not necessarily retained forever. For example, the ability to visually locate the black stripe on the wall and move towards it is acquired at a certain stage and disappears once it is no longer necessary. Instead, the attraction towards the light develops early on, even

Node Encoding / Adaptive Synapses

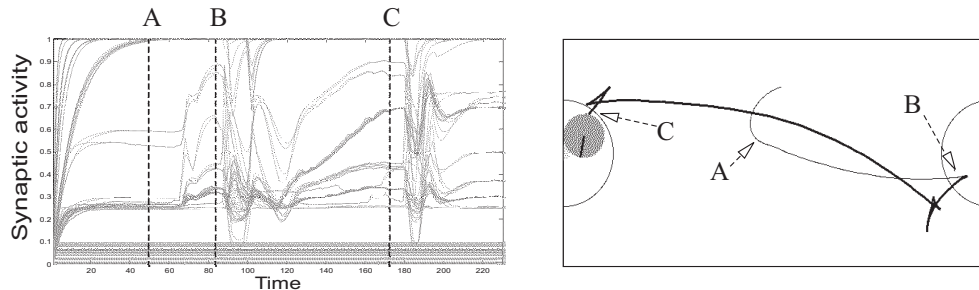


Figure 12: Relation between synaptic activity and behavior of an adaptive individual. (Left) Synaptic activity recorded during the trajectory of the robot. Each line corresponds to the instantaneous strength of a synapse measured every 100 ms. Ticks on the time scale represent the measurements. (Right) Trajectory of the robot. Points A, B, and C correspond to attraction by the black stripe, switching on of light, and arrival to the fitness area, respectively.

before experiencing the light bulb switched on (however, notice that light sensors are always activated to some extent by normal ambient light) and is retained also after having reached the fitness area. The graph of synaptic activity shows that after each event, there is a major temporary change in all synapses. This always corresponds to the situations when the robot is close to the walls and (re-) develops online the ability to move away from them. From the experiments described in this paper, it is not yet possible to tell whether some behaviors are not retained because they are mutually exclusive in weight space with other behaviors, or simply because this type of solution is easier to find by evolution than one where the weights can accommodate and retain all the necessary behaviors. In order to do so, one should modify the experimental conditions so that selective reproduction requires conservation of previously acquired abilities. Our current work is aimed at investigating this and other issues related to the evolution of “conservative learning.”

To summarize, the behavioral and functional analysis described in this section shows that evolved adaptive individuals exploit plastic synapses to develop and modify abilities depending on environmental information and behavioral states. As we see in the next sections, this characteristic is very useful in coping with unpredicted sources of change.

6 Online Adaptation to New Environments

In this section, we show that evolved adaptive controllers can cope with several types of change that were not included during evolutionary training. The best individuals evolved under the three different conditions (adaptive, genetically determined, and noisy) are tested in new environments and their performances are compared. Notice that during these tests, evolution is disabled.

We consider four types of environmental change: (i) new sensory appearances; (ii) transfer from simulations to physical robots; (iii) transfer across different robotic platforms; (iv) re-arrangement of environmental layout.

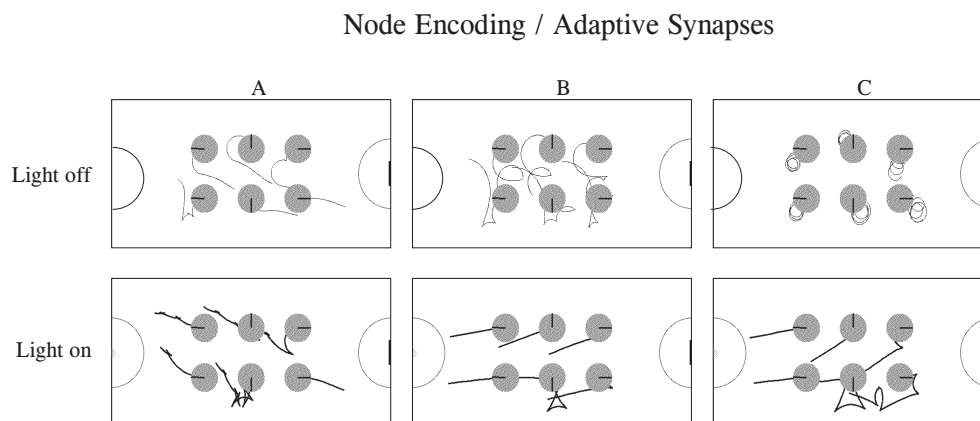


Figure 13: Behavior of the robot in points A, B, and C, which correspond to the beginning of the epoch, the activation of the light, and the arrival of the robot to the fitness area, respectively. (Upper row) Behavior of the robot when the light is turned off. (Lower row) Behavior of the robot when the light is turned on.

6.1 Changing Sensory Appearances

New sensory conditions are a major cause of failure for pre-programmed and evolved robots. Typically, when the lighting conditions or the reflective properties of the objects are not the same as those used in the laboratory, robots are no longer capable of performing correctly.

In order to measure the performance of our evolved robots in environments with new sensory characteristics, the best individuals of the last generation for each of the 10 replications are tested 10 times in environments with white (used during evolution), gray, and black walls.⁸ The resulting 100 fitness values are averaged for each environment type. This testing procedure is repeated for each experimental condition (adaptive, genetically determined, noisy). Gray and black walls reflect much less infrared light and therefore can be detected at a shorter distance by the robot.

Figure 14 shows average fitnesses corresponding to environments with white, gray, and black walls in the case of individuals with node encoding of adaptive synapses (left), individuals with synapse encoding of genetically determined synapses (center), and individuals with node encoding of noisy synapses (right). Although fitness values are generally lower in gray and black environments, adaptive individuals are capable of successfully performing the task in all three environments. Instead, genetically determined individuals can perform normally only in the environment that has been used during evolution (white walls). When tested in gray and black environments, none of the genetically determined individuals is capable of solving the task. The difference in performance loss between adaptive controllers and genetically determined controllers is statistically significant, $p < 0.05$ for a two-tailed t-test of the difference of the means. Individuals with noisy synapses score very low fitness values in all conditions.

Figure 15 displays some typical behaviors of individuals with adaptive synapses

⁸Color modification is applied only to the bottom of the walls in order to change the response of the infrared sensors without modifying the output of the linear camera.

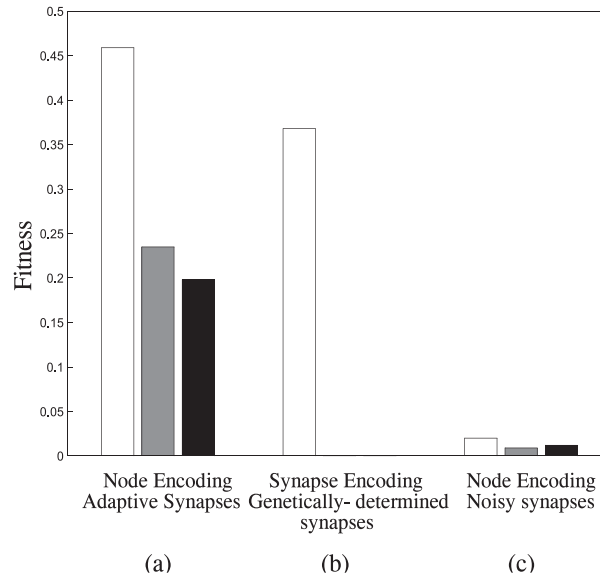


Figure 14: Comparison of node encoding of adaptive synapses: (a) versus synapse encoding of genetically determined synapses (b) and node encoding of noisy synapses (c) in white, gray, and black environments. Each fitness value is the average over 100 data (the best individual of each of the 10 replications is tested 10 times with different random initializations).

(left column), genetically determined synapses (center column), and noisy synapses (right column) in environments with white (top row), gray (center row), and black (bottom row) walls. The behavior of the adaptive individual is not considerably affected by the color of the walls, and it reaches the fitness area in all conditions. Instead, the genetically determined individual can accomplish the task only when walls are covered with white paper but gets stuck on gray and black walls. Since darker walls are detected only when the robot gets closer, a behavioral strategy successful for white walls can cause collisions for dark walls. The individual with noisy synapses takes advantage of the random variability to get away from the walls but it scores a low fitness because its strategy is based in random navigation.

6.2 From Simulations to Real Robots

Another way of measuring the adaptive abilities of evolved controllers is to transfer them from simulated to real robots. Since simulated and physical worlds have different characteristics, controllers evolved in simulation typically fail when tested on real robots (Brooks, 1992).

The solutions envisaged so far consist of incorporating special types of noise tailored to sensory-motor properties of the robot (Miglino et al., 1996) or to vary key-features of the environment during simulated evolution (Jakobi, 1997). The success of both methods depends upon the ability of the experimenter to spot crucial aspects of variation that must be considered in the simulations. Another solution consists of incrementally carrying on artificial evolution in the physical environment (Floreano and Mondada, 1998), but this can take long time.

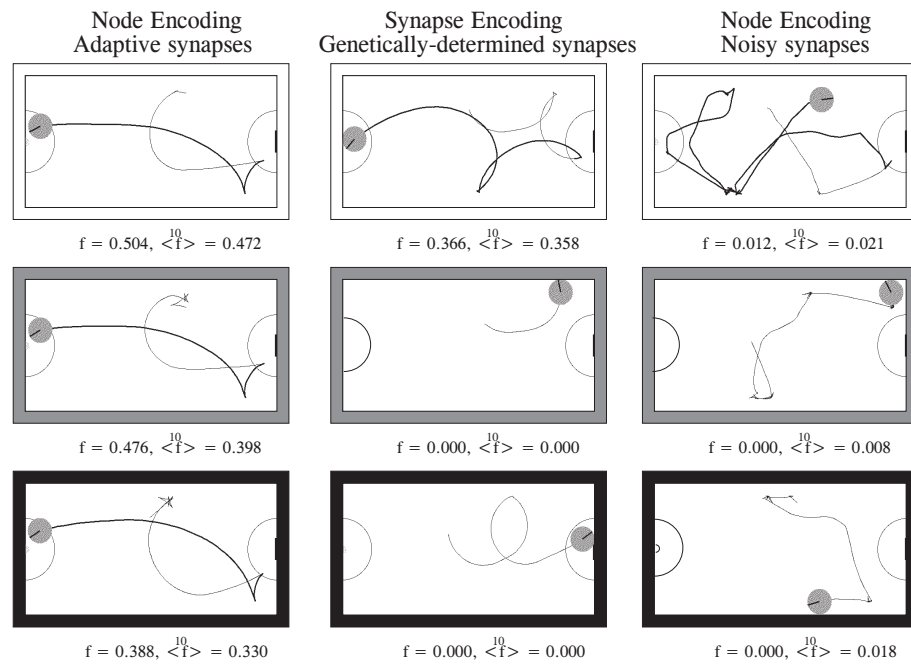


Figure 15: Typical behaviors of evolved individuals tested in environments with white (top row), gray (center row), and black (bottom row) walls. (Left column) Node encoding of adaptive synapses. (Center column) Synapse encoding of genetically determined synapses. (Right column) Node encoding of noisy synapses. Individuals belong to the last generation evolved in the environment of the top row. The trajectory line is thin when the light is off and becomes thick when the light is turned on. The corresponding fitness value is printed under each box along with the average fitness of the same individual tested ten times from different random positions and orientations.

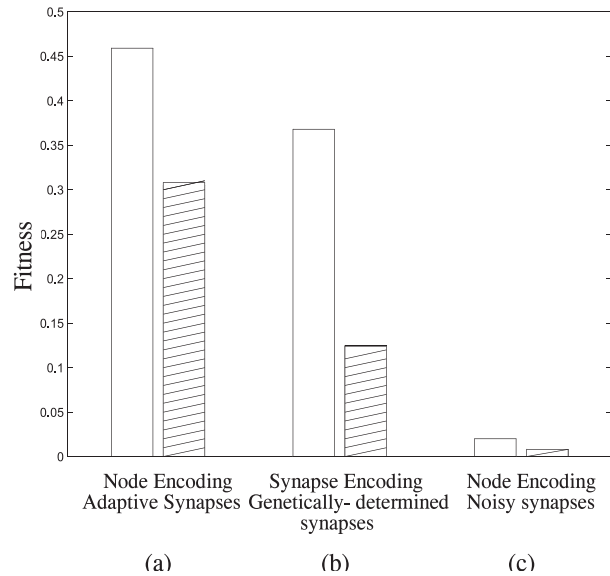


Figure 16: Comparison of node encoding of adaptive synapses (a) versus synapse encoding of genetically determined synapses (b) and node encoding of noisy synapses (c) in simulated environments (white bars) and on a real Khepera robot (striped bars). Each fitness value is the average over 30 data (the best individual of each of the 10 replications is tested 3 times with different random initializations).

Here, we have transferred to a physical Khepera robot the best individuals of the last generation for each of the 10 populations evolved in simulation (walls were covered with white paper, Figure 3). Figure 16 shows that the performance of adaptive individuals is less affected by the transfer to the physical environment than genetically determined individuals (the difference is statistically significant, $p < 0.05$ for a two-tailed t-test of the difference of the means). Individuals with noisy synapses are not affected by the transfer because their behavior is always random and not effective in both simulated and physical environments. Slight loss of performance in adaptive individuals is caused by the fact that in some cases the robot performs looping trajectories around the fitness area without coming to rest on it. Instead, the two major reasons of failure for genetically determined individuals are more serious: they either get stuck against the walls or cannot approach the light.

6.3 Cross-Platform Adaptation

Cross-platform transfer refers to the case where a controller designed or evolved for a given robot is transferred to another type of robot.

Cross-platform transfer is a very useful feature, but we are not aware of any control system that can be transferred across different robots without external changes. Cross-platform becomes useful in adaptive and evolutionary systems where certain initial training experiences can damage large or complex robots. One may train (or evolve) control systems for a desktop sturdy robot like the miniature Khepera and then download them to larger and more fragile robots.⁹ In this case, it would be desirable that the

⁹Obviously, the two robots must share some characteristics, such as type of sensors and actuators used, in

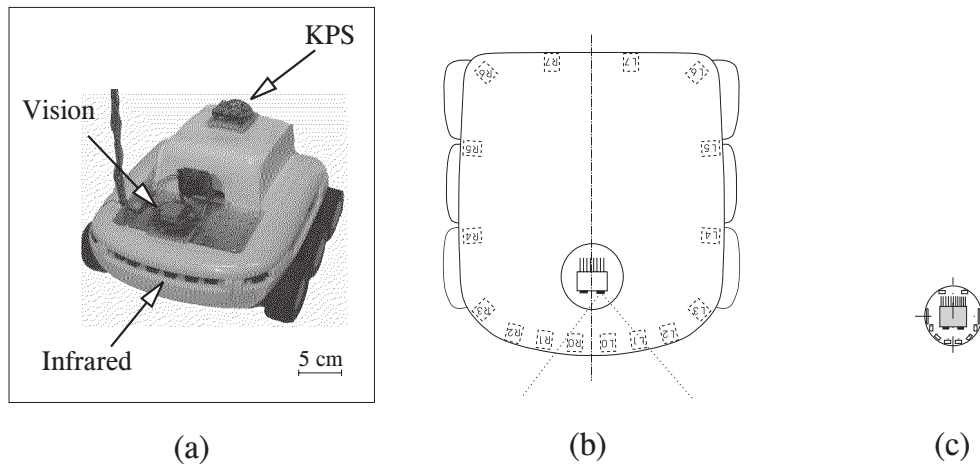


Figure 17: The Koala robot used in the experiments. (a) The controller receives the activation of the infrared sensors (with different detection range and response profile to those used by the Khepera) and the linear vision camera (same as Khepera) as input and generates motor commands for the robot. The localization module (KPS) provides the position of the robot at every time step in order to plot its trajectory. (b) Sensory configuration of the Koala robot. (c) Size of the Khepera robot compared to the Koala robot.

control system self-adapts to the new sensory-motor characteristics and morphology. Floreano and Mondada (1998) have shown that this can be achieved by using incremental evolution of genetically determined networks. However, even for a simple reactive navigation behavior it took 20 additional generations for the controllers to re-adapt to the new robotic platform.

Here, we transfer onto the larger Koala robot (Figure 17) the best individuals of the last generation evolved on the miniature Khepera robot in the light-switching problem (Section 4). A mobile robot Koala equipped with a vision module (Figure 17(a)) is positioned in the rectangular environment shown in Figure 18. As in the previous experiment with the Khepera robot, the Koala robot must find the light-switching area, go there in order to switch the light on, and then move towards the light as soon as possible and stay there in order to score fitness points.

The Koala robot (32 x 32 cm, rectangular shape) has six wheels driven by two motors (one on each side) and 16 infrared sensors (Figure 17(b)) with a different and longer detection range. Only 8 equally spaced sensors of the 16 available are selected as input to the neurocontroller. An external positioning system emitting laser beams at predefined angles and frequencies is positioned on the top of the environment and the Koala robot is equipped with an additional turret capable of detecting laser (Floreano and Mondada, 1998) and computing in real-time the robot displacement (KPS, Figure 17(a)). This information is used by the computer to detect the robot position, and switch the light on and increment fitness points when the robot is by the switch and by the lighted lamp, respectively.

Figure 19 shows that the performance of adaptive individuals is only slightly af-

order to allow interfacing of the control system.

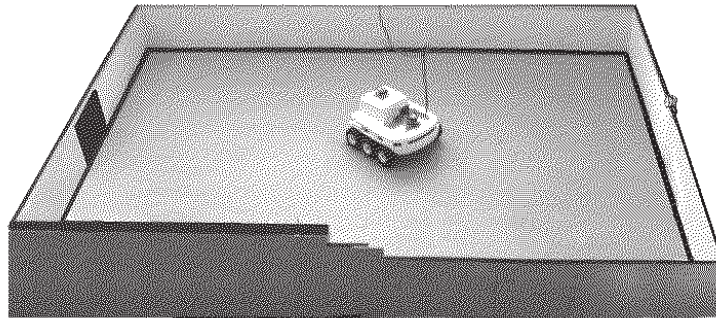


Figure 18: A mobile robot Koala equipped with a vision module gains fitness by staying near the lamp (right side) only when the light is on. The light is normally off, but it can be switched on if the robot passes near the black stripe (left side) positioned on the other side of the arena. The robot can detect ambient light and the color of the wall. Since position of the robot is monitored by an external positioning system and passed to the computer in order to control the light and to compute the fitness, we do not need painted areas on the floor like in the experiment with the Khepera. See Saucy (2000) for more details.

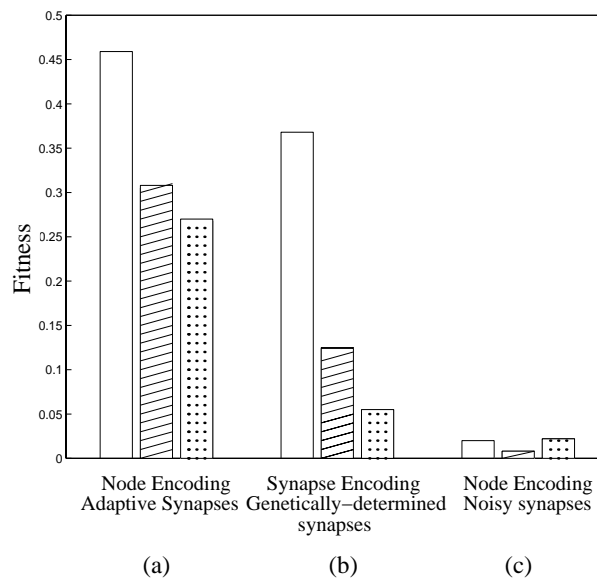


Figure 19: Comparison of node encoding of adaptive synapses (a) versus synapse encoding of genetically determined synapses (b) and node encoding of noisy synapses (c) in simulated environments (white bars), on the Khepera robot (striped bars), and on the Koala robot (dotted bars). Each fitness value is the average over 30 data (the best individual of each of the 10 replications is tested 3 times with different random initializations).

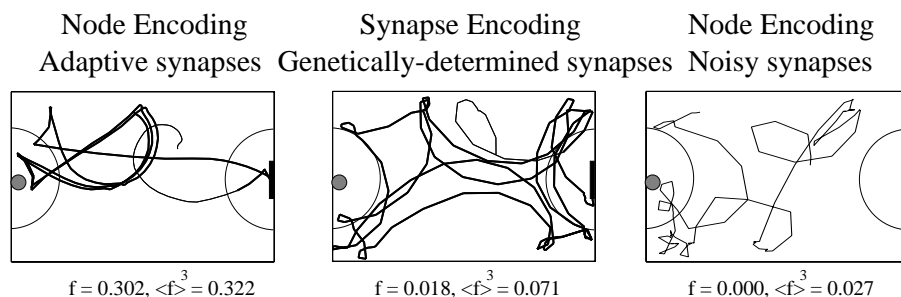


Figure 20: Behavior of best individual of the last generation evolved in simulation for the Khepera robot and tested on the physical Koala robot. (Left) Node encoding of adaptive synapses. (Center) Synapse encoding of genetically determined synapses. (Right) Node encoding of noisy synapses. The trajectory line is thin when the light is off and becomes thick when the light is turned on. The corresponding fitness value is printed under each box along with the average fitness of the same individual tested three times from different random positions and orientations. Trajectories are obtained from the laser positioning system described in the text of the article.

ected by the transfer from the Khepera robot (striped bars) to the Koala robot (dotted bars), whereas genetically determined individuals report a significantly larger fitness loss ($p < 0.05$ for a two-tailed t-test of the difference of the means). Individuals with noisy synapses are not affected by the transfer because their behavior is always random and not effective in both Khepera and Koala robots.

Adaptive individuals correctly approach the light-switching area and then become attracted by light (Figure 20, left). As in the case of the real Khepera robot (see section 6.2), once arrived under the light, the Koala robot moves around the fitness area while remaining close to it until the testing time is over. This is the only reason why their performance is slightly inferior to that measured during evolution on the Khepera. On the other hand, genetically determined individuals (center) perform looping trajectories around the environment and do not display any attraction by the black stripe or the light. They eventually manage to pass through the light-switching area, turn the light on, and occasionally score some fitness points passing over the fitness area by chance. In several cases, genetically determined individuals get stuck against the walls of the environment (behaviors not shown). Individuals with noisy synapses (right) score low fitness because their strategy consists of random movements.

6.4 Changing Spatial Relationships

In this section, we address variations induced by changed spatial relationships of the environment. The best individuals of the last generation are tested in environments where the light-switching area, the fitness area, and the robot are located at random positions at the beginning of each individual's life. Since in the original experiment the positions of the light-switching area and of the fitness area were constant for every individual, this experiment gives us a measure of adaptation capabilities of evolved individuals to new spatial relationships. In order to automate the re-arrangement of the environment, these experiments have been carried out in simulation. The best individuals for each of the 10 populations evolved in the environment of Figure 3 are tested in 3 new environments with different random spatial relationships.

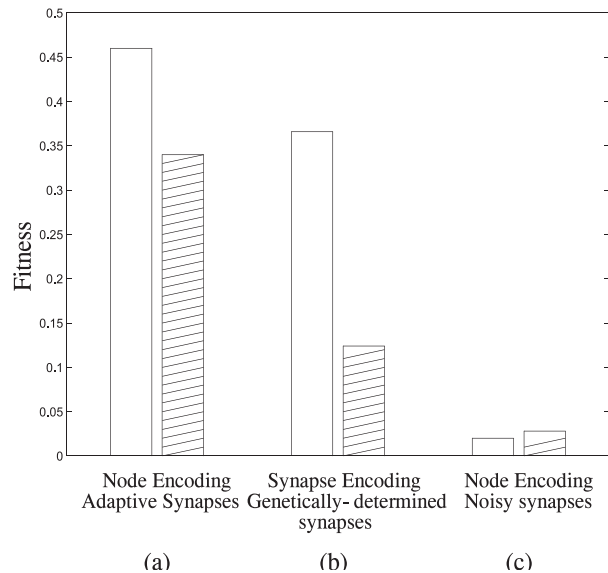


Figure 21: Comparison of node encoding of adaptive synapses (a) versus synapse encoding of genetically determined synapses (b) and node encoding of noisy synapses (c) in the environment used during evolution (white bars) and in an environment where light-switching area and fitness area are randomly positioned (striped bars). Each fitness value is the average over 30 data (the best individual of each of the 10 replications is tested in 3 new environments with different random initializations).

The results reported in Figure 21 show that individuals with adaptive synapses are much more robust to new configurations of the environment than individuals with genetically determined synapses. Average performance loss is about 25% in the case of adaptive individuals (left) but goes up to about 65% in the case of genetically determined individuals (center). The difference in performance loss is statistically significant ($p < 0.05$ for a two-tailed t-test of the difference of the means). Individuals with noisy synapses (right) score very low fitness in both cases.

The fact that genetically determined individuals performed very poorly in new environments indicates that the solutions generated by evolution alone are tightly coupled to the layout of the environment. Genetically determined individuals take advantage of invariant environmental aspects, such as the size of the arena, the position of the light-switching area and of the fitness area. Instead, evolved adaptive controllers are capable of generating solutions that produce successful behaviors for a large variety of environmental layouts.

Figure 22 shows the behavior of three evolved individuals (adaptive, genetically determined, noisy) tested in the environment used during the evolutionary process (top row) and in an environment where both the light-switching area and the fitness area are placed against the same wall of the environment (bottom row). The adaptive individual (left column) is capable of solving the task in the original environment, and changes the strategy by performing some additional maneuvers that allow the robot to reach the fitness area in the new environment. Instead, the genetically determined individual (center column) is capable of solving the task in the original environment

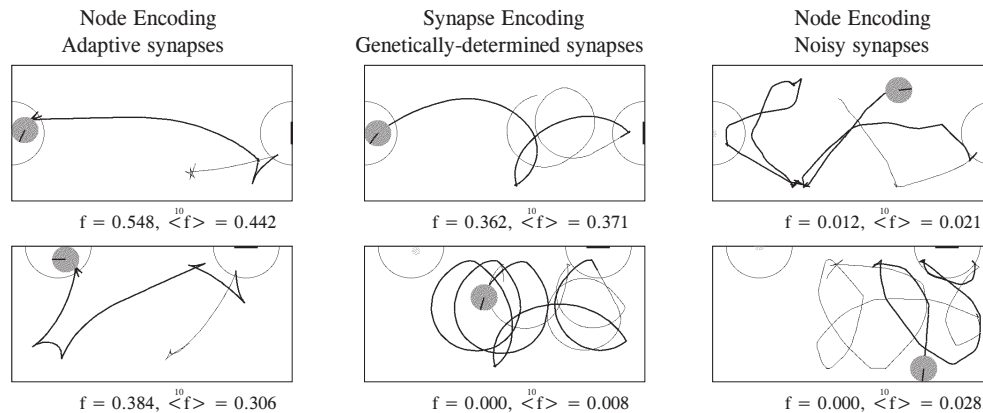


Figure 22: Behaviors of best individuals of the last generation when tested in the original environment used during evolution (top row) and in an environment where light-switching area and fitness area are placed at new positions (bottom row) in the “light-switching” environment. Left column: Node encoding of adaptive synapses. Center column: Synapse encoding of genetically determined synapses. Right column: Node encoding of noisy synapses. The trajectory line is thin when the light is off and becomes thick when the light is turned on. The corresponding fitness value is printed at the bottom of each box along with the average fitness of the same individual tested ten times from different random positions and orientations.

by performing circular movements and avoiding the walls until it reaches the fitness area, but these circular movements are not effective to approach the fitness area in the new environmental layout because the robot is repelled by the walls as it approaches the light. Finally, the individual with noisy synapses (right column) performs random trajectories that provide low fitness in both environments.

7 Conclusion

We have shown through a set of systematic comparisons that evolution of adaptive synapses brings a number of advantages with respect to evolution of fixed synaptic weights. It can generate viable controllers in much fewer generations and such evolved controllers display more successful behaviors. Since the strength of adaptive synapses need not be specified on the genetic string (their strength is always randomly initialized at the beginning of an individual’s test), this approach can rely on a very compact genetic encoding that specifies the adaptive properties (and the sign) only of individual neurons. Such a compact encoding scales up very well to large networks with many synapses. The data obtained from control experiments with noisy synapses and from behavioral tests of evolved individuals with adaptation disabled, together with the behavioral analysis related to synaptic activity of adaptive individuals, all suggest that evolved Hebbian adaptation plays a functional role in the development of the controllers both during evolution and during the life of an individual.

When describing our controllers with adaptive synapses, we have deliberately avoided the term “learning”¹⁰ because we have no evidence that the controller acquires

¹⁰Except when we referred to “Hebbian *learning* rules” and to the “*learning* rate η ” in order to be consistent

new knowledge or skills, or that it may easily acquire new abilities for a different task (implementing, for example, something functionally similar to reinforcement learning). Instead, we have used the term “adaptation” because synapses change according to the states of the sensors and of the other neurons in the controller of the robot. In other words, they adapt their initial random configuration to a set of dynamically stable configurations (attractors) in ways that are functionally related to the selection criterion and to the actual environmental situations. In previous work (Floreano and Mondada, 1996b), we showed that although some of the synaptic weights continue to change while the observed behavior of the evolved robot is very stable, the pattern of change becomes stable. In other words, after an initial period of synaptic updates, the set of synaptic values can be considered as a dynamical system cycling around a stable attractor. Within this framework, the effect of the initial set of synaptic change would correspond to the trajectory of the system from a random initial point (when the synaptic values are randomly initialized) to such a stable attractor. The role of artificial evolution consists of selecting the appropriate set of rules and conditions (Hebbian rules, synaptic signs, and learning rates) by which the system can go as fast as possible toward such an attractor and stay there.

In this paper, we have shown that this type of evolved adaptivity is quite useful when environmental conditions change. Adaptive individuals are capable of successfully performing in environments that are different from the one used during evolution by adapting their strategy to the constraints of the new environments. Instead, genetically determined individuals often fail in adapting to different environments because their behavior is tightly coupled to the characteristics of the environment used during evolution. We have studied adaptation to four major sources of environmental change: new sensory appearances, transfer from simulations to real robots, cross-platform transfer, and new spatial relationships. In all cases, evolved adaptive controllers can autonomously modify their parameters and behavior on-line without requiring additional evolutionary training or ad-hoc manipulation of the evolutionary procedure. This robustness is in part due to the fact that synaptic strengths are not encoded in the genetic string and therefore must develop the appropriate values on the fly, depending on the features of the environment where the individual is placed. The role of evolution is to select a combination of learning rules and signs such that the development of synaptic strengths in interaction with the environment will converge towards attractors that correspond to behaviors capable of maximizing the fitness function. We have shown that such attractors correspond to obstacle avoidance, visually-guided navigation towards the light switch, light following, and remaining close to the light bulb. The details of how these behaviors are implemented (i.e., the exact pattern of synaptic values) depend largely on the properties of the environment, of the sensors, motor system, and of robot morphology. In recent work that appeared while this article was being reviewed, Di Paolo (2000) used a method very similar to that presented here and showed that adaptation to changed environmental properties can be viewed as a homeostatic process whereby a dynamical system returns to its equilibrium point after having being displaced from there.

We think that the method presented in this paper does represent a promising solution to a major bottleneck in applied evolutionary robotics, that is the amount of time required to evolve control systems in the physical world. Evolution alone is not always feasible for real-world applications where on-line adaptation is required and for situations where control systems must be evolved for large and fragile robots (Matiarić

with the literature on neural networks.

and Cliff, 1996). However, by resorting to evolution of adaptive mechanisms, we may rely more heavily on simulations or on simple robots to carry out an initial evolutionary phase and then transfer the evolved adaptive controllers on the final physical robot where they will quickly develop the appropriate set of synaptic strengths. For example, in scenarios like those of robots probing an asteroid surface or robots interacting with an handicapped person, it is impossible to evolve the control system on the spot (not even incrementally). Instead, one might reproduce the expected working conditions in the laboratory to some degree of approximation and evolve the adaptive controller in those simplified conditions through a combination of simulations and simple robots. The evolved adaptive controllers would then be transferred on the final robot and let free to adapt to actual working conditions in a few seconds.

7.1 Future Work

In the experiments presented in this article, once an individual has carried out the task for which it has been evolved, it does not retain some of the abilities developed earlier. For example, if we switch the light off when the robot is on the fitness area, it will not be able to develop again attraction towards the black stripe. This happens because there is not a path between the two attractors corresponding to points C and A in synaptic space (see Figure 12). Instead, in that situation, the robot begins to rotate (see Figure 13, top right box), which is an effective strategy to detect light gradient in the environment. In other words, since evolutionary training never required a repetition of the task, the controller “expects” that the light is still on and that the light gradient has been lost.

Our current work aims at evolving adaptive controllers that may be capable of preserving previously acquired abilities. One way of doing so could be adding synapses that enable or disable the plasticity of other synapses. Another possibility might be to enable or disable plasticity by means of neural modulators (Eggenberger et al., 1999) or by means of neural gases (Husbands, 2000). However, one should carefully consider the extent to which such evolved systems become fixed instead of remaining completely adaptive.

Another current research direction is concerned with the evolution of more complex and powerful neural morphologies.¹¹ In current methods of evolutionary morphology (Nolfi and Floreano, 2000), there is a trade-off between the complexity of the genotype to phenotype mapping and the evolvability of the system, partly due to the fact that the phenotype largely depends on genetic instructions. In other words, current methods tend to have very complicated mapping rules and several different genetic parameters to encode both the morphology and the synaptic strengths. We think that by *co-evolving* the learning rules along with morphological factors (e.g., position and type of neurons) and by letting the final structure develop in close interaction with the environment, simpler genetic encodings and higher tolerance to mutations can be obtained. This would make the evolved controllers more viable, add neutrality to the genetic landscape, and ultimately improve evolvability.

Acknowledgments

We thank the three anonymous referees and the action editor for their detailed and constructive feedback that helped to improve the clarity of the manuscript. We are grateful to Patrick Saucy for his help on the cross-platform experiments. Joseba Urzelai is supported by grant nr. BF197.136-AK from the Basque government. Dario Floreano

¹¹The *morphology* of a neural network can be determined by the number and the position of neurons, their connectivity pattern, and other characteristics of neurons and connections.

acknowledges support from the Swiss National Science Foundation, grant no. 620-58049.

References

- Ackley, D. H. and Littman, M. L. (1992). Interactions between learning and evolution. In Langton, C. et al., editors, *Artificial Life II: Proceedings Volume of Santa Fe Conference*, volume XI, pages 487–510, Addison Wesley, Redwood City, California.
- Angeline, P. J., Saunders, G. M., and Pollack, J. B. (1993). An evolutionary algorithm that constructs recurrent neural networks. *IEEE Transactions on Neural Networks*, 5:54–65.
- Brooks, R. A. (1992). Artificial Life and real robots. In Varela, F. J. and Bourgine, P., editors, *Toward a practice of autonomous systems: Proceedings of the First European Conference on Artificial Life*, pages 3–10, MIT Press, Cambridge, Massachusetts.
- Chalmers, D. J. (1990). The evolution of learning: An experiment in genetic connectionism. In Touretzky, D. S. et al., editors, *Proceedings of the 1990 Connectionist Models Summer School*, pages 81–90, Morgan Kaufmann, San Mateo, California.
- DiPaolo, E. A. (2000). Homeostatic adaptation to inversion of the visual field and other sensorimotor disruptions. In Meyer, J. et al., editors, *From Animals to Animats VI: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*, pages 440–449, MIT Press, Cambridge, Massachusetts.
- Eggenberger, P. et al. (1999). Towards seamless transfer from simulated to real worlds: A dynamically-rearranging neural network approach. In Wyatt, J. and Demiris, J., editors, *Proceedings of the Eighth European Workshop on Learning Robots*, pages 4–13, EPFL, Lausanne, Switzerland.
- Floreano, D. and Mondada, F. (1996a). Evolution of homing navigation in a real mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 26:396–407.
- Floreano, D. and Mondada, F. (1996b). Evolution of plastic neurocontrollers for situated agents. In Maes, P. et al., editors, *From Animals to Animats IV: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 402–410, MIT Press, Cambridge, Massachusetts.
- Floreano, D. and Mondada, F. (1998). Evolutionary neurocontrollers for autonomous mobile robots. *Neural Networks*, 11:1461–1478.
- Floreano, D. and Nolfi, S. (1997). Adaptive behavior in competing co-evolving species. In Husband, P. and Harvey, I., editors, *Proceedings of the Fourth European Conference on Artificial Life*, pages 378–387, MIT Press, Cambridge, Massachusetts.
- Floreano, D., Nolfi, S., and Mondada, F. (2001). Co-evolution and ontogenetic change in competing robots. *Robotics and Autonomous Systems*, in press.
- Floreano, D. and Urzelai, J. (1999). Evolution of Neural Controllers with Adaptive Synapses and Compact Genetic Encoding. In Floreano, D., Nicoud, J., and Mondada, F., editors, *Advances in Artificial Life*. Springer Verlag, Berlin, Germany.
- Floreano, D. and Urzelai, J. (2000). Evolutionary robots with online self-organization and behavioral fitness. *Neural Networks*, 13:431–443.
- Flotzinger, D. (1996). Evolving plastic neural network controllers for autonomous robots. Msc dissertation 9580131, COGS, University of Sussex at Brighton, UK.
- Grefenstette, J. J. and Ramsey, C. L. (1992). An approach to anytime learning. In Sleeman, D. H. and Edwards, P., editors, *Proceedings of the Ninth International Machine Learning Conference (ML'92)*, pages 189–195, Morgan Kaufmann, San Mateo, California.

- Gruau, F. and Whitley, D. (1993). Adding learning to the cellular development of neural networks. *Evolutionary Computation*, 1(3):213–233.
- Harvey, I., Husbands, P., and Cliff, D. (1994). Seeing the light: Artificial evolution, real vision. In Cliff, D. et al., editors, *From Animals to Animats III: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*, pages 392–401, MIT Press, Cambridge, Massachusetts.
- Hebb, D. O. (1949). *The organisation of behavior*. Wiley, New York, New York.
- Hinton, G. E. and Nowlan, S. J. (1987). How learning can guide evolution. *Complex Systems*, 1:495–502.
- Husbands, P. (2000). Personal communication.
- Jakobi, N. (1997). Half-baked, ad-hoc and noisy: Minimal simulations for evolutionary robotics. In Husbands, P. and Harvey, I., editors, *Proceedings of the Fourth European Conference on Artificial Life*, pages 348–357, MIT Press, Cambridge, Massachusetts.
- Kelso, S., Ganong, A., and Brown, T. (1986). Hebbian synapses in hippocampus. *Proceedings of the National Academy of Sciences USA*, 83:5326–5330.
- Lund, H. H. and Miglino, O. (1998). Evolving and breeding robots. In Husbands, P. and Meyer, J., editors, *Proceedings of the First European Workshop on Evolutionary Robotics*, pages 192–210, Springer Verlag, Berlin, Germany.
- Mataric, M. and Cliff, D. (1996). Challenges in evolving controllers for physical robots. *Robotics and Autonomous Systems*, 19(1):67–83.
- Mayley, G. (1996). Landscapes, learning costs and genetic assimilation. *Evolutionary Computation*, 4(3):213–234.
- Miglino, O., Lund, H. H., and Nolfi, S. (1996). Evolving Mobile Robots in Simulated and Real Environments. *Artificial Life*, 2:417–434.
- Nolfi, S. (1999). How learning and evolution interact: The case of a learning task which differs from the evolutionary task. *Adaptive Behavior*, 7(2):231–236.
- Nolfi, S., Elman, J. L., and Parisi, D. (1994). Learning and evolution in neural networks. *Adaptive Behavior*, 3:5–28.
- Nolfi, S. and Floreano, D. (1999). Learning and evolution. *Autonomous Robots*, 7(1):89–113.
- Nolfi, S. and Floreano, D. (2000). *Evolutionary Robotics: Biology, Intelligence, and Technology of Self-Organizing Machines*. MIT Press, Cambridge, Massachusetts.
- Nolfi, S. and Parisi, D. (1997). Learning to adapt to changing environments in evolving neural networks. *Adaptive Behavior*, 5:99–105.
- Saucy, P. (2000). *Conception d'un environnement réparti pour le contrôle de robots mobiles distants*. Ph.D. thesis 2142, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.
- Singer, W. (1987). Activity-dependant self-organisation of synaptic connections as a substrate of learning. In Changeux, J. P. and Konishi, M., editors, *The Neural and Molecular Bases of Learning*. Wiley, London, UK.
- Stanton, P. K. and Sejnowski, T. J. (1989). Associative long-term depression in the hippocampus induced by hebbian covariance. *Nature*, 339:215–218.
- Stent, G. (1973). A physiological mechanism for hebb's postulate of learning. *Proceedings of the National Academy of Sciences USA*, 70:997–1001.

- Thompson, A. (1998). On the automatic design of robust electronics through artificial evolution. In M. Sipper, D. M. and Prez-Urbe, A., editors, *Proceedings of the Second International Conference on Evolvable Systems: From biology to hardware (ICES98)*, pages 13–24. Springer-Verlag, Berlin, Germany.
- Todd, P. M. and Miller, G. F. (1991). Exploring adaptive agency II: simulating the evolution of associative learning. In Meyer, J. A. and Wilson, S. W., editors, *Proceedings of the First International Conference on Simulation of Adaptive Behavior*, pages 306–315, MIT Press, Cambridge, Massachusetts.
- Widrow, B. and Hoff, M. E. (1999). Adaptive switching circuits. In *Proceedings of the 1960 IRE WESCON Convention*, volume IV, pages 96–104, Institute of Radio Engineers, New York.
- Willshaw, D. and Dayan, P. (1990). Optimal plasticity from matrix memories: What goes up must come down. *Neural Computation*, 2:85–93.
- Yamamoto, Y., Sasaki, T., and Tokoro, M. (1999). Adaptability of darwinian and lamarckian populations toward an unknown new world. In Floreano, D., Nicoud, J., and Mondada, F., editors, *Advances in Artificial Life*. Springer Verlag, Berlin, Germany.
- Yao, X. (1993). A review of evolutionary artificial neural networks. *International Journal of Intelligent Systems*, 4:203–222.