Evolutionary Approaches to Neural Control in Mobile Robots

Jean-Arcady Meyer

Abstract— **This article is centered on the application of evolutionary techniques to the automatic design of neural controllers for mobile robots. About 30 papers are reviewed and classified in a framework that takes into account the specific robots involved, the behaviors that are evolved, the characteristics of the corresponding neural controllers, how these controllers are genetically encoded, and whether or not an individual learning process complements evolution. Related research efforts in evolutionary robotics are occasionally cited. If it is yet unclear whether such approaches will scale up with increasing complexity, foreseeable bottlenecks and prospects of improvement are discussed in the text.**

Keywords— **Evolutionary Robotics, Neural Networks, Control Architectures, Behavior.**

I. Introduction

THE design of the control architecture of a robot able to fulfil its mission in changing and possibly unpredictable environments is a highly challenging task for a human. This is due to the virtual impossibility of foreseeing each difficulty the robot will be confronted with and to the lack - as of today at least - of basic principles upon which such design might rely. For these reasons, drawing inspiration from the process of natural selection, many researchers resort to so-called *evolutionary robotics*, i.e., to the automatic design of the control architectures, and occasionally the morphology, of successive generations of robots that progressively adapt to the various challenges afforded by their environment. These research efforts call upon the definition of a *fitness function* - that assesses how well the behavior of a given robot fits its assigned mission - and upon an *encoding scheme* that relates the robot's genotype - i.e., the information that evolves from generation to generation - to its phenotype - i.e., the robot's control architecture or morphology. These research efforts also call upon some evolutionary procedure - like a *genetic algorithm* ([13]), an *evolution strategy* ([60]), or a *genetic programming* ([32]) approach - that eliminates poorly fit individuals and favors the propagation of genotypes coding for well-adapted behaviors. Usually, such an artificial selection process is performed through simulation and generates controllers with ever-increasing fitness, until it converges to some local or global optimum. Then, the best controller is downloaded into a real robot and its ability to generate the desired behavior is checked. With the simple behaviors evolved so far, results obtained in reality turn out to be similar enough to those obtained in simulation for practical purposes. However, if needed, additional evolutionary steps can be performed with the real robot, to fine-tune the controller. In some applications, evolution is performed directly on the robot from scratch.

This paper reviews specific applications of evolutionary robotics, which involve real robots, on the one hand, and control architectures implemented as neural networks, on the other hand. More general reviews are to be found in [18], [4], [22], [52], [14], [37] and [20]. Examples of evolutionary robotics applications involving non-neural controllers are [5], [56], [15] or [26].

J.A. Meyer is with the AnimatLab, Ecole Normale Supérieure, Paris, France. E-mail: meyer@wotan.ens.fr .

II. The review

Since 1994, about 30 papers have been published that describe results obtained when the neural controllers of real robots have been automatically designed through an evolutionary process. These papers are classified in Table I below, according to a general 5-dimensional framework that takes into account the specific robots involved, the behaviors that are evolved, the characteristics of the corresponding neural controllers, how these controllers are genetically encoded, and whether or not an individual learning process complements evolution.

Evolutionary robotics experiments usually involve simple mobile robots equipped with wheels and with sensors that detect obstacles or light targets. Accordingly, the behaviors that are evolved are mere exploration, obstacle-avoidance, wall-following or target-finding, under the selective pressure that dedicated fitness functions afford. For instance, to evolve the controller of a Khepera robot moving and avoiding obstacles in a given environment, the following fitness function with three components is used in [9], [44], [36], [58]:

$$F = V.(1-\sqrt{D}).(1-I) \tag{1}$$

where V is the sum of the wheel speeds at each time step, D is the signed sum of the absolute differences between the speeds of the two wheels at each time step, and I is the sum of the largest of the eight infra-red proximity sensor values at each time step.

The same behaviors are evolved in [25] with a simplified fitness function:

$$F = V.(1-\sqrt{D}) \tag{2}$$

in which the third term of equation (1) has been found to be implicit if the environment is cluttered enough, because the robot is obliged to avoid obstacles if it has to go as fast and as straight as possible. Also the D term in equation (1) is changed to the absolute value of the sum of the signed differences between wheel speeds.

In [11], [42], [43] similar behaviors are evolved endowing the Khepera robot with a simulated metabolism such as, when the robot moves away from its initial position, its energy level increases and, conversely, when it moves towards the initial position, its energy level decreases. The robot is assumed to die when it hits an obstacle or when its energy level reaches zero. Its fitness value is the maximum distance it occurred to be from its initial position during its life-time. Likewise, in other realizations, the robot is endowed with a simulated motivational system and different behaviors are sought depending upon which motivation is currently the highest ([11], [40]).

When the robot is equipped with the proper actuators, more elaborated behaviors - like area cleaning - can be evolved ([54], [49], [50], [51]). Sometimes also, besides controlling simple sensorimotor behaviors, neural controllers integrate perceptions and actions over time into some form of internal memory that is used to choose which action to perform. This is, for example, the case in [66] where an evolved controller is capable of identifying one of two landmarks based on the time-varying sonar signals received as the robot turns around the landmark. This is also the case in [24] where a robot memorizes on which side of a corridor it passed through a beam of light and, when it arrives at a T-junction at the end of the corridor, it turns on the same side and moves down the corresponding arm.

A couple of experiments have been made with legged robots and dealt with locomotion only. In [17], the fitness of each individual is determined interactively by the experimenter who assigns fitness points to various behavioral characteristics like the number of legs which oscillate, and the correctness of the corresponding frequencies, phases and couplings. On the contrary, in [12], the fitness is automatically evaluated by the forward distance the robot travels in a fixed amount of time.

Typically, the individual neurons that are used in evolutionary robotics are traditional threshold units ([39], [57]). However, a few applications ([65], [66], [12], [17]) involve neurons exhibiting an internal dynamics, according to the *leaky integrator* model [61]. In this model, the mean membrane potential m_i of a neuron N_i is governed by the equation:

$$\tau \cdot dm_i/dt = -m_i + \sum w_{i,j} x_j + I_i \qquad (3)$$

where $x_j = (1+e^{-(m_j+B_j)})^{-1}$ is the neuron's short-term average firing frequency, B_j is a uniform random variable whose mean b_j is the neuron's firing threshold, and τ is a time constant associated with the passive properties of the neuron's membrane. I_i is the input that neuron N_i may receive from a given sensor, and $w_{i,j}$ is the synaptic weight of a connection from neuron N_j to neuron N_i.

Within the so-called *Sussex approach* [20], neurons of intermediate complexity are used, which propagate excitatory and veto signals to other units after specific time-delays associated with each connection.

The architectures of the neural controllers that have been evolved to control robots range from simple perceptrons (e.g., [44], [36], [58]), to partially recurrent Elman-like [8] networks (e.g., [9], [45]), to fully recurrent continuous-time (e.g., [66], [17]) or discrete-time (e.g., [4], [19]) networks. The use of recurrent connections affords the possibility of managing an internal memory, as mentioned above (e.g., [65], [24]. Recurrent connections also make it possible to implement oscillators that are useful to control locomotion ([12], [17]).

Most often, only the neural controller of a given robot is evolved. However, in [4], [19], [23], [24], evolution also adapts the visual morphology of a robot equipped with two photoreceptors, setting their acceptance angles and their positions relative to the longitudinal axis of the robot. Depending upon which variety of individual neurons is to be included in which architecture, the genotypes used in evolutionary robotics directly code synaptic weights (and neural biases) - as in [11] and [2] for example - or they also code additional characteristics, like time delays or neuron numbers - as in [25] and [66]. However, several research efforts ([42], [7], [17]) call upon an indirect encoding scheme, according to which the genotype is a developmental program that usually acts upon a small set of initial neurons provided by the experimenter and ultimately generates a possibly complex neural network connected to the robot's sensors and actuators - thanks to various biomimetic processes like cell division, cell death, axonal growth, etc.

Finally, it should be mentioned that, in some applications, an individual learning process is added to that of evolution to improve the behavior of the robot while it experiences its environment. In [38] a given unsupervised Hebbian learning scheme involves specific connections that are genetically determined. In [10], evolution determines which Hebbian learning rule applies to each synapse in the controller. Another variety of unsupervised learning process, although calling upon a backpropagation algorithm, is used in [55].

III. Discussion

For obvious reasons of lack of hindsight, it is not yet possible to assess either the general potentialities of evolutionary robotics or the advantages of specific methodological options.

On a general level, if it is clear that the current methodology makes it possible to evolve simple sensorimotor behaviors in simple robots equipped with simple sensors and simple motors, it is difficult to foresee how this methodology will scale up and apply to more complex behaviors and more sophisticated robots. According to [36], "sufficient neurocontrollers can be surprisingly simple" and, according to [17], the evolved locomotion controller of an octopod robot is more efficient than the human-designed controller to which it has been compared. Nevertheless, it is unclear how long it will take to evolve controllers likely to compete with clever human designs, like those that implement elaborated neural architectures (e.g., [41], [28], [6]) or behavioral strategies (e.g., [1], [64]). In particular, if first steps have been made towards evolving rudimentary memories ([23], [24], [30]) that could be used to implement the

TABLE I

Authors	Robot	Behaviors	Neural Controller	Genotype	Learning
Floreano and Mondada (1994)	Khepera	Obstacle-avoidance	Two-layer Elman	Weights	No
Miglino et al.(1995a); Lund and Miglino (1996); Salomon (1996)	Khepera	Obstacle-avoidance	Two-layer Perceptron	Weights	No
Michel (1996); Michel and Collard (1996)	Khepera	Obstacle-avoidance	Recurrent network of threshold units	Developmental program	No
Jakobi et al. (1995)	Khepera	Obstacle-avoidance or Light-seeking	Recurrent network of threshold units	Weights; time delays; number of units	No
Eggenberger (1996)	Khepera	Obstacle-avoidance and Light-seeking	Recurrent network of threshold units	Developmental program	No
Mayley (1996)	Khepera	Wall- following	Two-layer Perceptron	Weights; learnable connections	Yes
Floreano and Mondada (1996a)	Khepera	Obstacle-avoidance	Three-layer Elman	Weights; learning rules	Yes
Floreano and Mondada (1996b)	Khepera with simulated battery and internal energy sensor	Obstacle-avoidance and Motivated Battery-recharge	Three-layer Elman	Weights	No
Nolfi (1996)	Khepera	Wall-avoidance and Target-detection	Two-layer Perceptron	Weights	No
Nolfi and Parisi (1997)	Khepera	Wall-avoidance and Target-finding	Two-layer Perceptron with auto- teaching units	Weights	Yes
Nolfi and Parisi (1995); Nolfi (1997a,b,c)	Khepera with gripper	Area Cleaning	Two-layer Perceptron	Weights	No
Jakobi (1997a,b)	Khepera	Memory-based Action-selection	Recurrent network of threshold units	Weights	No
Cliff et al. (1993); Harvey et al. (1994)	Gantry Robot with CCD camera	Target-seeking/ avoidance	Discrete-time Dynamical Recurrent Neural Network	Visual morphology; weights; time delays; number of units	No
Jakobi (1997a,b)	Gantry Robot with CCD camera	Target-seeking/ avoidance	Recurrent network of threshold units	Developmental program	No
Miglino et al. (1995b)	Two-wheeled Lego robot	Exploration	Four- layer Elman	Weights	No

continued on next page

TABLE I

continued from previous page

Authors	Robot	Behaviors	Neural Controller	Genotype	Learning
Yamauchi (1993)	Nomad 200	Obstacle Avoidance; Mobile Predator Avoidance	Continuous-time Dynamical Recurrent Network	Weights; time constants	No
Yamauchi and Beer (1994)	Nomad 200	Landmark identification	Continuous-time Dynamical Recurrent Network	Weights; time constants	No
Baluja (1996)	Navlab	Steering control	Three-layer Perceptron	Weights	No
Meeden (1996)	Modified toy car	Wall- avoidance and Motivated Light- seeking	Three-layer Elman	Weights	No
Gallagher et al. (1996)	Six-legged robot	Locomotion	Continuous-time Dynamical Recurrent Neural Network	Weights; time constants	No
Gruau and Quatramaran (1997)	Eight-legged robot	Locomotion	Discrete-time Dynamical Recurrent Network	Developmental program	No

simplest navigation strategy - i.e., that of *guidance* - more complex representations are required to implement higher-level strategies - like *place recognition-triggered response, topological navigation* or *metric navigation* ([64]). Moreover, to exploit topological or metric strategies to their best avail, planning capacities are required, which are themselves almost certainly not trivial to implement through an evolutionary approach.

As far as methodological options are concerned, much more experience should be accumulated before the respective advantages and drawbacks of simulations versus on-board evolution, of automatic versus interactive fitness evaluation procedures, of direct versus indirect encoding schemes, of learning versus evolution could be assessed. At least one may foresee how difficult it will be to devise fitness functions likely to automatically select complex behaviors, even if so-called *incremental approaches* - according to which the overall behavior is decomposed into simpler behavioral primitives that are successively evolved and combined together (e.g., [30], [31], [34]) - seem to be helpful. Likewise, if indirect coding affords the evolutionary process the possibility of exploring smaller search spaces than direct coding does, it is likely that devising and adjusting the corresponding genetic operators - e.g., mutations and crossovers - will prove to be much more intricate when such operators influence dynamical processes like developmental programs than when they just change static structures like the chromosomes of traditional genetic algorithms [29]. Another pending issue is that of assessing whether it is easier to evolve neural controllers than, for example, explicit control programs (e.g., [56]) or production rules systems (e.g., [5]), although it has been argued that the former approach offers over the latter the advantages of generating smoother fitness landscapes [4] and of facilitating realistic injections of noise in specific parts of the controller [18]. Likewise, it is presently unclear whether or not co-evolving controllers and robot bodies as in [4], [19], [33] entails greater synergic effects than disadvantages caused by the subsequent increase of the search space. Finally, the technology of so-called *evolvable hardware* offers great prospects of speeding up the evolutionary process because evolved hardware controllers are not *programmed* to follow a sequence of instructions, they are *configured* and then allowed to behave in real time acording to semiconductor physics ([59], [21]). If current use of such a technology to robot control do not involve neural controllers ([62], [26], [27], [47], [63]), its first application to neural network design is said to be two orders of magnitude faster than an equivalent one on a Sun SS20 computer [46]. However, here again, only accumulated experiences will make it possible to fully assess the potentialities and limitations of such an approach.

IV. Conclusions

Evolutionary approaches to neural control in mobile robots is clearly a burgeoning research area that has already produced promising results. At present, such results have been mostly limited to the evolution of simple sensorimotor mechanisms, but some success at evolving more cognitive architectures have been reported. It is yet unclear whether such automatic approaches will scale up with increasing complexity and whether they will ultimately compete with human capacities for designing efficient robots. Important steps in these directions will probably be accomplished should progress be made at adapting the fitness functions to the problems to be solved or at exploiting the synergies that interactions between development, learning and evolution certainly afford.

References

[1] Aloimonos, Y. *Active Perception.* Lawrence Erlbaum. 1993.

[2] Baluja, S. Evolution of an Artificial Neural Network Based Autonomous Land Vehicle Controller. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics.* 26, 3, 450-463, 1996.

[3] Beer, R.D. and Gallagher, J.C. Evolving Dynamical Neural Networks for Adaptive Behavior. *Adaptive Behavior.* 1, 1, 91-122, 1992.

[4] Cliff, D., Harvey, I. and Husbands, P. Explorations in Evolutionary Robotics. *Adaptive Behavior.* 2,1, 73-110, 1993.

[5] Colombetti, M. and Dorigo, M. Training agents to perform sequential behavior. *Adaptive Behavior.* 2, 3, 247-276, 1994.

[6] Corbacho, F.J. and Arbib, M.A. Learning to Detour. *Adaptive Behavior.* 3,4, 419-468. 1995.

[7] Eggenberger, P. Cell Interactions as a Control Tool of Developmental Processes for Evolutionary Robotics. In Maes, Mataric, Meyer, Pollack and Wilson (Eds.). *Proceedings of the Fourth International Conference on Simulation of Adaptive behavior: From Animals to Animats 4.* The MIT Press/Bradford Book. 1996.

[8] Elman, J.L. Finding structure in time. *Cognitive Science.* 2, 179-211. 1990.

[9] Floreano, D. and Mondada, F. Automatic Creation of an Autonomous Agent: Genetic Evolution of a Neural-Network Driven Robot. 1994. In Cliff, Husbands, Meyer and Wilson (Eds.). *Proceedings of the Third International Conference on Simulation of Adaptive behavior: From Animals to Animats 3.* The MIT Press/Bradford Book. 1994.

[10] Floreano, D. and Mondada, F. Evolution of plastic neurocontrollers for situated agents. In Maes, Mataric, Meyer, Pollack and Wilson (Eds.). *Proceedings of the Fourth International Conference on Simulation of Adaptive behavior: From Animals to Animats 4.* The MIT Press/Bradford Book. 1996.

[11] Floreano, D. and Mondada, F. Evolution of Homing Navigation in a Real Mobile Robot. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics.* 26, 3, 396-407. 1996.

[12] Gallagher, J.C., Beer, R.D., Espenschield, K.S. and Quinn, R.D. Application of evolved locomotion controllers to a hexapod robot. *Robotics and Autonomous Systems.* 19, 95-103. 1996.

[13] Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison-Wesley. 1989.

[14] Gomi, T. and Griffith, A. Evolutionary Robotics - An Overview. *Proceedings of the IEEE 3rd International Conference on Evolutionary Computation.* IEEE Society Press. 1996.

[15] Gomi, T. and Ide, K. Emergence of gaits of a legged Robot by Collaboration through Evolution. *Proceedings of the International Symposium on Artificial Life and Robotics.* Springer Verlag. 1997.

[16] Gruau, F. Automatic definition of modular neural networks. *Adaptive Behavior.* 3, 2, 151-183. 1995.

[17] Gruau, F. and Quatramaran, K. Cellular Encoding for Interactive Evolutionary Robotics. In Husbands and Harvey (Eds.). *Fourth European Conference on Artificial Life.* The MIT Press/Bradford Books. 1997.

[18] Harvey, I., Husbands, P. and Cliff, D. Issues in Evolutionary Robotics. In Roitblat, Meyer and Wilson (Eds.). *Proceedings of the Second International Conference on Simulation of Adaptive behavior: From Animals to Animats 2.* The MIT Press/Bradford Book. 1992.

[19] Harvey, I., Husbands, P. and Cliff, D. Seeing The Light: Artificial Evolution, Real Vision. In Cliff, Husbands, Meyer and Wilson (Eds.). *Proceedings of the Third International Conference on Simulation of Adaptive behavior: From Animals to Animats 3.* The MIT Press/Bradford Book. 1994.

[20] Harvey, I., Husbands, P., Cliff, D., Thompson, A. and Jakobi, N. Evolutionary Robotics: The Sussex Approach. *Robotics and Autonomous Systems.* 20, 205-224. 1997.

[21] Higuchi, T., Iwata, M. and Liu, W. (Eds.). *Evolvable Systems: From Biology to Hardware.* Springer Verlag. 1997.

[22] Husbands, P., Harvey, I., Cliff, D., and Miller, G. The Use of Genetic Algorithms for the Development of Sensorimotor Control Systems. In Nicoud and Gaussier (Eds.). *From Perception to Action.* IEEE Computer Society Press. 1994.

[23] Jakobi, N. Half-baked, Ad-hoc and Noisy: minimal Simulations for Evolutionary Robotics. In Husbands and Harvey (Eds.). *Fourth European Conference on Artificial Life.* The MIT Press/Bradford Books. 1997.

[24] Jakobi, N. Evolutionary Robotics and the Radical Envelope of Noise Hypothesis. *Adaptive Behavior.* 6,1, 131-174. 1997.

[25] Jakobi, N., Husbands, P. and Harvey, I. Noise and the reality gap: The use of simulation in evolutionary robotics. In Moran, Moreno, Merelo and Chacon (Eds.). *Advances in Artificial Life: Proceedings of the Third European Conference on Artificial Life.* Springer Verlag. 1995.

[26] Keymeulen, D., Durantez, M., Konaka, M., Kuniyoshi, Y. and Higuchi, T. An Evolutionary Robot Navigation System Using a Gate-Level Evolvable Hardware. In Higuchi, Iwata and Liu (Eds.). *Evolvable Systems: From Biology to Hardware.* Springer Verlag. 1997.

[27] Keymeulen, D., Konaka, K., Iwata, M., Kuniyoshi, Y. and Higuchi, T. Learning using gate-Level evolvable hardware.. Proceedings of the 6th European Workshop on Learning Robots. Brighton. 1997.

[28] Klopf, A.H., Morgan, J.S. and Weaver, S.E. A Hierarchical Network of Control Systems that Learn: Modelling Nervous System Function During Classical and Instrumental Conditioning. *Adaptive Behavior.* 1,3, 263-319. 1993.

[29] Kodjabachian, J. and Meyer, J.A. Evolution and development of control architectures in animats. *Robotics and Autonomous Systems.* 16, 161-182. 1995.

[30] Kodjabachian, J. and Meyer, J.A. Evolution and Development of Modular Control Architectures for 1-D Locomotion in Six-legged Insects. Submitted.

[31] Kodjabachian, J. and Meyer, J.A. Evolution and Development of Neural Controllers for Locomotion, Gradient-Following, and Obstacle-Avoidance in Artificial Insects. *IEEE Transactions on Neural Networks.* In Press.

[32] Koza, J. *Genetic Programming.* The MIT Press. 1992.

[33] Lee, W.P., Hallam, J. and Lund, H.H. A hybrid GP/GA approach for Co-evolving Controllers and Robot Bodies to Achieve Fitness-specific Tasks. *Proceedings of the 3th IEEE International Conference on Evolutionary Computation.* IEEE Computer Society Press. 1996.

[34] Lee, W.P., Hallam, J. and Lund, H.H. Learning Complex Robot Behaviours by evolutionary Approaches. *Proceedings of the 6th European Workshop on Learning Robots.* Brighton. 1997.

[35] Lund, H.H. and Hallam, J. Sufficient Neurocontrollers can be Surprisingly Simple. Research Paper 824. Department of Artificial Intelligence. Edinburgh University. 1996.

[36] Lund, H.H. and Miglino, O. From Simulated to Real Robots. *Proceedings of the 3rd IEEE International Conference on Evolutionary Computation.* IEEE Computer Society Press. 1996.

[37] Mataric, M. and Cliff, D. Challenges in evolving controllers for physical robots. *Robotics and Autonomous Systems.* 19, 67-83. 1996.

[38] Mayley, G. The Evolutionary Cost of Learning. In Maes, Mataric, Meyer, Pollack and Wilson (Eds.). *Proceedings of the Fourth International Conference on Simulation of Adaptive behavior: From Animals to Animats 4.* The MIT Press/Bradford Book. 1996.

[39] McClelland, J.L. and Rumelhart, D.E. *Parallel Distributed Processing. Vol. 1.* The MIT Press/Bradford Books. 1986.

[40] Meeden, L.A. An Incremental Approach to Developing Intelligent Neural Network Controllers for Robots. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics.* 26, 3, 474-485. 1996.

[41] Mel, B.W. *Connectionist Robot Motion Planning. A Neurally-inspired Approach to Visually-Guided Reaching.* Academic Press. 1990.

[42] Michel, O. An Artificial life Approach for the synthesis of Autonomous Agents. In Alliot, Lutton, Ronald, Schoenauer and Snyers (Eds.). *Artificial Evolution.* Springer Verlag. 1996.

[43] Michel, O. and Collard, P. Artificial Neurogenesis: An application to Autonomous Robotics. In Radle (Ed.). *Proceedings of The 8th. International Conference on Tools in Artificial Intelligence.* IEEE Computer Society Press. 1996.

[44] Miglino, O., Lund, H.H. and Nolfi, S. Evolving Mobile Robots in Simulated and Real Environments. *Artificial Life.* 2, 417-434. 1995.

[45] Miglino, O., Nafasi, K. and Taylor, C. Selection for Wandering Behavior in a Small Robot. *Artificial Life.* 2, 101-116. 1995.

[46] Murakawa, M., Yoshizawa, S., Kajitani, I and Higuchi, T. On-line Adaptation of Neural Networks with Evolvable Hardware. 9em Proceedings of the Seventh International Conference on Genetic Algorithms. Morgan Kaufmann. 1997.

[47] Naito, T., Odagiri, R., Matsunaga, Y., Tanifuji, M. and Murase, K. Evolution of a Logic Circuit Which Controls an Autonomous Mobile Robot. In Higuchi, Iwata and Liu (Eds.). *Evolvable Systems: From Biology to Hardware.* Springer Verlag. 1997.

[48] Nolfi, S. *Adaptation as a more powerful tool than decomposition and integration.* Technical Report, Institute of Psychology, CNR, Rome. 1996.

[49] Nolfi, S. Using Emergent Modularity to Develop Control Systems for Mobile Robots. *Adaptive Behavior.* 5,3/4, 343-363. 1997.

[50] Nolfi, S. Evolving Non-Trivial Behavior on Autonomous Robots: Adaptation is More Powerful Than decomposition and Integration. In Gomi (Ed.). *Evolutionary Robotics. From Intelligent Robots to Artificial Life (ER'97).* AAI Books. 1997.

[51] Nolfi, S. Evolving non-Trivial Behaviors on Real Robots: a garbage collecting robot. *Robotics and Autonomous Systems.* In Press.

[52] Nolfi, S., Floreano, D., Miglino, O. and Mondada, F. How to evolve autonomous robots: Different approaches in evolutionary robotics. In Brooks and Maes (Eds.). *Artificial Life IV.* The MIT Press/Bradford Books. 1994.

[53] Nolfi, S. and Parisi, D. Auto-teaching: Networks that develop their own teaching input. In Deneubourg, Bersini, Goss, Nicolis and Dagonnier (Eds.). *Proceedings of the Second European Conference on Artificial Life.* Free University of Brussels. 1993.

[54] Nolfi, S. and Parisi, D. Evolving non-trivial behaviors on real robots: an autonomous robot that picks up objects. In Gori and Soda (Eds.). *Topics in Artificial Intelligence. Proceedings of the Fourth Congress of the Italian Association for Artificial Intelligence.* Springer Verlag. 1995.

[55] Nolfi, S. and Parisi, D. Learning to Adapt to Changing Environments in Evolving Neural Networks. *Adaptive Behavior.* 5, 1, 75-98. 1997.

[56] Nordin, P. and Banzhaf, W. An On-Line Method to Evolve Behavior and to Control a Miniature Robot in Real Time with Genetic Programming. *Adaptive Behavior.* 5, 2, 107-140. 1996.

[57] Rumelhart, D.E. and McClelland, J.L. *Parallel Distributed Processing. Vol. 2.* The MIT Press/Bradford Books. 1986.

[58] Salomon, R. Increasing Adaptivity through Evolution Strategies. In Maes, Mataric, Meyer, Pollack and Wilson (Eds.). *Proceedings of the Fourth International Conference on Simulation of Adaptive behavior: From Animals to Animats 4.* The MIT Press/Bradford Books. 1996.

[59] Sanchez, E. and Tomassini, M. (Eds.). *Towards Evolvable Hardware. The Evolutionary Engineering Approach.* Springer Verlag. 1996.

[60] Schwefel, H.P. *Evolution and Optimum Seeking.* Wiley. 1995.

[61] Segev, I. Simple neuron models: Oversimple, complex and reduced. *Trends in Neurosciences.* 15,11, 414-421. 1992.

[62] Thompson, A. Evolving electronic robot controllers that exploit hardware resources. In Moran, Moreno, Merelo and Chacon (Eds.). *Advances in Artificial Life: Proceedings of the Third European Conference on Artificial Life.* Springer Verlag. 1995.

[63] Thompson, A. Artificial Evolution in the Physical World. In Gomi (Ed.). *Evolutionary Robotics. From Intelligent Robots to Artificial Life (ER'97).* AAI Books. 1997.

[64] Trullier, O., Wiener, S. Berthoz, A. and Meyer, J.A. Biologically Based Artificial navigation Systems: Review and Prospects. *Progress in Neurobiology.* 51, 483-544. 1997.

[65] Yamauchi, B. *Dynamical neural networks for mobile robot control.* Naval Research Laboratory Memorandum Report AIC-033-93. Washington. 1993.

[66] Yamauchi, B. and Beer, R. Integrating Reactive, Sequential, and Learning Behavior Using Dynamical Neural Networks. In Cliff, Husbands, Meyer and Wilson (Eds.). *Proceedings of the Third International Conference on Simulation of Adaptive behavior: From Animals to Animats 3.* The MIT Press/Bradford Book. 1994.

Jean-Arcady Meyer is a graduate engineer of the Ecole Nationale Supérieure de Chimie de Paris, graduate in Human Psychology, graduate in Animal Psychology, and holds a French PhD in Biology. He is currently Research Director at the CNRS and heads the AnimatLab of the Ecole Normale Supérieure in Paris. His main scientific interests are the interactions of learning, development and evolution in adaptive systems, both natural and artificial. Dr. Meyer is the Editor-in-Chief of the journal Adaptive Behavior.