Presented at the 2nd International Conference on Artificial Intelligence in Engineering and Technology (ICAIET 2004), volume 1, pages 7-13, Kota Kinabalu, Sabah, Malaysia, August 2004.

Darwin + Robots = Evolutionary Robotics: Challenges in Automatic Robot Synthesis

Jason Teo

Artificial Intelligence Group, School of Engineering and Information Technology
Universiti Malaysia Sabah, Locked Bag No. 2073, 88999 Kota Kinabalu, Sabah, Malaysia
Tel: +6-088-320000 ext. 3104, Fax: +6-0888-320348, E-mail: jtwteo@ums.edu.my

Abstract

This paper reviews the use of artificial evolution as a means for automatic generation of locomotion controllers for physical robots and autonomous agents from the perspectives of evolutionary objectives and control architecture. An overview beginning with the pioneering works in evolutionary robotics is given, leading up to the latest state-of-the-art research in these fields. Some key shortfalls in mainstream approaches are identified, concluding with some promising research directions.

Keywords:

Evolutionary robotics, evolutionary artificial neural networks, fitness functions, controller architecture, autonomous robots.

Introduction

Evolutionary robotics is defined to be the synthesis of autonomous robots using artificial evolutionary methods [1]. An early review of this field of research is given by Mataric [2] where the majority of studies focused mainly on the evolution of control structures only. A more recent overview highlights the move of evolutionary robotics into evolving both the control and morphology of robots where the interplay between brain and body is considered to be a crucial factor in the successful synthesis of autonomous robots [3]. A thorough treatment of the field can be found in the seminal textbook written by Nolfi and Floreano [1] on this subject.

As pointed out by Harvey [4], the design of controllers for robots is a complex task not suited to human divide-and-conquer design strategies. There are 3 major problems: (1) it is not obvious how the controller system should be decomposed, (2) interactions are not limited to direct connecting links but are also mediated through the environment, and (3) interactions between sub-parts grows exponentially as system complexity increases. Thus, evolutionary approaches to controller design are desirable, where the only benchmark is the overall behavior that should be achieved by the system.

However, Ronald and Sipper [5] recently pointed out that emergence stemming from the use of biologically-inspired solutions in engineering problems may be problematic because unexpected and sometimes unwanted results or behaviors might arise. Using the so-called emergence test, it was claimed that evolutionary robotics exhibited mild emergence where the degree of surprise is limited to well-defined boundaries *unsurprising surprise*. On the other hand, traditional hard-wired engineering solutions exhibited no surprise (*unsurprising*) while artificial life exhibits a very high degree of surprise (*surprising surprise*). Nonetheless, it was surmised that emergence in engineering solutions that draw on inspirations from nature such as evolutionary robotics and the related reliability issues are unavoidable consequences if the desire is to design smart, adaptive and evolvable machines. In general, evolutionary robotics can be grouped into three main categories, those involving the evolution of (1) wheeled, (2) legged, and (3) abstract robots.

Wheeled Robots

A hybrid genetic programming (GP)/genetic algorithm (GA) methodology was used to evolve both the controller and parameters of a wheeled robot's morphology in simulation [6]. The controller consisting of a tree-like program was evolved using the GP part of the system while morphological parameters such as the robot's body size, wheel radius and wheel base size encoded in a linear string of real numbers were evolved using the GA part of the system. Individuals were assessed for obstacle avoidance behaviors using a fitness function that combined multiple terms such as distance from obstacles, forward speed and rotating speed into a single objective. It was claimed to be the first study which co-evolved both the controller and morphology of robots and concluded that because the evolved controller only functioned within the co-evolved body, the evolution of the body component played a significant role in the success of the evolutionary process. An island-GA model was used to maintain genetic diversity during the evolutionary process. In a related study using simulations, Khepera wheeled robots were shown to require only simple perceptron controllers that directly connected sensors to motors for evolving behaviors such as exploration and homing [7]. It was claimed that the robot's perception of its environment's geometries allowed time-related components to be encoded without requiring any recurrent connections in the controller. GP alone has also been used to evolve controllers for Khepera robots for obstacle avoidance and object tracking behaviors utilizing a combination of simulated and real-world testing of evolved controllers [8].

The Species Adaptation Genetic Algorithm (SAGA) algorithm was used to evolve both the controller and visual morphology parameters for simple navigational tasks in a two-wheeled mobile autonomous robot [9]. Network topologies with variable number of nodes as well as feed-forward and recurrent connections between input, hidden and output units were used. The desired behavior was evolved within 50-100 generations using 40-60 individuals that were evaluated using a simple single-objective distance-based fitness function. SAGA allows for increases in length to genotypes and hence it was argued that it permitted incremental evolution to occur during the evolutionary process. Conversely, Eggenberger [10] reported the use of biological cell differentiation techniques in order to reduce the length of the genotype encoding when evolving neural network controllers for Khepera robots in simulation. It was claimed that using such a developmental method, the genome need not necessarily increase in length whenever the number of neurons increased since no specific data relating to the presence or otherwise of neurons need to be stored in the genome, which will now be specified as part of the cell differentiation process rather than being directly encoded for in the genome. This cell differentiation system has subsequently been used to evolve only the morphologies of static 3D virtual organisms [11] and more recently to grow the connectivity of a neural network for controlling a foveating retina of a real physical robot [12].

Related work with wheeled robots have also shown promising results in robustness and the ability to cope with changing environments by evolving plastic individuals that are able to adapt both through evolution and lifetime learning [1,13,14]. Neural networks capable of producing a number of different reactive navigation behaviors were generated using evaluation functions that typically included different terms for rewarding speed, wall avoidance and straight-line motion combined into a single objective. These networks were fixed in architecture, typically using only a single hidden node and having full recurrency between nodes. Instead of evolving the synaptic weights, the learning rules governing the behavior of individual synapses were evolved when generating a neural network controller for Khepera robots. It was demonstrated that the evolved controllers were adaptive to changes in the environment due to their synaptic plasticity. Lifetime learning or ontogenetic adaptation has several adaptive functions within evolution: (1) allowing for individuals to adapt to fast-changing environmental conditions, (2) channelling information extracted from the environment to evolution, (3) helping to guide evolution, (4) reducing genotype length, and (5) maintaining genetic diversity [14]. Learning and evolution were shown to be able to solve tasks that evolution alone could not solve. Performance increases were also noticed even when the learning tasks differed from the selection tasks. Learning individuals were thus better adapted to changing environments than non-learning individuals. Interaction between learning and evolution deeply altered both these processes in that learning enabled evolution to extract supervision information from the environment. In terms of generality, plastic-general individuals required less complex control systems compared to full-general individuals. Ontogenic adaptation has also been studied in a competitive co-evolutionary context of predator-prey simulations using Khepera robots [15].

Pure reactive agents that do not use any internal representation were shown to able to solve complex tasks through the use of sensory-motor coordination only [16]. By exploiting agent-environment interactions, these embodied robots were able to coordinate perception and action that enabled them to perform complex tasks without needing to react differently to the same sensory states in different contexts. The experiments involving physical agents were carried out using Khepera robots and neural networks weights were evolved for the control of the agents. Sensory-motor coordination allowed the robots to (1) select the most effective feedback, (2) simplify harder tasks, (3) exploit emergent behaviors, and (4) exploit environmental constraints. Pure reactive agents although effective were found to be sub-optimal in most conditions. As a remedy, it was suggested that more complex behaviors could be allowed to emerge through a simple process of adding internal representations to the existing reactive behaviors.

In a departure from classical connectionist models, Floreano [17] recently demonstrated the use of evolutionary spiking neurons for the control of an autonomous microbot. A single "spike" in a spiking neural network is a discrete binary event that simply encodes whether a stimulus is present or absent. Instead of using conventional non-linear, real-valued sigmoidal activation functions, the use of spiking neurons in neural circuits were shown to transfer easily to microcontrollers by virtue of their binary nature, which can be mapped onto low-level digital circuits using only a few logic operations such as *AND* and *NOT*. In an earlier study, it was shown that viable controllers were easier to evolve using spiking neurons than sigmoidal neurons for a vision-based navigation task of a Khepera robot [18].

The control structures consisting of fixed 2-layer ANNs with no hidden layer for a population of robots was evolved using a fully decentralized evolutionary algorithm (EA) [19]. The EE (Embodied Evolution) methodology was defined as conducting evolution in a group of real physical robots where evaluation, selection, and reproduction took place by and between robots in a distributed, asynchronous and autonomous manner. The robots were simple two-wheeled self-designed mobile agents with inter-agent communication capabilities. Evolved controllers outperformed hand-designed controllers for a phototaxis task.

A gaseous signalling mechanism was used in the GasNet algorithm for generating robot controllers with variable spatio-temporal network properties in visual discrimination and navigation tasks [20]. The networks had variable numbers of internal nodes which were arbitrarily recurrent and possessed properties which were modulated according to the diffusion of simulated gases. The fitness of generated controllers was evaluated using a single function that combined the weighted sum of navigational scores. Recent related work has also extended the family of GasNet neural networks to include more details of biological gaseous signalling mechanisms into two new versions called the *plexus* and *receptor* models, which were shown to be more evolvable than the earlier version of GasNet [21].

Legged Robots

The pioneering work of Beer and Gallagher [22] documented the use of GA to evolve continuous-time recurrent neural networks for controlling the legged locomotion of a hexapod insect, although this study was conducted using a highly simplified physics model. It was shown in a later study that the evolved controllers could still perform the locomotion successfully when transferred to a real hexapod robot [23]. Related studies based on this simplified six-legged hexapod model have been conducted to investigate the evolution of neural network architectures rather than synaptic weights alone using a developmental scheme specified by the Simple Geometry Oriented Cellular Encoding (SGOCE) algorithm [24].

The control mechanism based on different models of fully-recurrent neural networks for generating legged locomotion for a range of fixed morphology robots were evolved in simulation using a simple GA by Reeve [25]. It was found that simple single-termed fitness measures based on performance attributes such as speed was sufficient to generate the desired behavior and that more complex fitness measures relating to inner workings of neurons and joints were not advantageous. It was also found that higher-order neural networks were significantly better at performing the required tasks and that very densely connected controllers performed better than sparsely connected ones.

A dynamically-rearranging neural network (DRNN) was evolved to act as a controller for legged locomotion in a simulated biped robot [26]. Generated controllers were assigned fitness values based on a single-objective function of horizontal movement achieved. Neuromodulators were used to dynamically change synaptic weights as well as network architecture by activating and blocking neurons and synapses. However, it was observed that many of the evolved controllers did not actually make use of the modifiable synaptic weights, in other words normal neural networks with fixed synaptic weights would have sufficed. Nonetheless, it was claimed that the DRNN would have exhibited superior performance in a changing environment due to their polymorphic characteristics although this was not investigated using the biped robot. In related work using a simulated quadruped robot, a DRNN was again evolved to act as a controller for legged locomotion [27]. It was claimed that the controllers generated were adaptive to changes in the environment (retardant forces and uneven slopes) due to the neuromodulations present in the DRNN. However, as no analysis was provided on the actual dynamics of the neuromodulators during the legged locomotion of the quadruped, it remains unclear what roles these elements actually played towards the generation of a successful legged locomotion in the changing environments.

Both the controller and morphology of a biped robot were evolved using a GA with a simple single-objective fitness function based on horizontal distance travelled [28]. It was claimed that the experiments produced the first reported results of stable bipedal locomotion achieved through the optimization of both controller and morphology. An interesting point to note was that only 6 out of the 60 evolutionary runs were successful in evolving a stable gait. The architecture of the partially-recurrent neural networks that were used as the controllers remained fixed with only the synaptic weights being evolved, which utilized three hidden units in its internal layer. Also, only certain parameters of robot's morphology were allowed to be modified during evolution. A related study using similar biped robots where both the controller and morphology were co-evolved found that the inclusion of certain morphological parameters allowed for fitter individuals to be discovered by evolutionary search [29]. It was shown that fitter individuals did not arise simply because a better morphology was found but rather the addition of morphological parameters into the genotype space allowed for extra-dimensional bypasses to be formed in the higher dimensional search space, thereby allowing the evolutionary search to find these fitter individuals. This phenomenon facilitated the connection of otherwise isolated adaptive peaks in the objective space, making it easier for the evolutionary search process to proceed smoothly from one adaptive peak to the next.

Central pattern generators (CPGs) were evolved as controllers for generating planar walking behaviors in two different physically simulated bipeds [30]. It was shown that using the appropriate mechanical construction, Hopfield neural network controllers and optimization through a GA with a single-objective distance-based fitness function, minimal bipedal locomotion can be achieved by CPGs that do not require sensor inputs. These networks did not require any sensor input, which only required four internal nodes and six actuator nodes. In the second more sophisticated biped, incremental evolution was used where a weak stabilizing controller was used during the initial stages of evolution and later removed after a certain fitness level was achieved. The lower portions of the more sophisticated biped's legs were implemented as passive limbs to allow for a more anthropomorphic gait to emerge. Only 10% of the first biped's evolutionary runs produced successful controllers whereas 80% of the second biped's evolutionary runs produced successful controllers. However no analysis was given on whether the two search spaces differed significantly in terms of optimization difficulty.

In a related study, CPGs were again evolved to generate bipedal locomotion in a simulated robot in a real-time physics environment [31]. Once more, it was shown that no sensory inputs were necessary to generate successful straight-line walking behavior although this was achieved only on a homogenous planar surface. It was suggested that the fitness landscape underlying the evolutionary search space of the fully-recurrent ANN architecture is very smooth leading to successful evolution of controllers despite using only a very simple single-objective fitness function based on a combination of two objectives of maximizing distanced travelled from origin and minimizing occurrences of falling below a certain height threshold for the robot’s center of gravity. However it was also reported that only 10% of the evolutionary runs resulted in stable controllers and that an additional fitness term that rewarded cyclic activity in the ANN was necessary to improve the success rate. The authors also noted a shortfall in the experimental setup in that the effect of network size on the efficiency of the approach was not studied. A number of important contributions of the evolutionary robotics approach to designing controllers for legged locomotion of embodied robots were highlighted: (1)

fully automated process that allows for changes or additions to the creature's structure to be accommodated very easily through re-evolution, (2) diversity of solutions, and (3) relatively cheap evolutionary computational requirements.

Real physical robots have also been used to study the generation of legged locomotion using EAs. Online evolution was used by Gruau [32] as well as Gomi and Ide [33] to generate static gaits for an octopod robot, and by Hornby and his co-researchers to generate dynamic gaits for a Sony quadruped robot [34] as well as for the Sony entertainment robot dog AIBO [35]. The cellular encoding method of Gruau was used to evolve not only the weights but also the architecture of the neural network controller for the octopod robot and also relied on interactive user assignment of fitness values rather than integrating a fully automated fitness assignment into the artificial evolutionary process [32]. Jakobi [36] utilized his "minimal simulation" method to also evolve gaits in simulation for the same octopod robot in order to reduce the time requirements of evolution on the real physical robot.

Abstract Robots

The variable-topology neural network controller and visual morphology for visually guided behaviors in a specialized gantry robot was evolved using the SAGA algorithm for a visual discrimination task [37]. Minimal vision systems and small networks were found to be sufficient for generating the required behaviors using a weighted sum combination of navigational scores as the evaluation function. Small population sizes and small number of generations were also sufficient for successfully evolving these controllers. A good choice of control system primitives was suggested as the main reason for the success of these evolutionary runs. Work has also been carried where only the morphology of a compound eye on an abstract robot was evolved while the 2-layer, perceptron-like neural network controller was kept fixed [38].

GP was used to evolve controllers for a robot with manually reconfigurable morphology called a random morphology robot (RM robot) [39]. Evaluation of controller fitness was carried out using a single-objective function consisting of weighted scores for the robot's speed and distance. The fitness landscape was found to be highly dynamic because the robot moves around on a carpeted floor and hence encounters situations with different levels of difficulties arising from the directionality of individual carpet strands. It was shown that discrimination between good and bad individuals was hard during certain periods of the evolutionary process where the noise level was high. Hence it was proposed that reference individuals be employed to enable a differential fitness value to be calculated for evolving individuals in order to better capture the actual performance of individuals throughout the highly variable evolutionary periods. Nevertheless, it was later observed that there were periods where the fitness landscape oscillated, which created a problem for the proposed relative fitness methodology as well.

Lipson and Pollack [40] combined both simulated and physical approaches for evolving simple robots composed of bars, actuators and arbitrarily-recurrent artificial neurons for the single objective of maximizing horizontal distance moved. The authors claimed that to fully realize artificial life, autonomy must be achieved not only at the level of power and behavior but also at the levels of design and fabrication. They demonstrated this point in their experiments where artificial evolution was conducted to automatically design abstract robots that could perform locomotion in simulation and then the best virtual designs were fabricated into real robotic body parts using 3D thermoplastic solid printing techniques. The results from testing the physical versus the virtual robots showed that in one case, the distance travelled was almost identical while in the two other cases, the distances travelled were quite dissimilar although it was argued that the overall control and mechanics of the motion were still maintained when moving from simulation to reality.

Evolvable Hardware

Evolvable hardware circuits in the form of field programmable gate arrays (FPGAs) were utilized to evolve obstacle avoidance controllers for Khepera robots [41] and was claimed to be the first example of intrinsic hardware evolution [4], where every actual hardware specified during the evolutionary process was tested in situ rather than in simulation. Another series of studies also utilized FPGAs as evolvable controllers for producing visual tracking and obstacle avoidance behaviors in Khepera robots [42]. Solutions generated were evaluated using a fitness function that took a weighted sum combination of two objectives of minimizing the robot-target distance and minimizing the number of steps required to complete the task. It has been argued that *true evolvable hardware* should allow for both control circuits and body plans to be evolved [43]. Such true evolvable hardware using a modified version of the Khepera robot with a reconfigurable auditory morphology was developed by Lund et. al. [43] as a framework for studying the evolution of phonotaxis in crickets although no result from actual experimentation was reported.

Challenges in Evolutionary Robotics

Fitness Functions

The survey conducted showed that the research into evolving robots has focused mainly on generating the desired behavior using single-objective fitness functions. These evaluation functions typically consist only of a single term for assigning the fitness of individuals generated [25,26,28,31,40,44] or a combination of multiple terms into a single weighted objective when the desired behavior cannot be achieved with simpler single-termed functions [6,13,29,30,39,42]. A Pareto multi-objective optimization approach involving optimization of explicitly distinct objectives have not been explored yet thus far for artificial creature evolution. Such an investigation might very well reveal significant advantages over standard single-objective EAs in terms of the evolutionary optimization process itself in addition to the possibility of generating greater varieties of creature morphologies and behaviors by virtue of multiple evolutionary objectives versus single-termed evolutionary systems. Hence, the computational time required to conduct these evolutionary runs can be significantly reduced using a multi-objective approach compared to single-objective approach since an entire set of controllers with varying network sizes and locomotion capabilities can be generated in a single run, allowing for comparisons between creatures with different abilities and controllers to be made after just a single run is conducted for each type of creature. This represents a significant advantage over single-objective evolutionary systems that need to be re-run multiple times in order to test the effect of other factors such as number of hidden units on the locomotion capability of the virtual robots [44]. Such a setup would require a significantly larger number of evolutionary runs before a suitable set of controllers with different network characteristics and locomotion capabilities can be obtained in order to conduct comparisons between different creature designs.

Furthermore, there are a number of disadvantages associated with using a weighted sum approach compared to a Pareto approach. Firstly, the weighted sum method would only be able to generate a single Pareto solution in a single run compared to an entire set of Pareto solutions in a single run using the Pareto approach, meaning multiple runs will be required to generate a Pareto-front when using the weighted sum method. Secondly, the determination of the weights is arbitrary in a weighted sum method. Some form of hand-tuning these weights will need to be carried out in order to obtain good results and as such, extra runs will again be required compared to the Pareto approach. Thirdly, the different objectives combined in a weighted sum method are assumed to be somehow commensurable, that is the objectives can be measured in the same units. In the case where they are not, the use of the correct relative weights will be necessary to overcome this problem. Again, the Pareto approach does not require any such assumption to hold true since it treats each objective independently from the other. Lastly, the weighted sum method assumes that the Pareto-front of the multi-objective optimization problem is of a convex nature. If the Pareto-front of the multi-objective optimization problem is actually non-convex, then the Pareto solutions generated by the weighted sum method will result in a discontinuous Pareto-front since the single-objective hyperplane will not be able to sample the non-convex regions of the Pareto-front. As such, in order to use a weighted sum method, the experimenter will first need to ascertain whether the particular problem is convex or otherwise, and no such information is usually available until the actual experiments are carried out and the Pareto-front plotted. Conversely, knowledge of such properties about the multi-objective optimization problem is not required since the solutions generated using a Pareto approach is not constrained or limited to a specific Pareto-front of the multi-objective optimization problem.

Network Architecture

The choice of ANN architectures used for controller evolution is normally made without proper explanation to the reader when evolving both simulated [25,28,29,30,31,44] and physical robots [7,13,14,16]. Usually some form of recurrency is used in the ANN, either partially [28,29,44] or fully [13,14,16,25,30,31]. On the other hand, simple direct connections between sensor inputs and motor outputs have also proven to be sufficient for evolving robots controllers with simple behaviors that can accomplish the set task [16,19,38,43]. As such, it remains unclear from the body of literature what types of ANN architecture should be used to evolve controllers for embodied robots.

Furthermore, there is the question of what is the minimum hidden layer size required to produce locomotion controllers in physical robots and autonomous agents. Most works in this area simply choose an arbitrary number of hidden units to be included in the network control architecture, which then remain fixed throughout the evolutionary process [13,14,28,30,31,44]. The capacity of an ANN is determined by its so-called Vapnik-Chervonenkis (VC) dimension [45], which in turn is determined by the number of free parameters in the network such as the connection weights for feed-forward ANNs [46]. One way to control the weights is by controlling the number of hidden units present in the ANN. Hence, the importance of implementing a suitably-sized hidden layer within the ANN architecture needs to be ascertained. Although some papers report the inclusion of number of hidden nodes as an evolvable network parameter within their artificial evolutionary system [20,29,40], none of these experiments explicitly impose evolutionary pressures on minimizing the size of the hidden layer. Firstly, finding the ANN controller with the minimum network size will reduce the amount of computation that needs to be carried out by the artificial creature's controller, thereby further enhancing its efficiency during operation. Secondly, to be able to use the controllers as some type of complexity measure for the comparison between evolved agents, we need to ensure that the amount of redundancy in the network is minimized as far as possible in order to avoid false indications given by large redundant networks. Thirdly, although redundancy may be beneficial for life-long learning, we need to avoid evolving networks with unseen redundancy to be able to reduce the risk of unpredictable behavior. Redundancy can be later added manually, with its corresponding effects analyzed by the designer. Thus, minimizing the number of redundant hidden units can reduce the amount of "*surprise*" [5] arising from the use of biologically-inspired solutions since some of this "*surprise*" may result in undesirable behaviors that can reduce the reliability of such systems (see Introduction). Hence, the inclusion of an explicit second objective which minimizes the size of the hidden layer would be highly beneficial to this end, which can again be achieved by adopting a Pareto multi-objective evolutionary approach.

Conclusion

A literature review of the related fields of evolutionary robotics was presented in this chapter. A comprehensive survey of the various methods employed for evolving both simulated and real physical robots was given. Key shortfalls in terms of fitness function design and controller architecture were highlighted to emphasize the need to re-focus some of the research efforts towards these answering some parts of these fundamental open questions in the development of automatic generation systems for robot design. Finally, some promising research directions in the development of artificial evolutionary systems that use a Pareto multi-objective optimization approach as well as evolvable network architectures were presented.

Acknowledgments

The author appreciates the comments from Dr. Hussein Abbass on an earlier draft of this paper.

References

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

[2] Mataric, M., and Cliff, D., 1996. Challenges in Evolving Controllers for Physical Robots. *Robotics and Autonomous Systems* 19:67-83.

[3] Nolfi, S., and Floreano, D. 2002. Synthesis of Autonomous Robots Through Evolution. *Trends in Cognitive Science* 6(1):31-36.

[4] Harvey, I. 1997. Artificial Evolution and Real Robots. *Artificial Life and Robotics* 1(1): 35-38.

[5] Ronald, E., and Sipper, M. 2001. Surprise Versus Unsurprise: Implications of Emergence in Robotics. *Robotics and Autonomous Systems* 37:19-24.

[6] Lee, W., Hallam, J., and Lund, H. 1996. A Hybrid GP/GA Approach for Co-evolving Controllers and Robot Bodies to Achieve Fitness-specific Tasks. 3rd IEEE International Conference on Evolutionary Computation, 384-389. Piscataway, NJ: IEEE Press.

[7] Lund, H., and Hallam, J. 1997. Evolving Sufficient Robot Controllers. 4th IEEE International Conference on Evolutionary Computation, 495-499. Piscataway, NJ: IEEE Press.

[8] Nordin, P., and Banzhaf, W. 1996. 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.

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

[10]Eggenberger, P. 1996. Cell Interactions as a Control Tool of Developmental Processes for Evolutionary Robotics. 4th International Conference on the Simulation of Adaptive Behavior, 440-448. Cambridge, MA: MIT Press.

[11]Eggenberger, P. 1997. Evolving Morphologies of Simulated 3D Organisms Based on Differential Gene Expression. 4th European Conference on Artificial Life, 205-213. Cambridge, MA: MIT Press.

[12]Eggenberger, P., Gomez, G., and Pfeifer, R. 2002. Evolving the Morphology of a Neural Network for Controlling a Foveating Retina. 8th International Conference on Artificial Life, 243-251. Cambridge, MA: MIT Press.

[13]Floreano, D., and Mondada, F. 1998. Evolutionary Neurocontrollers for Autonomous Mobile Robots. *Neural Networks* 11:1461-1478.

[14]Nolfi, S., and Floreano, D. 1999. Learning and Evolution. *Autonomous Robots* 7(1):89-113.

[15]Floreano, D., Nolfi, S., and Mondada, F. 2001. Co-evolution and Ontogenetic Change in Competing Robots. *Advances in the Evolutionary Synthesis of Intelligent Agents*, 273-306. Cambridge, MA: MIT Press.

[16]Nolfi, S. 2002. Power and Limits of Reactive Agents. *Neurocomputing* 42:119-145.

[17] Floreano, D., Schoeni, N., Caprari, G., and Blynel, J. 2002. Evolutionary Bits 'N' Spikes. 8th International Conference on Artificial Life, 333-344. Cambridge, MA: MIT Press.

[18]Floreano, D., and Mattiussi, C. 2001. Evolution of Spiking Neural Controllers for Autonomous Vision-based Robots. 8th International Symposium on Evolutionary Robotics, 38-61. Berlin: Springer-Verlag.

[19]Watson, R., Ficici, S., and Pollack, J. 2002. Embodied Evolution: Distributing an Evolutionary Algorithm in a Population of Robots. *Robotics and Autonomous Systems* 39:1-18.

[20]Husbands, P., Philippides, A., Smith, T., and O'Shea, M. 2001. Volume Signaling in Real and Robot Nervous Systems. *Theory in Biosciences* 120(3-4):253-269.

[21]Philippides, A., Husbands, P., Smith, T., and O'Shea, M. 2002. Fast and Loose: Biologically Inspired Couplings. 8th International Conference on Artificial Life, 293-301. Cambridge, MA: MIT Press.

[22]Beer, R., and Gallagher, J. 1992. Evolving Dynamical Neural Networks for Adaptive Behavior. *Adaptive Behavior* 1(1):91-122.

[23]Gallagher, J., Beer, R., Espenschied, K., and Quinn, R. 1996. Application of Evolved Locomotion Controllers to a Hexapod Robot. *Robotics and Autonomous Systems* 19:95-103.

[24]Kodjabachian, J., and Meyer, J. 1998. Evolution and Development of Modular Control Architectures for 1-D Locomotion in Six-legged Animats. *Connection Science*

10:211-237.

[25]Reeve, R. 1999. Generating Walking Behaviors in Legged Robots. PhD diss., University of Edinburgh, Scotland.

[26]Fujii, A., Ishiguro, A., Aoki, T., and Eggenberger, P. 2001. Evolving Bipedal Locomotion with a Dynamically-rearranging Neural Network. 6th European Conference on Artificial Life, 507-518. Berlin: Springer-Verlag.

[27]Otsu, K., Ishiguro, A., Fujii, A., Aoki, T., and Eggenberger, P. 2001. Evolving an Adaptive Controller for a Quadruped-robot with Dynamically-rearranging Neural Networks. 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2036-2044. Piscataway, NJ: IEEE Press.

[28]Paul, C., and Bongard, J. 2001. The Road Less Travelled: Morphology in the Optimization of Biped Robot Locomotion. 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, 226-232. Piscataway, NJ: IEEE Press.

[29]Bongard, J., and Paul, C. 2001. Making Evolution an Offer it Can't Refuse: Morphology and the Extradimensional Bypass. 6th European Conference on Artificial Life, 401-412. Berlin: Springer-Verlag, Berlin.

[30]Reil, T., and Husbands, P. 2002. Evolution of Central Pattern Generators for Bipedal Walking in a Real-time Physics Environment. *IEEE Transactions on Evolutionary Computation* 6(2):159-168.

[31]Reil, T., and Massey, C. 2001. Biologically Inspired Control of Physically Simulated Bipeds. *Theory in Biosciences* 120:(3-4), 327-339.

[32]Gruau, F. 1997. Cellular Encoding for Interactive Evolutionary Robotics. 4th European Conference on Artificial Life, 368-377. Cambridge, MA: MIT Press.

[33]Gomi, T., and Ide, K. 1998. Evolution of Gaits of a Legged Robot. 1998 IEEE World Congress on Computational Intelligence, 159-164. Piscataway, NJ: IEEE Press.

[34]Hornby, G., Fujita, M., Takamura, S., Yamamoto, T., and Hanagata, O. 1999. Autonomous Evolution of Gaits with the Sony Quadruped Robot. 1999 Genetic and Evolutionary Computation Conference, 1297-1304. San Francisco: Morgan Kaufmann.

[35]Hornby, G., Takamura, S., Yokono, J., Hanagata, O., Yamamoto, T., and Fujita, M. 2000. Evolving Robust Gaits with AIBO. International Conference on Robotics and Automation 2000, 3040-3045. Piscataway, NJ: IEEE Press.

[36]Jakobi, N. 1998. Running Across the Reality Gap: Octopod Locomotion Evolved in a Minimal Simulation. 1st European Workshop on Evolutionary Robotics, 39-58. Berlin: Springer-Verlag.

[37]Harvey, I., Husbands, P., and Cliff, D. 1994. Seeing the Light: Artificial Evolution, Real Vision. 3rd International Conference on the Simulation of Adaptive Behavior, 392-401. Cambridge, MA: MIT Press.

[38]Lichtensteiger, L., and Eggenberger, P. 1999. Evolving the Morphology of a Compound Eye on a Robot. 3rd European Workshop on Advanced Mobile Robots, 127-134. Piscataway, NJ: IEEE Press.

[39]Dittrich, P., Skusa, A., Banzhaf, W., and Kantschik, W. 1999. Dynamical Properties of the Fitness Landscape of a GP Controlled Random Morphology Robot. 1999 Genetic and Evolutionary Computation Conference, 1002-1008. San Francisco: Morgan Kaufmann.

[40]Lipson, H., and Pollack, J. 2000. Automatic Design and Manufacture of Robotic Lifeforms. *Nature* 406: 974-978.

[41]Thompson, A. 1997. Artificial Evolution in the Physical World. 5th International Symposium on Evolutionary Robotics, 101-125. Ontario: AAI Books.

[42]Keymeulen, D., Iwata, M., Konaka, K., Suzuki, R., Kuniyoshi, Y., and Higuchi, T. 1998. On-line Model-free and On-line Model-based Evolution for Tracking Navigation Using Evolvable Hardware. 1st European Workshop on Evolutionary Robotics, 211-226. Berlin: Springer-Verlag.

[43]Lund, H., Hallam, J., and Lee, W. 1997. Evolving Robot Morphology. 4th IEEE International Conference on Evolutionary Computation, 197-202. Piscataway, NJ: IEEE Press.

[44]Bongard, J., and Pfeifer, R. 2002. A Method for Isolating Morphological Effects on Evolved Behavior. 7th International Conference on the Simulation of Adaptive Behavior, 305-311. Cambridge, MA: MIT Press.

[45]Vapnik, V., and Chervonenkis, A. 1971. On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Theoretical Probability and Its Applications* 16(2): 264-280.

[46]Haykin, S. 1999. *Neural Networks: A Comprehensive Foundation* (2nd edition). Upper Saddle River, NJ: Prentice-Hall.