Three Generations of Automatically Designed Robots

Jordan B. Pollack, Hod Lipson, Gregory Hornby, and Pablo Funes

27th August 2002

DEMO Laboratory
Computer Science Dept.,
Brandeis University,
Waltham, MA 02454, USA,
www.demo.cs.brandeis.edu
pollack@cs.brandeis.edu

Abstract

The difficulties associated with designing, building and controlling robots have led their development to a stasis: applications are limited mostly to repetitive tasks with predefined moves. Over the last few years we have been trying to address this challenge through an alternative approach: Rather than trying to control an existing macine, or create a general-purpose robot, we propose that both the morphology and the controller should evolve at the same time. This process can lead to the automatic design and fabrication of special purpose mechanisms and controllers that achieve specific short-term objectives. Here we provide a brief review of three generations of our recent research, underlying the robots shown on the cover of this issue: Automatically designed static structures, automatically designed and manufactured dynamic electromechanical systems, and modular robots automatically designed through a generative DNA-like encoding.

1 Introduction

The high costs associated with designing, building and controlling robots have led their development to a stasis [28]. Robots in industry are only applied to simple and highly repetitive manufacturing tasks. Even though sophisticated teleoperated machines with sensors and actuators have found important applications (exploration of inaccessible environments for example), they leave very little decision, if at all, to the on-board software [29].

The central issue addressed by our work is a way to get a higher level of complex physicality under control at lower cost. We seek more controlled and moving mechanical parts, more sensors, more nonlinear interacting degrees of freedom - without entailing both the huge fixed costs of human design and programming and the variable costs in manufacture and operation. We suggest that this can be achieved only when robot design and construction are fully automatic. Considering a robot as an artificial life form, we achieve automatic design by evolving creatures — body and brain — through interaction with (simulated) reality, and transfer the designs into (real) reality.

Traditionally, robots are designed on a hardware first, software last basis: Mechanical and electrical engineers design complex articulated bodies with state-of-the-art sensors, actuators and multiple degrees of freedom. The next task should be simply to “write the software”. But humans have drastically underestimated animal brains: looking into nature we see animal brains of very high complexity, controlling bodies which have been selected by evolution precisely because they were controllable by those brains. We believe that the costs of writing intelligent controllers for arbitrary mechanical devices are so high that direct engineering of robots will continue to be an economic failure.

In nature, the body and brain of a creature are tightly coupled, the fruit of a long series of small mutual adaptations — like chicken and egg, neither one was designed first. There is never a situation in which the hardware has no software, or where a growth or mutation — beyond the adaptive ability of the brain — survives. Autonomous robots, like living creatures, require a highly sophisticated correspondence between brain, body and environment.

Therefore, we have been working to co-evolve both the brain and the body, simultaneously and continuously, from a simple controllable mechanism to one of sufficient complexity for a particular specialized task. Although we were not first to propose brain/body coevolution [6, 24, 27], we have been able to put the idea in practice.

Over the next decade, we see three technologies that are maturing past threshold to make possible a new industry of inexpensive automatically designed machines. One is the increasing fidelity of advanced mechanical design simulation, stimulated by profits from the CAD software industry [34]. The second is rapid, one-off prototyping and manufacture, which is proceeding from 3D plastic layering to stronger composite and metal (sintering) technology [7]. The third is most central to Artificial Life, our understanding of the dynamics of coevolutionary learning in the self-organization of complex systems [31, 19, 1, 8].

2 Coevolution

Coevolution, when successful, dynamically creates a series of learning environments each slightly more complex than the last, and a series of learners which are tuned to adapt in those environments. Sims' work [33] on body-brain coevolution and the more recent Framsticks simulator [23] demonstrated that the neural controllers and simulated bodies could be coevolved. The goal of our research in coevolutionary robotics is to replicate and extend results from virtual simulations like these to the reality of computer designed and constructed special-purpose machines that can adapt to real environments.

We are working on coevolutionary algorithms to develop control programs operating realistic physical device simulators, both commercial-off-the-shelf and our own custom simulators, where we finish the evolution inside real embodied robots. We are ultimately interested in mechanical structures that have complex physicality of more degrees of freedom than anything that has ever been controlled by human designed algorithms, with lower engineering costs than currently possible because of minimal human design involvement in the product.

It is not feasible that controllers for complete structures could be evolved (in simulation or otherwise) without first evolving controllers for simpler constructions. Compared to the traditional form of evolutionary robotics [9, 5, 26, 13, 22] which serially downloads controllers into a given piece of hardware, it is relatively easy to explore the space of body constructions in simulation. Realistic simulation is also crucial for providing a rich and nonlinear universe. However, while simulation creates the ability to explore the space of constructions far faster than real-world building and evaluation could, there remains the problem of transfer to real constructions and scaling to the high complexities used for real-world designs.

3 Results

We describe three generations of work in our lab towards fully automated design and manufacture of high-parts-count autonomous robots. The fundamental method is evolution inside simulation, but in simulations more and more realistic so the resulting blueprints are not simply visually believable, as in Sims' work, but also buildable, either manually or automatically. Our first results involved automatically creating high part-count structures that could be transferred from simulation to the real world. In the second generation we evolved automatically buildable dynamic machines that are nearly autonomous in both their design and manufacture, using Rapid Prototyping technology. The third generation begins to address scaling, by handling high part-count structures through modularity. Even with these three demonstrations, we feel the work is at a very early stage, with major issues only beginning to be addressed such as the integration of sensors and automating the feedback from “live” interactions.

3.1 Generation 1: Legobots

The first step towards our vision of fully evolved creatures was to demonstrate that evolving morphology for the real world was indeed possible.

In order to evolve both the morphology and behavior of autonomous mechanical devices that can be built, one must have a simulator that operates under many constraints, and a resultant controller that is adaptive enough to cover the gap between the simulated and real world.

Features of a simulator for evolving morphology are:

- Representation —- should cover a universal space of mechanisms.
- Conservative — because simulation is never perfect, it should preserve a margin of safety.
- Efficient — it should be quicker to test in simulation than through physical production and test.
- Buildable — results should be convertible from a simulation to a real object.

One approach is to custom-build a simulator for modular robotic components, and then evolve either centralized or distributed controllers for them.

In advance of a modular simulator with dynamics, we built a simulator for (static) Lego bricks, and used simple evolutionary algorithms to create complex Lego structures, which were then manually constructed [10, 11, 12].

Our model considers the union between two bricks as a rigid joint between the centers of mass of each one, located at the center of the actual area of contact

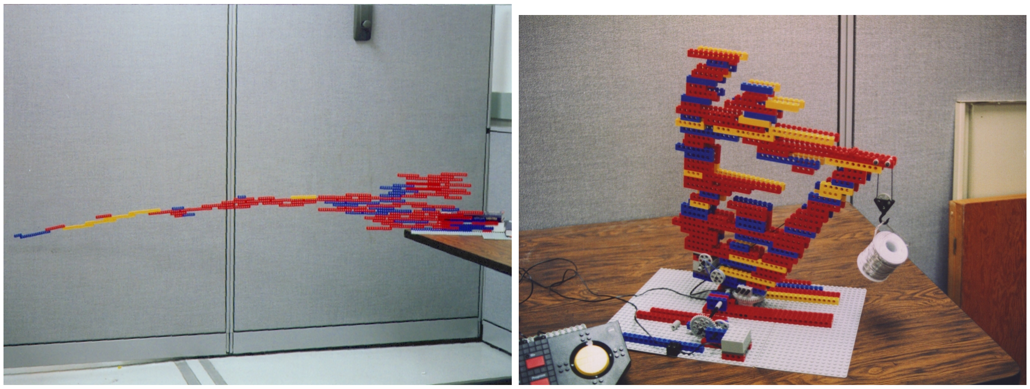


Figure 1: Photographs of the FAD Lego Bridge (Cantilever) and Crane (Triangle). Photographs copyright Pablo Funes & Jordan Pollack, used by permission.

between them. This joint has a measurable torque capacity: more than a certain amount of force applied at a certain distance from the joint will break the two bricks apart. The fundamental assumption of our model is the idealization of the union of two Lego bricks as a rotational joint with limited capacity.

The evolutionary algorithm reliably builds structures that meet fitness goals, exploiting physical properties implicit in the simulation. Building the results of the evolutionary simulation (by hand) demonstrated the power and possibility of fully automated design: the long bridge of figure 1 shows that our simple system discovered the cantilever, while the weight-carrying crane shows it discovered the basic triangular support.

3.2 Generation 2: Genetically Organized Lifelike Electromechanics (GOLEM)

The Lego machines, with computer generated blueprints, and manual construction, demonstrated that the interaction between simulated physics and evolution leads to a primitive form of discovery which can be transferred into reality. The next goal is to add some motion to these machines, and address the issue of manufacture. While Lego kits have motion components, the design space is very broad and difficult to model, and no robot can match the manual dexterity of a 10 year old human in assembly.

We started with a whole new process in which robot morphology was constrained to be buildable by a commercial off the shelf rapid prototyping machine. We evolve the bodies and controllers in simulation and were essentially able to replicate them automatically into reality [25].

These robots are comprised of only linear actuators and sigmoidal control neurons embodied in an arbitrary thermoplastic body. The entire configuration is evolved for a particular task and selected individuals are printed pre-assembled (except motors) using 3D solid printing (rapid prototyping) technology, later to be recycled into different forms. In doing so, we establish for the first time a complete physical evolution cycle. In this project, the evolutionary design approach assumes two main principles: (a) to minimize inductive bias, we must strive to use the lowest level building blocks possible, and (b) we coevolve the body and the control, so that that they stimulate and constrain each other. We use arbitrary networks of linear actuators and bars for the morphology, and arbitrary networks of sigmoidal neurons for the control. Evolution is simulated starting with a soup of disconnected elements and continues over hundreds of generations of hundreds of machines, until creatures that are sufficiently proficient at the given task emerge. The simulator used in this research is based on quasi-static motion. The basic principle is that motion is broken down into a series of statically-stable frames solved independently. While quasi-static motion cannot describe high-momentum behavior such as jumping, it can accurately and rapidly simulate low-momentum motion. This kind of motion is sufficiently rich for the purpose of the experiment and, moreover, it is simple to induce in reality since all real-time control issues are eliminated. Several evolution runs were carried out for the task of locomotion. Fitness was awarded to machines according to the absolute average distance traveled over a specified period of neural activation. The evolved robots exhibited various methods of locomotion, including crawling, ratcheting and some forms of pedalism (Figure 2). Selected robots are then replicated into reality: their bodies are first fleshed to accommodate motors and joints, and then copied into material using rapid prototyping technology. Temperature-controlled print head extrudes thermoplastic material layer by layer, so that the arbitrarily evolved morphology emerges pre-assembled as a solid three- dimensional structure without tooling or human intervention. Motors are then snapped in (manually), and the evolved neural network is activated (Figure 3). The robots then perform in reality as they did in simulation.

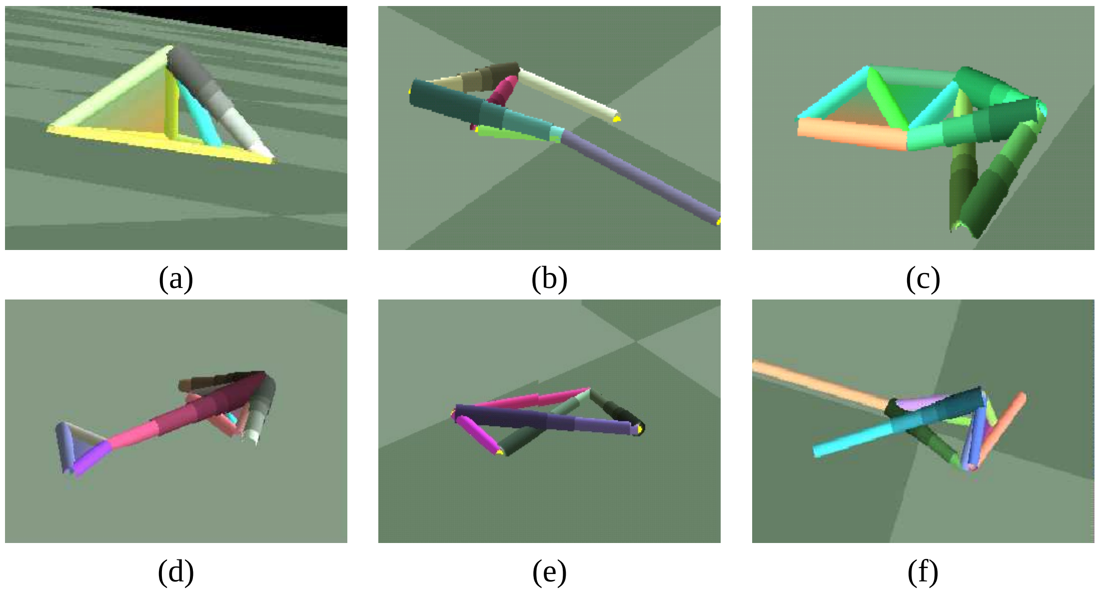
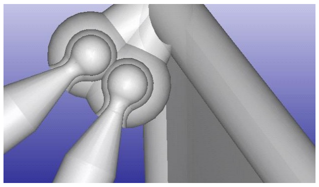
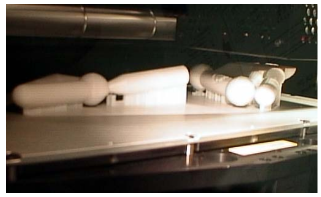
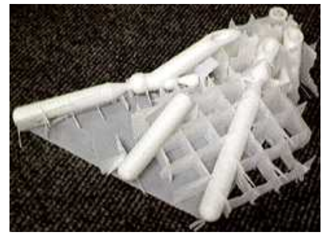
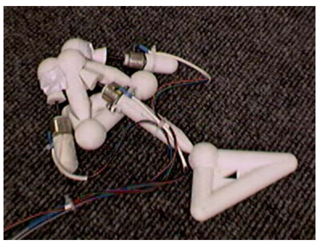

(a) (b) (c) (d) (e) (f)

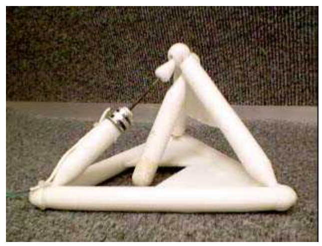
Figure 2: (a) A tetrahedral mechanism that produces hinge-like motion and advances by pushing the central bar against the floor. (b) Bipedalism: the left and right limbs are advanced in alternating thrusts. (c) Moves its two articulated components to produce crab-like sideways motion. (d) While the upper two limbs push, the central body is retracted, and vice versa. (e) This simple mechanism uses the top bar to delicately shift balance from side to side, shifting the friction point to either side as it creates oscillatory motion and advances. (f) This mechanism has an elevated body, from which it pushes an actuator down directly onto the floor to create ratcheting motion. It has a few redundant bars dragged on the floor.


(a)


(b)


(c)


(d)


(e)

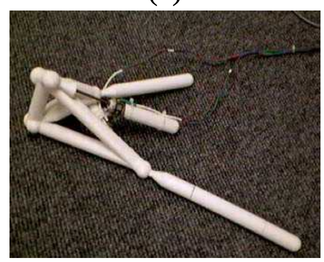

(f)

Figure 3: (a) Fleshed joints, (b) replication progress, (c) pre-assembled robot (figure 2f), (d,e,f) final robots with assembled motors

3.3 Generation 3: Modularity Generative Design (Tinkerbots)

While the GOLEM project demonstrated validity of our approach to automatic design and manufacture, the machines which were produced are obviously fairly simple compared to the kinds of robots buildable by teams of human engineers. In fact most work in automatic design of engineering products using techniques inspired by biological evolution, [21, 3, 17, 10, 4, 25] suffers the same criticism.

Our third generation starts to address the issue of whether evolutionary automatic design techniques can attain the higher level of complexity necessary for practical engineering projects. Since the search space grows exponentially with the size of the problem, search algorithms that use a direct encoding for designs may not scale to large designs. An alternative to a direct encoding is a generative specification, which is a grammatical encoding that specifies how to construct a design, [32] and [2]. Similar to a computer program, a generative specification can allow the definition of re-usable sub-procedures allowing the design system to scale to more complex designs than can be achieved with a direct encoding.

Ideally an automated design system would start with a library of basic parts and would iteratively create new, more complex modules, from ones already in its library. The principle of modularity is well accepted as a general characteristic of design, as it typically promotes decoupling and reduces complexity [35]. In contrast to a design in which every component is unique, a design built with a library of standard modules is more robust and more adaptable, and enhances field repair.

Our third generation of automatically designed robots focus on modular design, and uses L-systems as the genotype evolved by the evolutionary algorithm. L-systems are a grammatical rewriting system introduced by Lindenmeyer in 1968 to model the biological development of multicellular organisms. Rules are applied in parallel to all characters in the string just as cell divisions happen in parallel in multicellular organisms. Complex objects are created by successively replacing parts of a simple object by using the set of rewriting rules. Using this system we have evolved 3D static structures [15], and locomoting mechanisms [14, 16], some of which are shown in figure 4, and and transferred successfully into reality, as seen in figure 5 [14]. The creature on the cover is our first 3D machine using these techniques.

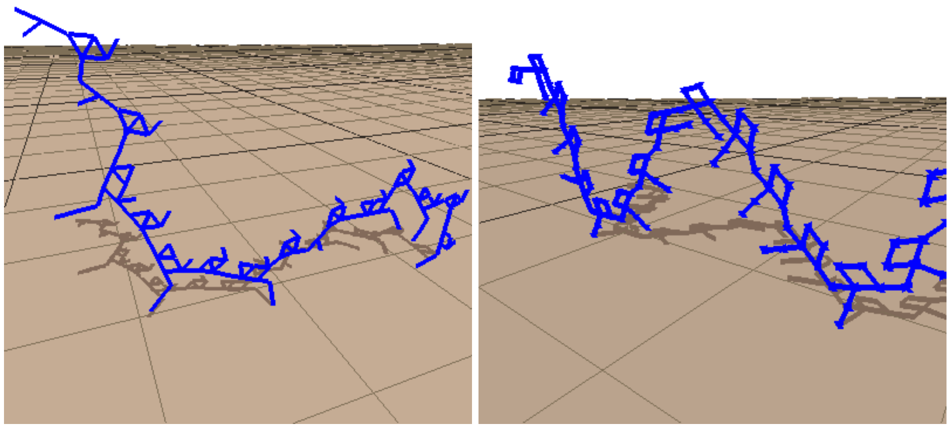


Figure 4: Examples of evolved, modular creatures.

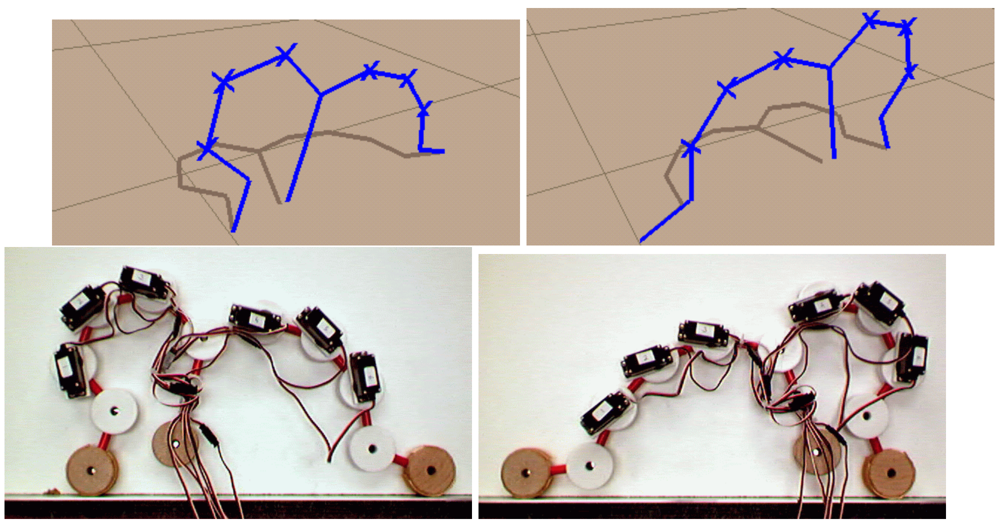


Figure 5: Two parts of the locomotion cycle of a 2D, modular locomoting creature in both simulation and reality.

4 Discussion and Conclusion

Can evolutionary and coevolutionary techniques be used in the design of real robots as "Artificial Lifeforms?" In this paper we have presented three generations of our work, each of which addresses one or more dimensions of the problem. We have overviewed research in use of simulations for handling high part-count static structures that are buildable, dynamic electromechanical systems with complex morphology that can be built automatically, and generative encodings as a means for scaling to complex structures.

The limitations of the work are clearly apparent: these machines do not yet have sensors, and are not really interacting with their environments. Feedback from how robots perform in the real world is not automatically fed-back into the simulations, but require humans to refine the simulations and constraints on design. Finally, there is the question of how complex a simulated system can be, before the errors generated by transfer to reality are overwhelming.

We cannot claim immediate solution to these problems. In other work, however, we have demonstrated how coevolution can lead to complex performance in domains like game-playing [31] and design of complex algorithms like sorting networks [18] and cellular automata [20]. In our next generations of evolved creatures we expect to see some sensor integration, and we have already demonstrated robot "cultural" evolution, learning from interacting in the real environment. [36]

The issue of whether or not this kind of artificial life work will ever be practical and scaleable is best related to the history of computer chess. The theory that machines could play a game like chess was from the 1920's, the first chess playing computer was built in the mid 1950's, and made random legal moves. While proponents of funding for the new field of AI were over-optimistic, by the end of the century, nevertheless, with almost unlimited CPU time at its disposal, Deep Blue was able to win a tournament against the leading human player - using 80 year old theory [30].

Perhaps the small demonstrations of automatic design will lead — with continued development, and increases in computer speed and simulation fidelity, coupled to increases in basic theory of coevolutionary dynamics — over time, to the point where fully automatic design is taken for granted, much as computer aided design is taken for granted in manufacturing industries today.

Our current research moves towards the overall goal via multiple interacting paths, of simulation, theory, building and testing in the real world, and applications. It is a broad, multidisciplinary long-term endeavor, where what we learn in one path aids the others. We believe such a broad endeavor is the only way to ultimately construct complex autonomous machines which can economically justify their own existence.

Acknowledgements

This research was supported over the years in part by the National Science Foundation (NSF), the office of Naval Research (ONR), and by the Defense Advanced Research Projects Agency (DARPA).

References

[1] Peter J. Angeline, Gregory M. Saunders, and Jordan B. Pollack. An evolutionary algorithm that constructs recurrent networks. *IEEE Transactions on Neural Networks*, 5(1):54–65, 1994.

[2] P. Bentley and S. Kumar. Three ways to grow designs: A comparison of embryogenies of an evolutionary design problem. In Banzhaf, Daida, Eiben, Garzon, Honavar, Jakiel, and Smith, editors, *Genetic and Evolutionary Computation Conference*, pages 35–43, 1999.

[3] P. J. Bentley. *Generic Evolutionary Design of Solid Objects using a Genetic Algorithm*. PhD thesis, Division of Computing and Control Systems, School of Engineering, The University of Huddersfield, 1996.

[4] Peter Bentley, editor. *Evolutionary Design by Computers*. Morgan-Kaufmann, San Francisco, 1999.

[5] D. Cliff, I Harvey, and P. Husbands. Evolution of visual control systems for robot. In M. Srinivisan and S. Venkatesh, editors, *From Living Eyes to Seeing Machines*. Oxford, 1996.

[6] D. Cliff and J. Noble. Knowledge-based vision and simple visual machines. *Philosophical Transactions of the Royal Society of London: Series B*, 352:1165–1175, 1997.

[7] D. Dimos, S.C. Danforth, and M.J. Cima. Solid freeform and additive fabrication. In J. A. Floro, editor, *Growth instabilities and decomposition during heteroepitaxy*. Elsevier, 1999.

[8] S. Ficici and J. B. Pollack. Challenges in coevolutionary learning: Arms-race dynamics, open-endedness, and mediocre stable states. In Adami, Belew, Kitano, and Taylor, editors, *Proceedings of the Sixth International Conference on Artificial Life,* 1998.

[9] D. Floreano and F. Mondada. Evolution of homing navigation in a real mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics*, 1996.

[10] Pablo Funes and Jordan B. Pollack. Computer evolution of buildable objects. In Phil Husbands and Inman Harvey, editors, *Fourth European Conference on Artificial Life,* pages 358–367, Cambridge, 1997. MIT Press.

[11] Pablo Funes and Jordan B. Pollack. Evolutionary body building: Adaptive physical designs for robots. *Artificial Life*, 4(4):337–357, 1998.

[12] Pablo Funes and Jordan B. Pollack. Computer evolution of buildable objects. In Peter Bentley, editor, *Evolutionary Design by Computers*, pages 387 – 403. Morgan-Kaufmann, San Francisco, 1999.

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

[14] G. S. Hornby, H. Lipson, and J. B. Pollack. Evolution of generative design systems for modular physical robots. In *IEEE International Conference on Robotics and Automation*, pages 4146–4151, 2001.

[15] G. S. Hornby and J. B. Pollack. The advantages of generative grammatical encodings for physical design. In *Congress on Evolutionary Computation*, pages 600–607, 2001.

[16] G. S. Hornby and J. B. Pollack. Body-brain coevolution using l-systems as a generative encoding. In *Genetic and Evolutionary Computation Conference*, pages 868–875, 2001.

[17] P. Husbands, G. Germy, M. McIlhagga, and R. Ives. Two applications of genetic algorithms to component design. In T. Fogarty, editor, *Evolutionary Computing. LNCS 1143*, pages 50–61. Springer-Verlag, 1996.

[18] H. Juillé and J. B. Pollack. Co-evolving intertwined spirals. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 351–358, 1996.

[19] H. Juillé and J. B. Pollack. Dynamics of co-evolutionary learning. In *Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 526–534. MIT Press, 1996.

[20] H. Juillé and J. B. Pollack. Coevolving the "ideal trainer": Discovery of cellular automata rules. In Koza, editor, *Proceedings Third Annual Genetic Programming Conference*, July 1998.

[21] C. Kane and M. Schoenauer. Genetic operators for two-dimentional shape optimization. In J.-M. Alliot, E. Lutton, E. Ronald, M. Schoenauer, and D. Snyers, editors, *Artificiale Evolution - EA95*. Springer-Verlag, 1995.

[22] Y. Kawauchi, M. Inaba, and T. Fukuda. Genetic evolution and self-organization of cellular robotic system. *JSME Int. J. Series C. (Dynamics, Control, Robotics, Design & Manufacturing)*, 38(3):501–509, 1999.

[23] M. Komosinski and S. Ulatowski. Framsticks: Towards a simulation of a nature-like world, creatures and evolution. In Jean-Daniel Nicoud Dario Floreano and Francesco Mondada, editors, *Proceedings of 5th European Conference on Artificial Life (ECAL99)*, volume 1674 of *Lecture Notes in Artificial Intelligence*, pages 261–265. Springer-Verlag, 1999.

[24] W. Lee, J. Hallam, and H. Lund. A hybrid gp/ga approach for co-evolving controllers and robot bodies to achieve fitness-specified tasks. In *Proceedings of IEEE 3rd International Conference on Evolutionary Computation*, pages 384–389. IEEE Press, 1996.

[25] Hod Lipson and Jordan B. Pollack. Automatic design and manufacture of robotic lifeforms. *Nature*, 406(6799):974–978, 2000.

[26] H. Lund. Evolving robot control systems. In Alexander, editor, *Proceedings of 1NWGA*. University of Vaasa, 1995.

[27] H. Lund, J Hallam, and W. Lee. Evolving robot morphology. In *Proceedings of IEEE Fourth International Conference on Evolutionary Computation*, pages 197–202. IEEE Press, 1997.

[28] Hans P. Moravec. Rise of the robots. *Scientific American*, pages 124–135, December 1999.

[29] J. Morrison and T. Nguyen. On-board software for the mars pathfinder microrover. In *Proceedings of the Second IAA International Conference on Low-Cost Planetary Missions*. John Hopkins University Applied Physics Laboratory, April 1996.

[30] M. Newborn. *Kasparov vs. Deep Blue: Computer Chess Comes of Age*. Springer Verlag, New York, 1996.

[31] J. B. Pollack and A. D. Blair. Coevolution in the successful learning of backgammon strategy. *Machine Learning*, 32:225–240, 1998.

[32] M. Schoenauer. Shape representations and evolution schemes. In L. J. Fogel, P. J. Angeline, and T. Back, editors, *Proceedings of the 5th Annual Conference on Evolutionary Programming*. MIT Press, 1996.

[33] K Sims. Evolving 3d morphology and behavior by competition. In R. Brooks and P. Maes, editors, *Proceedings 4th Artificial Life Conference*, pages 28–39. MIT Press, 1994.

[34] Mark Sincell. Physics meets the hideous bog beast. *Science*, 286(5439):398–399, october 1999.

[35] Nam P. Suh. *The Principles of Design*. Oxford University Press, 1990.

[36] R. Watson, S. Ficici, and J. B. Pollack. Embodied evolution: Embodying an evolutionary algorithm in a population of robots. In P. Angeline, Z. Michalewicz, M. Schoenauer, X. Yao, and A. Zalzala, editors, *1999 Congress on Evolutionary Computation*, 1999.