Jason D. Lohn
NASA Ames Research Center, USA

Gregory S. Hornby
University of California at Santa Cruz, USA

Evolvable Hardware

Using Evolutionary Computation to Design and Optimize Hardware Systems

Abstract: Evolvable hardware lies at the intersection of evolutionary computation and physical design. Through the use of evolutionary computation methods, the field seeks to develop a variety of technologies that enable automatic design, adaptation, and reconfiguration of electrical and mechanical hardware systems in ways that outperform conventional techniques. This article surveys evolvable hardware with emphasis on some of the latest developments, many of which deliver performance exceeding traditional methods. As such, the field of evolvable hardware is just now starting to emerge from the research laboratory and into mainstream hardware applications.

The central idea behind evolvable hardware is to gain the ability to automatically design and optimize electrical and mechanical structures by harnessing the power of an evolutionary algorithm. For example, one could apply a genetic algorithm to automatically design an airplane wing to maximize lift and minimize drag. The range of applications is wide and encompasses of multitude of application domains: jet engines, trusses, chip design and fabrication, antenna design, controller algorithms, optical systems, robotics, and a wide array of engineering optimization problems to improve metrics such as cost, power, size, thermal properties, and manufacturability.

At one level, the evolutionary algorithm is simply looking for combinations of input parameters to accomplish a hardware optimization problem of some sort. At a deeper level, the algorithm is searching and exploiting design spaces induced by the physics of the materials used to build the hardware. In this sense, the EA is exploring the dark corners of what is combinatorially possible given the imposed natural (physics) and artificial (human-specified) design constraints. These are subspaces that humans have left unexplored, and they can indeed be small corners or in some cases large expanses of virgin territory. A carefully constructed evolutionary algorithm will have little bias to limit it to only the known areas of the design space. Although cliched, "out-of-the-box thinking" captures the essence of what the algorithm is doing. As a result, many

1556-603X/06/$20.00©2006IEEE

EH practitioners have experienced surprise and amusement when they view their latest evolved result and came to realize what clever shortcut or dark corner their EA has explored.

Although the term "evolvable hardware" (EH) did not come into use until the early 1990s, and for many years thereafter referred solely to electronics applications, initial work on evolvable hardware applications began concurrently with the first work in evolutionary computation (EC) itself. One of the earliest works was the application of an evolutionary algorithm to the "1400-Terminal Black Box" for the Atlas Missile Guidance System [2]. In this work, published in 1963, an evolutionary algorithm was used to optimize the wiring configuration for subsystem on a ballistic missile. Other early applications include optimization of natural gas flow in a large network of pipeline [3] and General Electric's applications in the design of gas turbine engines.

Although there were relatively few EH applications from the 1960s through the 1980s, EC algorithms continued to improve. Beginning in the late 1980s, the EC field saw tremendous growth sparked by the widespread availability of computers, especially personal computers. Ever-increasing processing power following Moore's Law enabled larger, more complex EC applications to be approached, especially hardware applications.

A key development occurred in 1992 when Degaris had the insight that the Field Programmable Gate Array (FPGA) could be repeatedly re-programmed under the control of a genetic algorithm [1]. FPGA's are all-purpose chips that contain a matrix of cells that can be customized under software control to implement combinational and sequential logic circuits. They are programmed using a configuration bitstring and Degaris realized that a genetic algorithm could be used to craft them and hence automatically program the FPGA. Degaris identified two ways in which the hardware could be evolved: when the EA is wrapped around a software model of the hardware system and when the EA is directly changing the hardware itself, modes he termed extrinsic and intrinsic, respectfully. Intrinsic hardware evolution is also referred to as online or hardware-in-the-loop evolution, and the extrinsic mode is synonymous with offline evolution.

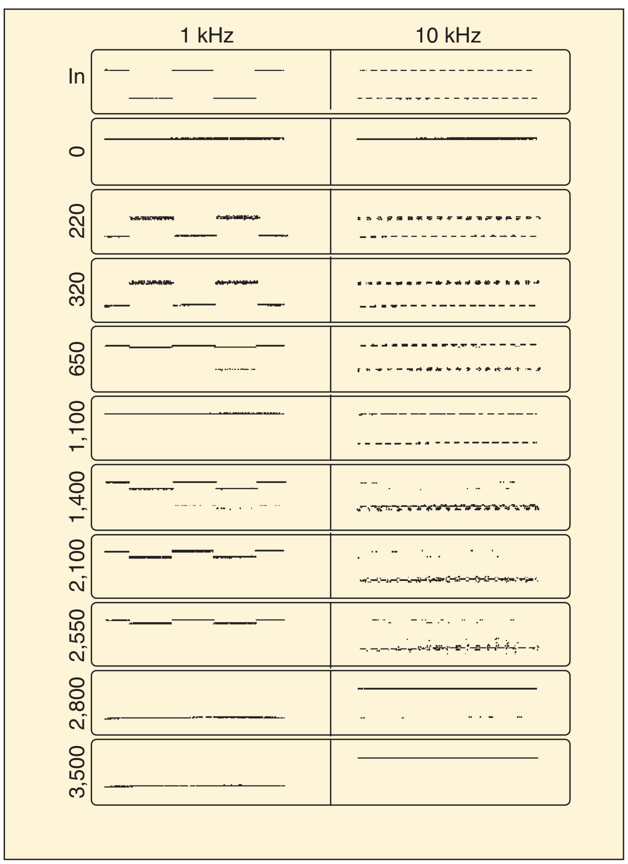


FIGURE 1 Thompson's evolved tone discriminator results: at top are the 1kHz and 10kHz input waveforms; below are the corresponding outputs of the best of generation individuals.

Degaris' work touched off a great deal of interest in the burgeoning EH field, so much so that by the mid-1990s a group of researchers organized the inaugural international meeting called ICES: the International Conference on Evolvable Systems : from Biology to Hardware. This successful meeting launched a bi-annual international conference that continues to bring together new ideas and applications in evolvable hardware. Later, in 1999, NASA and the US Department of Defense commenced a series of annual workshops and conferences that also published new developments, with a focus on aerospace applications. At the present, the body of work in EH continues to grow and mature, with more applications making the jump from laboratory demonstration to fielded, real-world application. In the following sections, we describe a few of these applications from the domains of analog and digital circuits, robotics, and antennas.

FPGA Hardware Evolution

A notable example that illustrates the promise of evolvable hardware is the ground-breaking work of Adrian Thompson [14]. Thompson used a digital Field Programmable Gate Array (FPGA) for a tone discrimination task, an application many thought would not be possible. This seminal work was the first demonstration of online hardware evolution. Thompson hypothesized that an evolutionary algorithm could explore the space of circuit designs in ways that humans would never think of, exploiting the physics of the silicon, and potentially outperform conventional design processes. His intuitions were confirmed when he successfully used evolution to coax a digital chip into performing the discrimination function.

The tone discriminator experiment aimed to use an FPGA to discriminate between square waves of 1kHz and 10kHz. With 100 FPGA logic cells at its disposal, evolution needed to devise a circuit where the output went to +5V as soon as one of the frequencies is present, and 0V for the other. This is not an easy task because the evolved circuit had to discriminate between input periods five orders of magnitude longer than the propagation time of each cell, without the aid of a clock or off-chip signals.

Figure 1 shows some of the results. The input is shown at top and the results from successive generations proceeds downwards until generation 3500 where the solution is found. Figure 2 shows the logic cell interconnections of the evolved tone discriminator.

What Thompson discovered is that the algorithm was able to exploit the underlying physics of the substrate on which it was operating. Evolution was able to devise a way to efficiently use the logic elements, and hence the transistors that compose them, to determine which input waveform was present at the input.

This result was notable for a few reasons. First, it stands as the first hardware-in-the-loop demonstration of evolvable hardware. It was a confirmation that such types of experiments were possible. Second, it was a convincing demonstration that evolution was clever enough to commandeer a chip to perform a task that it was not designed to handle. EC researchers have long been accustomed to being surprised by the results of an EA exploring new search spaces, but finally here was a hardware embodiment of this phenomenon. Third, it opened the door to many researchers as it showed that a particular part, the Xilinx 6200 FPGA, was particularly amenable to running evolvable hardware experiments.

Evolution of Analog Circuits and Optics

In this section we highlight some of the impressive evolvable hardware results in analog circuit design and optical systems. The work of John Koza in these areas has led to evolved designs that not only duplicate previously-patented inventions but are worthy to be patented themselves. Koza's genetic programming (GP) [6] is used in each of the hardware applications below. Other recent evolvable hardware applications by Koza, including antennas, controllers, and quantum computing circuits are described in [7].

Analog circuit design has proven to be a fertile area in evolvable hardware. While EC methods were not the first artificial intelligence techniques to be applied to analog design, they are quite effective at high-level design tasks. As evidence, Koza has demonstrated many GP-evolved analog circuit designs that infringe on patented inventions. For example, an evolved low-pass filter that was discovered by genetic programming and infringes on a 1925 patent by Otto Zobel of AT&T is shown in Figure 3. The circuit is called an "M-derived half section" filter and exhibited a sharper transition in its frequency domain behavior. The evolved filter has recognizable features of the patented circuit and the differences between the two are minor.

An evolved analog circuit that was discovered by GP and duplicates the functionality of the low-voltage balun circuit that was patented by Sang Gug Lee of the Information and Communications University in 2001 is shown in Figure 4. As with the previous filter, both the topology and sizing of the circuit elements were evolved. The evolved circuit duplicates the functionality of the patented circuit and contains the key part of Lee's invention: a coupling capacitor that blocks DC (see capacitor C302 in Figure 4).

An evolved eyepiece lens system that was discovered by genetic programming and infringes on a 2000 patent is shown in Figure 5. The inventors, Koizumi and Watanabe, desired to minimize the number of lenses while achieving a specified aberration correction and maintaining less than 6% distortion. The evolved lens satisfies these and other requirements and with GP discovering a design that used a different topology than the patented design.

Evolved Robots

Robotics is a domain where EC methods have demonstrated prowess [11]. In this section we describe a system developed by Hod Lipson and Jordan Pollack that co-evolved the body and brain of a simple walking robot and produced actual robots.

With human-designed robots, the hardware is typically designed and constructed first, then it is given to the programmer to write the controller. For complicated robots with many degrees of freedom, programming the software controller is a more challenging task than developing the hardware. The central issue addressed by Lipson and Pollack's work is the ability to automatically design robots with complex morphologies and a tightly adapted control system at low cost.

Inspired by nature, Lipson and Pollack [27] developed an evolutionary system to automatically design both the robot morphology and controller by using an artificial co-evolutionary

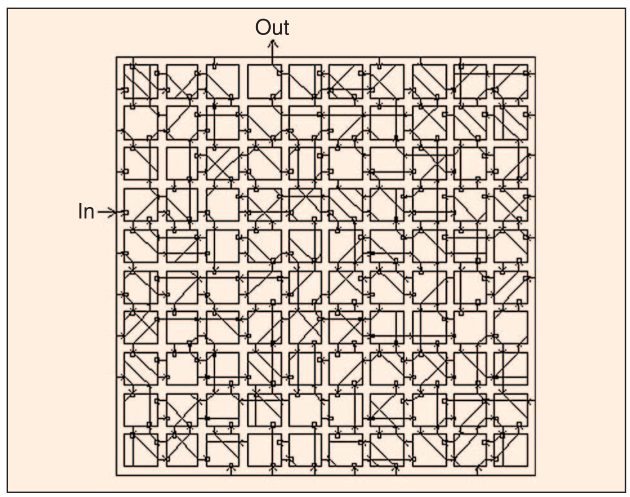


FIGURE 2 Thompson's evolved tone discriminator circuit showing the interconnections between the logic cells.

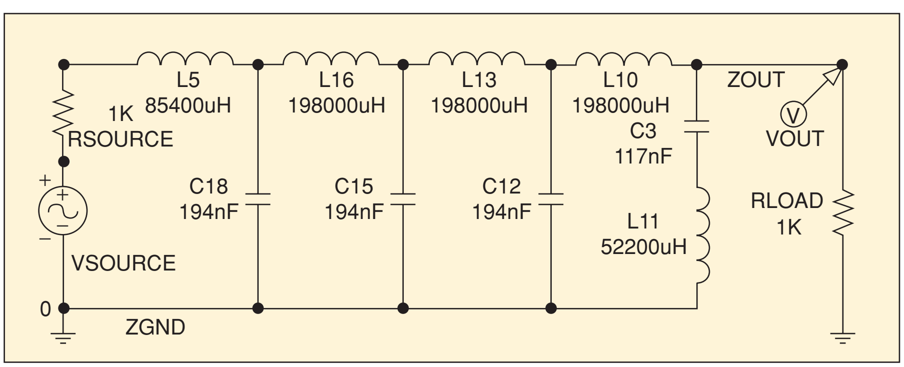


FIGURE 3 Evolved low-pass analog circuit that infringes on a 1925 patent.

process to simultaneously evolve both the body and brain of artificial life forms. The goal of their work was to evolve locomoting robots built out of fixed bars, ball-joints and linear actuators controlled by sigmoidal neurons. Also, the configuration of the bodies were constrained to be buildable out of thermoplastic using a commercial, off-the-shelf Stratasys rapid-prototyping machine. The artificial evolution process was performed in a realistic physical simulation and created simple robotic devices that inched their way across a floor. Several evolved designs were selected for manufacture and their chasis were printed on their rapid prototyping machine with the actuators and wires snapped in by hand (Figure 6). While this was not the first project to evolve body-brain coevolution [28], [29], it was the first to have gone from computer simulation to reality. They later evolved more sophisticated walking and rolling robots using a more powerful representation [30] (see Figure 7).

A more recent follow-on to the work of the automatic design and manufacture of robots is the evolution of self-reproducing machines by Zykov, Mytilinaios, Adams and Lipson [31]. Self-reproduction is necessary for the long-term sustainability and evolution of physical lifeforms. The self-replicating robots that Zykov et al. used were essentially a modular robotic system [32] in which each robotic element is a cube with two connection points and a single rotating joint about a plane through two diagonally opposing corners. Connections between modules are made using magnets.

Using these robots they evolved in simulation a configuration and a controller that would pick up new modules from a dispenser and move it in place to piece-by-piece assemble a duplicate of itself. They built two of the evolved self-replicators, consisting of a handful of modules, and demonstrated successful self-replication with the evolved configuration and controller (Figure 8).

Evolved Sony AIBO Gait

In addition to hardware design, another area of evolvable hardware is the evolutionary optimization of controllers for physical, robotic systems such as the evolution of a locomotion gait. Creating walking and running behaviors is a challenging task that must be accomplished for every legged robot and is an area that evolutionary techniques have shown much promise. Locomotion gaits for most robots are static and programmed by hand (for

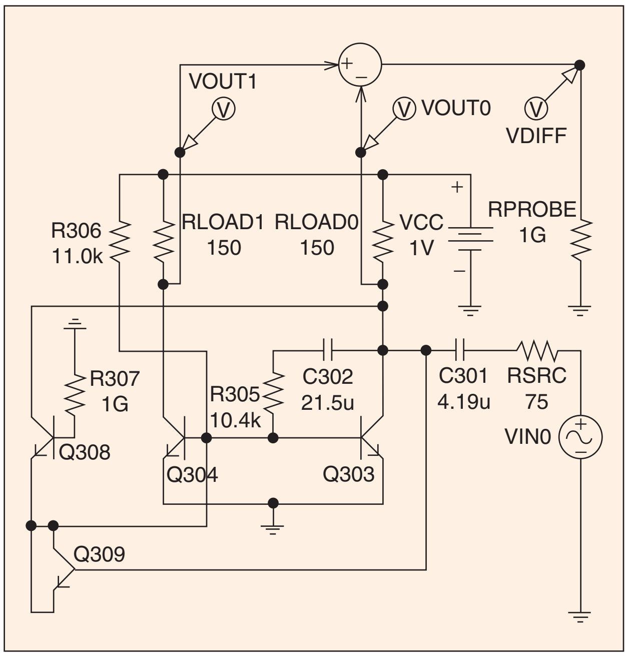


FIGURE 4 Evolved low-voltage balun circuit that was patented in 2001.

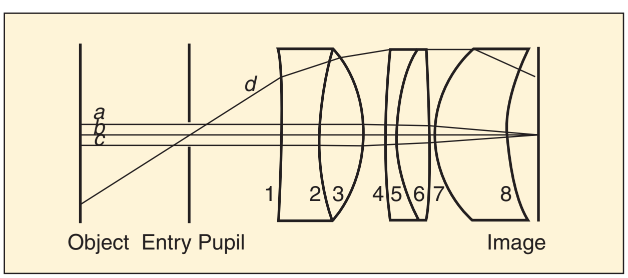


FIGURE 5 Evolved eyepiece lens system that infringes on a 2000 patent.

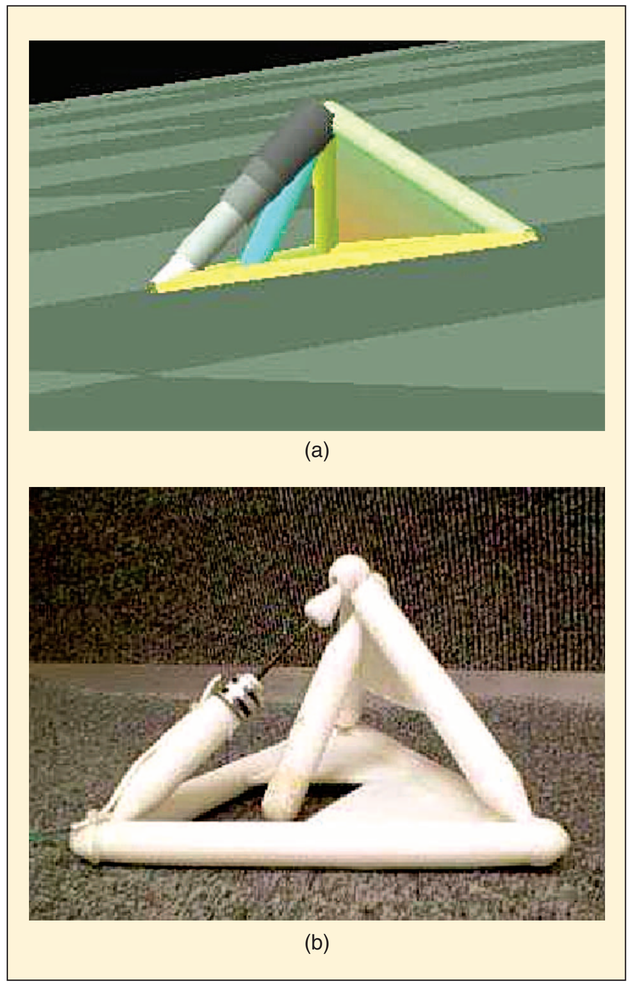


FIGURE 6 An evolved artificial life-form from the GOLEM project shown in (a) simulation and (b) reality.

surveys see [33], [34]). In initial work in evolving gaits for robots the experimenter manually evaluated the gait performance and entered this into the EA [35], [36]. As computers became faster and better quality robotics simulators were developed gaits were evolved in simulation for actual robots [37]–[39]. In this section we review the fully autonomous evolution of a dynamic gait on Sony's Entertainment robot, AIBO [40], [41].[1]

The system for developing gaits used an EA to optimize a vector of parameters that specify a gait. All processing was handled by the robot's onboard processor and each set of gait parameters was evaluated using the robot's sensors. Not only was this the first system to demonstrate the fully autonomous learning of a dynamic gait on a quadruped robot, but one of the evolved gaits was used in the first consumer version of AIBO.

To autonomously acquire gaits an evolutionary algorithm optimizes gait parameters by sending sets of parameters to the locomotion module and then evaluating the resulting performance using the robot's onboard sensors. AIBO has nineteen degrees of freedom from: three in each of the four legs, three in the neck, a two-degree of freedom tail and two actuated ears, each with a single degree of freedom. For sensors, AIBO has a micro-camera, stereo microphone, position sensitive device, and touch sensors located on the top of the head and on the bottom of each leg. The robot's body houses the CPU and battery as well as a gyroscope and accelerometers. See [42] for a more detailed description of the robots' hardware and software architectures.

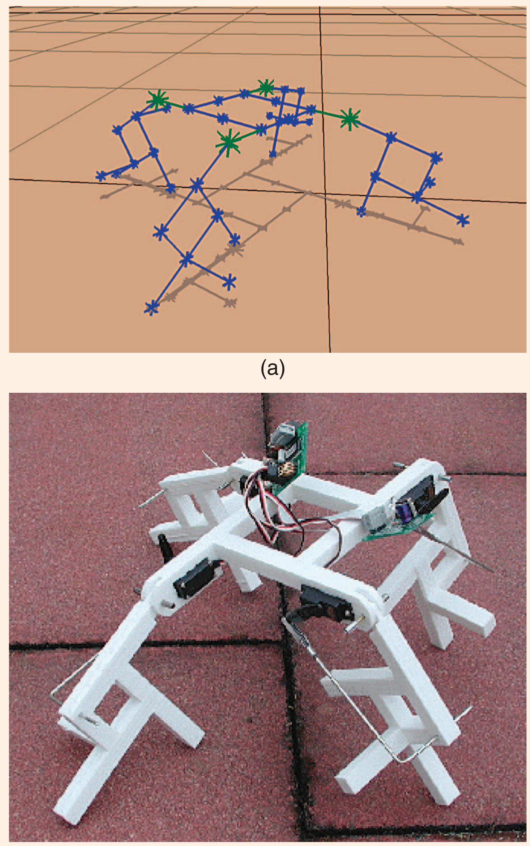
The gait parameters that control a gait specify such things as the position and orientation of the body, the swing path and rate of swinging of the legs, the amplitude of oscillation of the body's location and orientation, how the gain varies during the course of a swing cycle for each leg, and the relative phases of the legs.

Evolution takes place inside a walled area (Figure 9) with a strip of colored cloth to mark the center of each end. Cables attached to the robot supply power and allow the robot to communicate data back to a host computer. Evaluating a set of gait parameters consists of locomoting with them and then measuring the straightness and distance traveled. The procedure by which a robot evaluates its own performance consists of three parts and was influenced by our experiences from manually evaluating sets of gait parameters.

The first part of an evaluation trial consists of the robot centering on the color strip. Objects are detected from the image returned by the onboard Micro-Camera-Unit (MCU) by using an LSI chip with eight color detection tables (CDTs). These CDTs detect colors within a user-specified range for each pixel location in the image. Using a CDT for each color strip, the robot centers by turning its body in a fixed direction in search of the desired color. Once detected the robot continues turning, reversing directions if necessary, until the average horizontal location of the color strip falls within $\pm 6^\circ$ of the center of the image for a period of two seconds.

In the second part of an evaluation trial the robot determines how far it is from the color strip and then runs toward it. The distance to the color strip is measured using the robot's position sensitive device (PSD) sensor, which is located on the front part of its head.

The third part of an evaluation trial begins after the robot has stopped moving and consists of the robot using its sensors to determine the straightness of its movement and the distance it traveled.



(a)

(b)

FIGURE 7 Robots evolved with GENRE shown both (a) in simulation and (b) reality.

FIGURE 8 Zykov, Mytilinaios, Adams, and Lipson's self-replicating robot and a copy (photo courtesy of Cornell University).

[1] AIBO is a registered trademark of Sony Corporation.

Otherwise, if the robot did not fall, the trial ends successfully and the robot pans its head until it finds the color strip. Before optimization begins, an initial population of randomly generated gait parameters must be created. The particular EA that we use is a steady-state evolutionary algorithm [43] with a population of thirty individuals.

A steady-state EA works by iteratively selecting individuals from the population to act as parents and then using them to create a new individual. Each evolutionary run lasted 500 evaluations, which took approximately 25 hours. Using the best individual evolved from random populations, a secondary evolutionary run was made in which the initial population was created by adding small random values to each of the parameters of the seed individual. With further evolution based on this "seed" set of gait parameters an even better gait was found with a speed of 9m/min. This gait is much faster than any of the individuals evolved in the first set of experiments with the ERS-110 and was sufficiently robust that it was approved by Sony's quality assurance department and is used on the consumer version of AIBO.

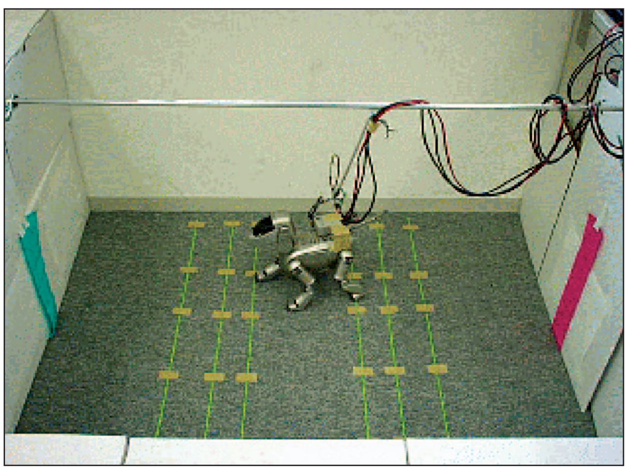


FIGURE 9 The experimental environment for evolving gaits.

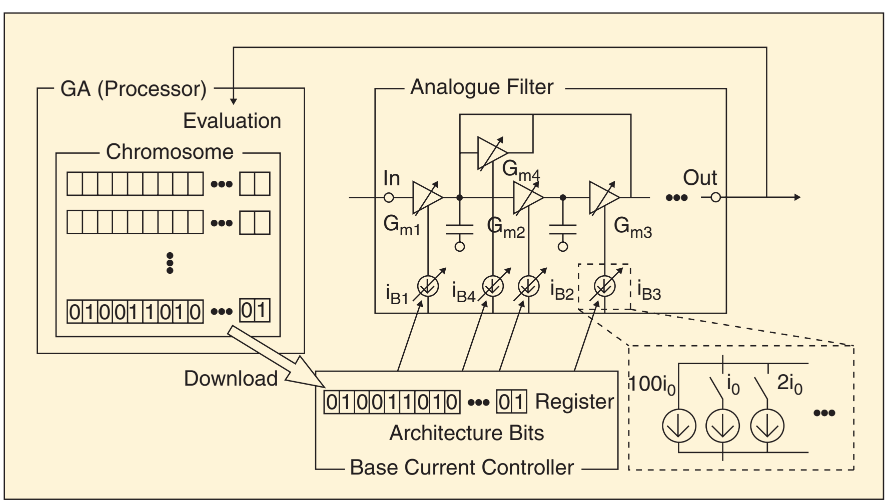


FIGURE 10 Block diagram of the intermediate filters showing the evolved values for the transconductance amplifiers.

Industrial Applications of Evolvable Hardware

Tetsuya Higuchi is one of the leading applications researchers in evolvable hardware. He has focused on industrial applications and in this section we highlight three of his EH applications: an amplifier circuit used in mobile phones, a data compression chip for printers, and a semiconductor yield-enhancement technique. Other industrial EH applications from his group include optical systems, robotics, lasers, and adaptive controllers [13], [12].

Fabrication variance in the manufacture of analog chips can cause problems in some high-performance analog applications. An intermediate filter (IF), commonly found in mobile telephones, is one such circuit: a 1% discrepancy from the center frequency is unacceptable. Chips that do not meet this stringent requirement must be discarded lowering the yield and hence profitability of the chip. Higuchi devised a way to correct the variations in the IF circuits by having a genetic algorithm control 39 transconductance amplifiers which properly tune currents to compensate for the errors introduced by fabrication variances (see Figure 10).

Date compression hardware is required for industrial printers that are required to rapidly process large amounts of data. For example, in order for an electrophotographic printer (EP) to print a book with 100 pages, approximately 7 gigabytes must be transferred to the printer at a rate of 1.8 gigabytes per minute. Conventional data compression/uncompression techniques were too slow so Higuchi devised a prediction technique implemented in reconfigurable hardware. The technique predicts the value of a given pixel by examining its neighboring pixels: those pixels that are correctly predicted by the evolved algorithm do not need to be stored, and hence the data stream is compressed. The prediction function (Figure 11) is evolved by a genetic algorithm and used to program a reconfigurable chip. In practice, the evolved data compressor was able to achieve compression ratios that were significantly better than the Lempel-Ziv and JBIG international standards.

Low yield rates of fabricated VLSI chips is a very expensive problem for the semiconductor industry. In the early stages of mass production, typical yield rates can be less than 10%. Out of specification timing delays between circuits, or "clock skew" are one of the causes for low yield. These delays can arise from variations in parasitic capacitances and resistors along the data lines between components. Higuchi devised a clock timing adjustment architecture to combat the clock skew problem by using a genetic algorithm to make the clock timings perform within the intended design specifications (see Figure 12). Simulation results showed yields rising from 2.9% to 51.1% using evolved clock-timing circuits.

Evolution of Physical Forms

One of the more obvious domains of evolvable hardware is simply for the evolutionary optimization of shapes and physical forms. A variety of projects of produced forms in simulation, such as cantilevers [15] and the cross-section of a beam [16], with a large amount of work done in architectural design [17]–[20]. There are only a few examples in which the evolved designs were manufactured in reality.

Bentley created one of the first design systems which was targeted toward producing forms for various applications [21], [22]. The basic building block of this system is cuboids, with variable width, height and depth as well as a plane of variable orientation allowing for resulting shapes to have surfaces at arbitrary angles. Evolved individuals in the resulting EA consisted of a tree-structured assembly procedure for attaching cuboids together to create a resulting shape. This system was able to evolve coffee tables, portable steps, heat sinks, optical prisms, and streamlined shapes, of which one of the coffee tables was constructed out of a dozen or so pieces of wood.

Expanding on functionality is the evolution of buildable structures by Funes [23], [24], using LEGO bricks as the basic component. With this system evolution was performed in a simulated physical environment that tested designs to determine if the LEGO structure would stay together or fall apart. Again, individuals in this evolutionary design system consisted of tree-structured assembly procedure that specified the connection of bricks, starting from a ground position. Using this system Funes evolved, and built, LEGO trusses, bridges and tables.

While past work has been successful at producing simple, albeit novel artifacts, a concern with these systems has been how well their search ability will scale to more sophisticated designs and the larger design spaces associated with more complex artifacts. Recognizing that human designers use modularity, regularity and hierarchy in creating designs, Hornby created GENRE, an evolutionary design system in which the representational language for encoding designs consisted of a computer-program-like assembly procedure with conditionals, iterative loops, and procedures allowing for the evolution of hierarchical assembly of reusable building blocks [25], [26]. Using GENRE, he demonstrated the evolution of tables comprised of as many as several thousand blocks, as well as artificial neural networks, computer programs and robots. Several of the evolved designs were manufactured using rapid-prototyping technology (Figure 13 and 7).

Evolved Antenna Designs

Researchers have been investigating evolutionary antenna design and optimization since the early 1990s [5], [8], and the field has grown in recent years with algorithm improvements, increased computer performance, and higher-fidelity electromagnetics simulators. A variety of antenna types have been investigated, and most of the work has been centered on optimizing numerical parameters of a predetermined design as opposed to allowing the evolutionary algorithm design the antenna's topology.

In this section we describe an evolved antenna design and flight hardware currently on schedule to be deployed on a NASA spacecraft in 2006. The mission, Space Technology 5 (ST5) [10], consists of three satellites that will take measurements in Earth's magnetosphere. The evolved antennas [4] have unusual shapes and were evolved to meet a challenging set of mission

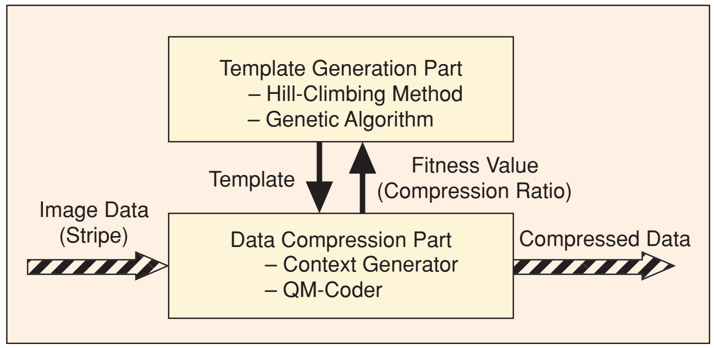


FIGURE 11 Data compression process showing the genetic algorithm.

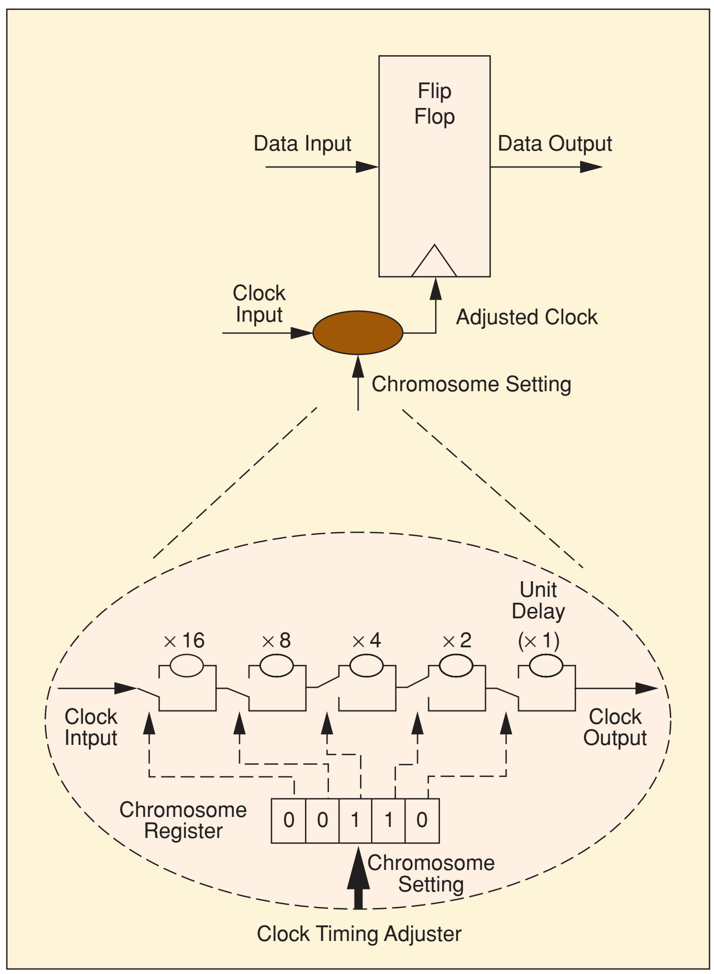


FIGURE 12 Clock timing adjustment architecture.

requirements, notably the combination of wide beamwidth for a circularly polarized wave and wide impedance bandwidth.

The key ST5 mission antenna requirements were: transmit and receive frequencies of 8470 and 7209 MHz, a omnidirectional gain pattern with at least -5 dBic between 0° and 40° of elevation and 0 dBic between 40° and 80°, and voltage standing wave ratios (VSWRs) of 1.2 and 1.5 at the transmit and receive frequencies, respectively.

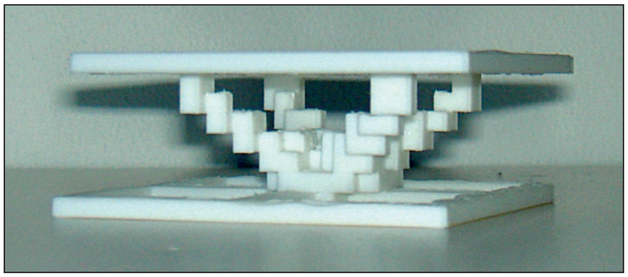
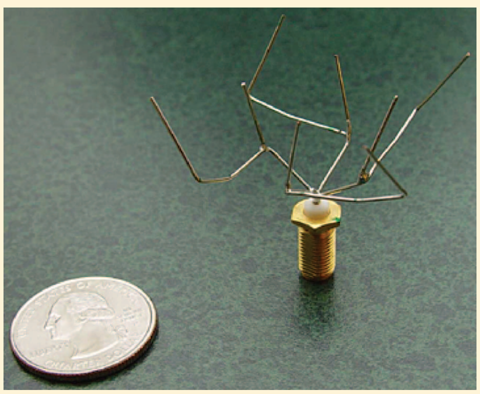
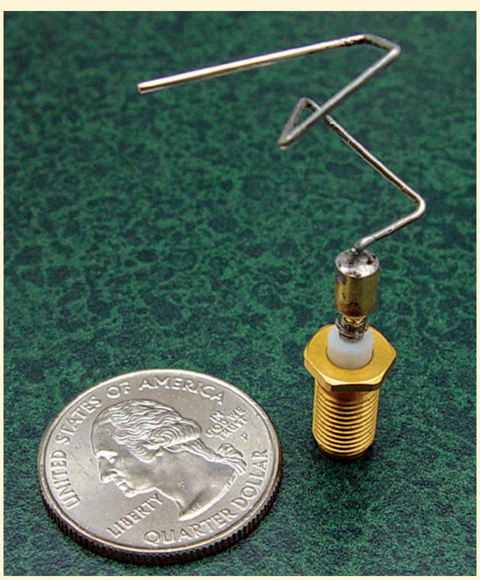


FIGURE 13 An evolved table consisting of several hundred blocks.



(a)



(b)

FIGURE 14 Photographs of ST5 evolved antennas: (a) Phase I prototype and (b) Phase II prototype.

Two sets of antenna designs were evolved and prototyped. In Phase I of the project, a four-branched symmetric design was favored. New antennas were evolved in Phase II due a requirements change. In this phase, a single wire without branches was preferred.

During the course of this project, a variety of artificial genotypes were used. One consisted of small antenna-constructing programs that are composed of commands from a simple programming language we devised. The language is composed of commands that "draw" wire segments and perform coordinate system rotations. An antenna design is created by starting with an initial feedwire and creating wires specified by executing the evolved antenna-constructing program. The command `forward(length, radius)` adds a wire with the given length and radius. The command `rotate-x(angle)` changes the coordinate system orientation by rotating it the specified amount about the x-axis. Similar commands are defined for the `y` and `z` axes. After constructing one antenna arm, the complete antenna is formed using four rotated copies of the evolved arm.

The fitness function used to evaluate antennas is the product of three scores involving the gain pattern, the VSWR, and pattern outliers. The gain pattern score compares the measured gain values across a range of elevation angles to the desired gain values and rewards antennas that exceed the requirements. VSWR is the ratio between the highest voltage and the lowest voltage in the signal envelope along a transmission line, with a ratio of 1 being ideal. The VSWR score is constructed to put strong pressure on driving VSWR values below the required values of 1.2 and 1.5. Since achieving gain values greater than 0 dBic is the main part of the require specifications, the outlier score rewards antenna designs for having sample points with gains greater than zero.

To take into account imprecision in manufacturing an antenna, antenna designs are evaluated four times: in each instance a small random perturbation is applied to joint angles and wire radii. The overall fitness of an antenna is the worst score of these evaluations. In this way, the fitness score assigned to an antenna design is a conservative estimate of how well it will perform if it were to be constructed. An additional side-effect of this is that antennas evolved with this manufacturing noise tend to perform well across a broader range of frequencies than do antennas evolved without this noise.

The setup for a typical run of the evolutionary algorithm consisted of a population of 200 individuals, with mutation and recombination applied with 50% probability. The algorithm was run until fitness scores were stagnant. The Numerical Electromagnetics Code, Version 4 [9] was used to evaluate all antenna designs.

The best antenna designs from both Phase I and II found was fabricated and tested (Figure 14). Compliancy with mission requirements was confirmed by testing the prototype antenna in an anechoic test chamber at NASA Goddard Space Flight Center.

One of the challenges in engineering design is adapting a set of created designs to a change in requirements. After the Phase I antennas had been produced, the mission's orbital vehicle was changed, putting it into a much lower earth orbit. This in turn changed the specifications for the antenna. With minimal changes to our evolutionary system, mostly in the fitness function, we were able to evolve antennas for the new mission requirements and, within one month of this change, two new antennas were designed and prototyped. Both antennas were tested and both had acceptable performance compared with the new specifications. This rapid response shows that evolutionary design processes are able to accommodate new requirements quickly and with minimal human effort.

The evolved antennas have a number of advantages in regard to power consumption, fabrication time and complexity, and performance. Lower power requirements result from achieving high gain across a wider range of elevation angles, thus allowing a broader range of angles over which maximum data throughput can be achieved. Since the evolved antenna does not require a phasing circuit, less design and fabrication work is required. In terms of overall work, the evolved antenna required approximately three person-months to design and fabricate whereas the conventional antenna required about five. Lastly, the evolved antenna has more uniform coverage in that it has a uniform pattern with small ripples in the elevations of greatest interest ($40° - 80°$). This allows for reliable performance as the elevation angle relative to the ground changes.

The evolved ST5 antennas represent the first antenna to be fielded with an evolved design topology, and, if successfully deployed into space in 2006, the first artificially evolved object to fly in space.

Conclusion

In this article we have given an introduction to the exciting and emerging field of evolvable hardware and highlighted some recent applications. After many years of development, evolvable hardware applications are maturing and seeing deployment into the real-world as fielded applications.

Evolvable hardware technology is not a panacea for all or even most hardware design and optimization problems. As with any new technology, there are certainly many unreported failures. But like evolution itself, the successful applications will survive and spawn wider interest. On the horizon we see more widespread acceptance and hence an increasing number of fielded applications as more technologists begin to use these methods.

References

[1] H. DeGaris, "EvolvableHardware:PrinciplesandPractice," http://www.cs.usu.edu/degaris/papers/CACM-E-Hard.html, Aug. 1997.

[2] B. Dunham, D. Fridshal, R. Fridshal, and J. North, "Design by natural selection," Synthese, D. Reidel Publication Company, Dordrecht-Holland, pp. 254–259, 1963.

[3] D. Goldberg, "Computer-Aided gas pipeline operation using genetic algorithms and rule learning Ph.D. dissertation," University of Michigan, 1983.

[4] J. D. Lohn, D.S. Linden, G. Hornby, W.F. Kraus, A. Rodriguez, and S. Seufert,"EvolutionaryDesignofanX-BandAntennaforNASA'sSpace Technology 5 Mission," Proc. 2004 IEEE Antenna and Propagation Society International Symposium and USNC/URSI National Radio Science Meeting, vol. 3, pp. 2313–2316, 2004.

[5] E. Michielssen, J.-M. Sajer, S. Ranjithan, and R. Mittra, "IEEE Trans. Microwave Theory & Techniques," vol. 41, no. 6, 1993.

[6] J. Koza, M. Keane, M. Streeter, W. Mydlowec, J. Yu, and G. Lanza, "Genetic ProgramminggIV:RoutineHuman-CompetitiveMachineIntelligence," Kluwer Academic Publishers, 2003.

[7] J. Koza, S. Al-Sakran and L. Jones, "Cross-domain features of runs of genetic programming Used to evolve designs for Analog Circuits, Optical Lens Systems. Controllers, Antennas, Mechanical Systems, and Quantum Computing Circuits," 2005 NASA/DoD Conference on Evolv-able Hardware, IEEE Computer Society Press, pp. 205–214, 2005.

[8] R.L. Haupt, "IEEE Antennas & Propagation Mag.," vol. 37, no. 4, 1995.

[9] G.J. Burke and A.J. Poggio, "Numerical Electromagnetics Code (NEC)-Method of Moments," Rep. UCID18834, Lawrence Livermore Lab. 1981.

[10] Space Technology 5 Mission, Website: http://nmp.jpl.nasa.gov/st5/

[11] S. Nolfi and D. Floreano, "Evolutionary robotics the biology, intelli-gence, and Technology of Self-Organizing Machines," MIT Press, 2000.

[12] T. Higuchi, M. Iwata, D. Keymeulen, H. Sakanashi, M. Murakawa, I. Kajitani, E. Takahashi, K. Toda, M. Salami, N. Kajihara and N. Otsu, "Real-World Applications of Analog and Digital Evolvable Hardware," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 3, Sept. 1999.

[13] T. Higuchi, M. Iwata, E. Takahashi, Y. Kasai, H. Sakanashi, M. Murakawa, and I. Kajitani, "Development of Evolvable Hardware at Elec-trotechnical Laboratory," IEEE International Conference on Industrial Electronics, Control and Instrumentation, 2000.

[14] A. Thompson, An evolved circuit, intrinsic in silicon, entwined with physics. 1st Int. Conf. on Evolvable Systems 1996, (c) Springer Verlag 1997.

[15] C. Kane and M. Schoenauer, "Topological optimum design," *Control and Cybernetics*, vol. 25, no. 5, pp. 1059–1088, 1996.

[16] P. Baron, A. Tuson, and R. Fisher, "A voxel-based representation for evolutionary shape optimisation," *Journal of Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Special Issue on Evolutionary Design*, vol. 13, no. 3, pp. 145–156, 1999.

[17] J. Frazer, *An Evolutionary Architecture*, Architectural Association Publications, 1995.

[18] M. Rosenman, "A growth model for form generation using a hierarchical evolutionary approach," *Microcomputer in Civil Engineering*, vol. 11, pp. 161–172, 1996.

[19] T. Broughton, A. Tan, and P.S. Coates, "The use of genetic programming in exploring 3d design worlds," in *CAAD Futures 1997*, R. Junge, Ed. Kluwer Academic, 1997, pp. 885–917.

[20] P. Coates, T. Broughton, and H. Jackson, "Exploring three-dimensional design worlds using lindenmayer systems and genetic programming," in *Evolutionary Design by Computers*, P.J. Bentley, Ed. San Francisco: Morgan Kaufmann, 1999, ch. 14.

[21] P.J. Bentley, "Generic evolutionary design of solid objects using a genetic algorithm," Ph.D. dissertation, Dept. of Engineering, University of Huddersfield, 1996.

[22] P.J. Bentley and J.P. Wakefield, "The evolution of solid object designs using genetic algorithms," in *Modern Heuristic Search Methods*, V. Rayward-Smith, Ed. John Wiley and Sons Inc., pp. 199–215, 1996.

[23] P. Funes and J. Pollack, "Computer evolution of buildable objects," in *Proceedings of the Fourth European Conference on Artificial Life*, P. Husbands and I. Harvey, Eds. Cambridge, MA: MIT Press, pp. 358–367, 1997.

[24] P. Funes and J.B. Pollack, "Evolutionary body building: Adaptive physical designs for robots," *Artificial Life*, vol. 4, no. 4, pp. 337–357, 1998.

[25] G.S. Hornby, "Functional scalability through generative representations: The evolution of table designs," *Environment and Planning B: Planning and Design*, vol. 31, no. 4, pp. 569–587, July 2004.

[26] G. Hornby, "Measuring, enabling and comparing modularity, regularity and hierarchy in evolutionary design," in *Proc. of the Genetic and Evolutionary Computation Conference*, H.-G. B. et al., Ed. New York: ACM Press, 2005.

[27] H. Lipson and J.B. Pollack, "Automatic design and manufacture of robotic lifeforms," *Nature*, vol. 406, pp. 974–978, 2000.

[28] K. Sims, "Evolving Virtual Creatures," in *SIGGRAPH 94 Conference Proceedings, ser. Annual Conference Series*, pp. 15–22, 1994.

[29] J. Ventrella, "Explorations in the emergence of morphology and loco-motion behavior in animated characters," in *Proceedings of the Fourth Workshop on Artificial Life*, R. Brooks and P. Maes, Eds. Boston, MA: MIT Press, 1994.

[30] G.S. Hornby, H. Lipson, and J.B. Pollack, "Generative representations for the automatic design of modular physical robots," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 4, pp. 703–719, 2003.

[31] V. Zykov, E. Mytilinaios, B. Adams, and H. Lipson, "Self-reproducing machines," *Nature*, vol. 435, no. 7038.

[32] M.H. Yim, Y. Zhang, and D. Duff, "Modulary robots," IEEE Spectrum, vol. 39, no. 2, pp. 30–34, 2002.

[33] D. Wettergreen and C. Thorpe, "Gait generation for legged robots," in *IEEE International Conference on Intelligent Robots and Systems.*, pp. 1413–1420, 1992.

[34] C. Ridderstrom, "Legged locomotion control—a literature survey," Dept. of Machine Design, Royal Institute of Technology, S-100 44 Stockholm, Sweden, Tech. Rep. TRITA-MMK 1999:27, ISSN pp. 1400–1179, Nov. 1999.

[35] M.A. Lewis, A.H. Fagg, and A. Solidum, "Genetic programming approach to the construction of a neural network for control of a walking robot," in *Proc. of Intl. Conf. on Robotics and Automation*, pp. 2618–2623, May 1992.

[36] F. Gruau and K. Quatramaran, "Cellular encoding for interactive evolutionary robotics," University of Sussex, Tech. Rep. pp. 425, 1996.

[37] J.C. Gallagher, R.D. Beer, K.S. Espenschied, and R.D. Quinn, "Application of evolved locomotion controllers to a hexapod robot," *Robotics and Autonomous Systems*, vol. 19, no. 1, pp. 95–103, 1996.

[38] N. Jakobi, "Running across the reality gap: Octopod locomotion evolved in a minimal simulation," in *Proceedings of Evorob 98*, P. Husbands and J.-A. Meyer, Eds. Springer Verlag, 1998.

[39] G.B. Parker, D.W. Braun, and I. Cyliax, "Learning gaits for the stiquito," in *Proc. of 8th Intl. Conf. on Advanced Robotics*, P. Husbands and J.-A. Meyer, Eds., pp. 285–290, 1997.

[40] G.S. Hornby, S. Takamura, J. Yokono, O. Hanagata, T. Yamamoto, and M. Fujita, "Evolving robust gaits with aibo," in *IEEE International Conference on Robotics and Automation*. IEEE, pp. 3040–3045, 2000.

[41] G.S. Hornby, S. Takamura, T. Yamamoto, and M. Fujita, "Autonomous evolution of dynamic gaits with two quadruped robots," *IEEE Transac-tions on Robotics*, vol. 21, no. 3, pp. 402–410, 2005.

[42] M. Fujita and H. Kitano, "Development of an autonomous quadruped robot for robot entertainment," *Autonomous Robotics*, vol. 5, pp. 1–14, 1998.

[43] D. Whitley, "The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best," in *Proc. of the Third Intl. Conf. on Genetic Algorithms*, J.D. Schaffer, Ed. MorganKaufmann, pp. 116–121, 1989.