Evolutionary Robotics for Legged Machines: From Simulation to Physical Reality

Hod Lipson, Josh Bongard, Victor Zykov, Evan Malone
Computational Synthesis Lab
Cornell University, Ithaca NY 14853, USA

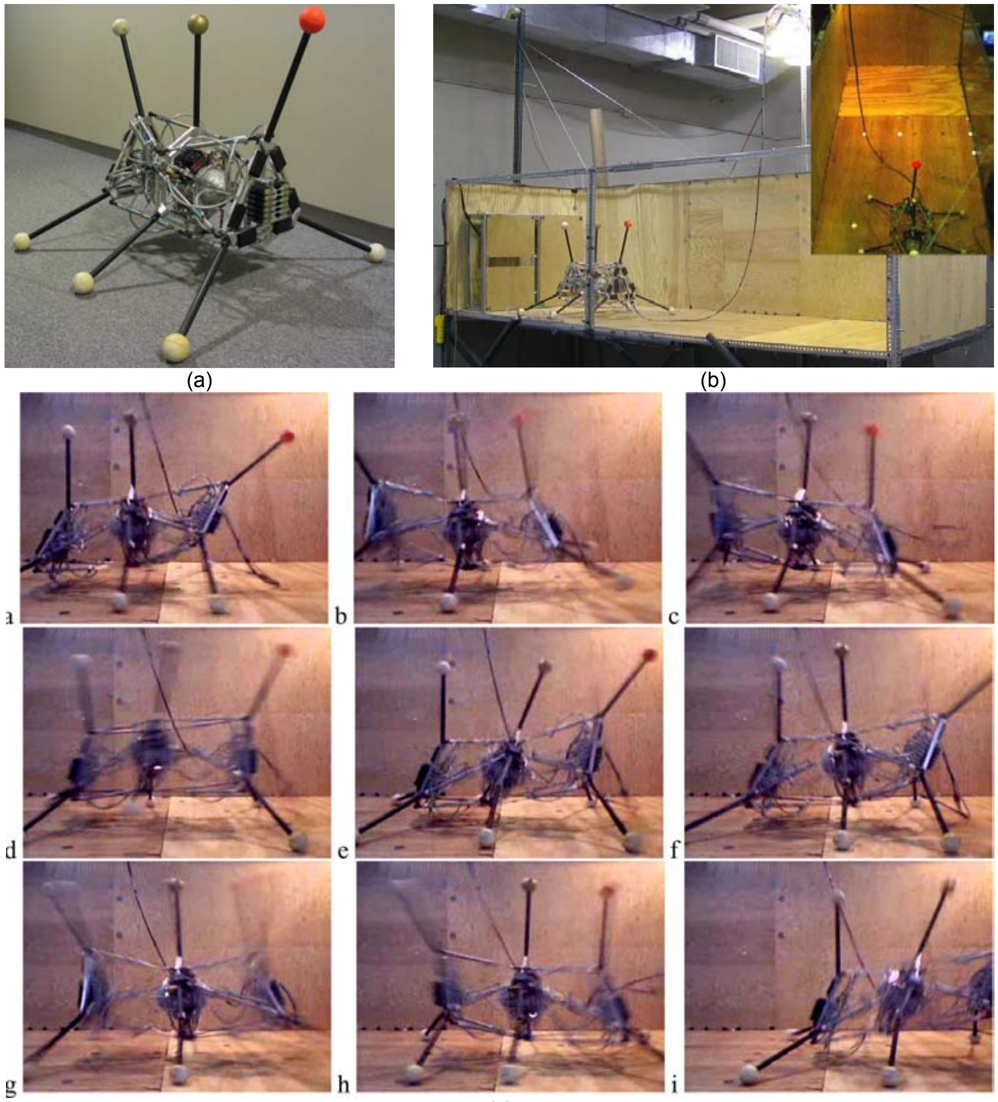
Abstract. This talk will outline challenges and opportunities in translating evolutionary learning of autonomous robotics from simulation to reality. It covers evolution and adaptation of both morphology and control, hybrid co-evolution of reality and simulation, handling noise and uncertainty, and morphological adaptation in hardware.

Keywords. Evolutionary robotics, co-evolutionary learning, estimation-exploration, rapid prototyping.

Introduction

The idea that machine learning processes inspired by biological evolution can be used to design autonomous machines, has its roots in the early days of evolutionary computation and has been implemented numerous times, starting with the seminal work of Sims [9]. Nevertheless, the transition of evolutionary robotics from simulation to reality has been met with many challenges, as is evident from the relatively few examples of successful implementations of these methods in physical reality. Though many robotic experiments are carried out in simulation, a robot must ultimately function in physical reality.

Consider the problem of evolving controllers for a dynamical, legged robot, shown in Figure 1 [13]. The nine-legged machine composed of two Stewart platforms back to back. The platforms are powered by twelve pneumatic linear actuators, with power coming from an onboard 4500psi paintball canister. While most robotic systems are use position-controlled actuators whose exact extension can be set, pneumatic actuators of the kind used here are force-controlled. Like biological muscle, the controller can specify the force and duration of the actuation, but not the position. It is therefore a challenging control problem. The controller architecture for this machine was an open-loop pattern generator that determines when to open and close pneumatic valves. The on-off pattern was evolved; Candidate controllers were evaluated by trying them out on the robot in a cage, and measuring fitness using a camera that tracks the red ball on the foot of one of the legs of the machine (see inset in Figure 1b for a view from the camera). Snapshots from one of the best evolved gates are shown in Figure 1c. Nolfi and Floreano [8] describe many other interesting hardware experiments evolving controllers for wheeled robots



(c)

Figure 1: Evolving a controller for physical dynamic legged machine: (a) The nine-legged machine is powered by twelve pneumatic linear actuators arranged in two Stewart platforms. The controller for this machine is an open-loop pattern generator that determines when to open and close pneumatic valves. (b) Candidate controllers are evaluated by trying them out on the robot in a cage, and measuring fitness using a camera that tracks the red foot (see inset). (c) Snapshots from one of the best evolved gates. From [13].

While evolution successfully generated viable gaits in this case, applying evolutionary processes to physical machines is difficult for two reasons. First, even if we are only evolving controllers for a machine with a fixed morphology, each evaluation of a candidate controller involves trying it out in reality. This is a slow and costly process that also wears out the target system. Performing thousand of evaluations is usually impractical. Second, if we are evolving morphology as well, then how would these morphological changes take place in reality? Changes to the controller can be done simply by reprogramming, but changes to the morphology require more sophisticated processes. Nature has some interesting solutions to this problem, such as growing materials, or self-assembling and self-replicating basic building blocks like cells.

1. Evolving controllers for physical morphologies

One approach to evolving controllers for fixed morphologies is to make a simulator with such fidelity that whatever works in simulation will also work in reality equally well. This is possible only for some types of locomotion, such as quasi-static kinematics that can be accurately predicted [6][4]. Figure 2a shows some of the machines that evolved for quasi-static locomotion in simulation; these machines were "copied" from simulation into reality using rapid-prototyping technology (Figure 2b) where they functioned in a way similar to their simulation. Unfortunately, however, it is unlikely that a similarly predictive *dynamic* simulator would exist, given that machine dynamics are inherently chaotic and sensitive to initial conditions and many small parameter variations. But even if such simulators existed, creating accurate models would be painstakingly difficult, or may be impossible because the target environment is unknown.

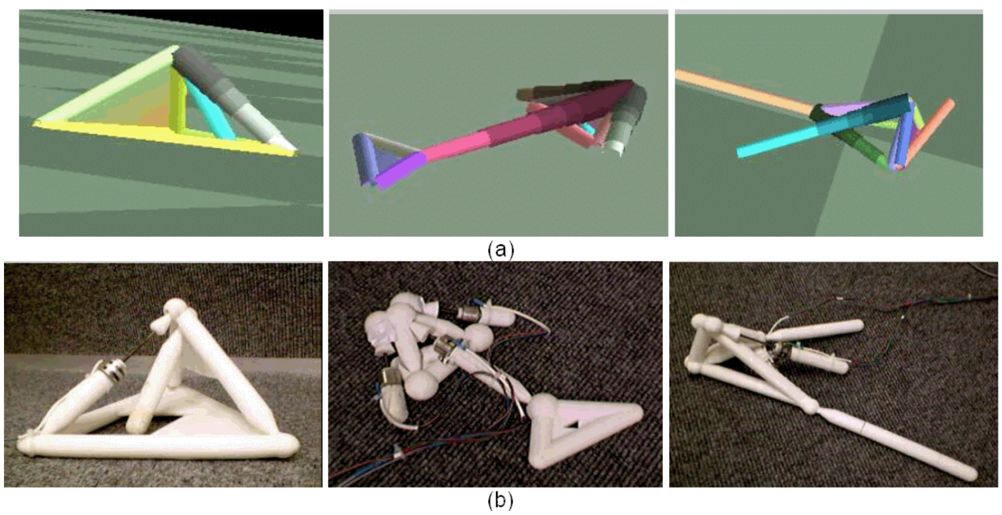


Figure 2: Evolving bodies and brains: (a) Three evolved robots, in simulation (b) the three robots reproduced in physical reality using rapid prototyping. From [6].

An alternative approach to "crossing the reality gap" is to use a crude simulator that captures the salient features of the search space. Techniques have been developed for creating such simulators and using noise to cover uncertainties so that the evolved controllers do not exploit these uncertainties [5]. Yet another approach is to use plasticity in the controller: Allow the robot to learn and adapt in reality. In nature, animals are born with mostly predetermined bodies and brains, but these have some ability to learn and make final adaptations to whatever actual conditions may arise.

A third approach is to co-evolve simulators so that they are increasingly predictive. Just as we use evolution to design a controller, we can use evolution to design the simulator so that it captures the important properties of the target environment. Assume we have a rough simulator of the target morphology, and we use it to evolve controllers in simulation. We then take the best controller and try it – once – on the target system. If successful, we are done; but if the controller did not produce the anticipated result (as is likely to happen since the initial simulator was crude), then we observed some unexpected sensory data. We then evolve a new set of simulators, whose fitness is their ability to reproduce the actual observed behavior when the original controller is tested on them. Simulators that correctly reproduce the observed data are more likely to be predictive in the future. We then take the best simulator, and use to evolve a new controller, and the cycle repeats: If the controller works in reality, we are done. If it does not work as expected, we now have more data to evolve better simulators, and so forth. The co-evolution of controllers and simulators is not necessarily computationally efficient, but it dramatically reduces the number of trials necessary on the target system.

The co-evolutionary process consists of two phases: Evolving the controller (or whatever we are trying to modify on the target system) – we call this the exploration phase. The second phase tries to create a simulator, or model of the system – we call this the estimation phase. To illustrate the estimation-exploration process, consider a target robot with some unknown, but critical, morphological parameters, such as mass distribution and sensory lag times. Fifty independent runs of the algorithm were conducted against the target robot. Figure 3a shows the 50 series of 20 best simulator modifications output after each pass through the estimation phase. Figure 3a makes clear that for all 50 runs, the algorithm was better able to infer the time lags of the eight sensors than the mass increases of the nine body parts. This is not surprising in that the sensors themselves provide feedback about the robot. In other words, the algorithm automatically, and after only a few target trials, deduces the correct time lags of the target robot's sensors, but is less successful at indirectly inferring the masses of the body parts using the sensor data. Convergence toward the correct mass distribution can also be observed. But even with an approximate description of the robot's mass distribution, the simulator is improved enough to allow smooth transfer of controllers from simulation to the target robot. Using the default, approximate simulation, there is a complete failure of transferal: the target robot simply moves randomly, and achieves no appreciable forward locomotion. It is interesting to note that the evolved simulators are not perfect; they capture well only those aspects of the world that are important for accomplishing the task.

The exploration-estimation approach can be used for much more than transferring controllers to robots – it could be used by the robot itself to estimate its own structure. This would be particularly useful if the robot may undergo some damage that changes some of its morphology in unexpected ways, or some aspect in its environment changes. As each controller action is taken, the actual sensory data is compared to that predicted by the simulator, and new internal simulators are evolved to be more predictive. These new simulators are then used to try out new, adapted controllers for the new and unexpected circumstances. Figure 3b shows some results applying this process to design controllers for a robot which undergoes various types of drastic morphological damage, like loosing a leg, motor, or sensor, or combinations of these. In most cases, the estimation-exploration process is able to reconstruct a new simulator that captures the actual damage using only 4-5 trials on the target robot, and then use the adapted simulator to evolve compensatory controllers that recover most of the original functionality. There are numerous applications to this identification and control process in other fields.

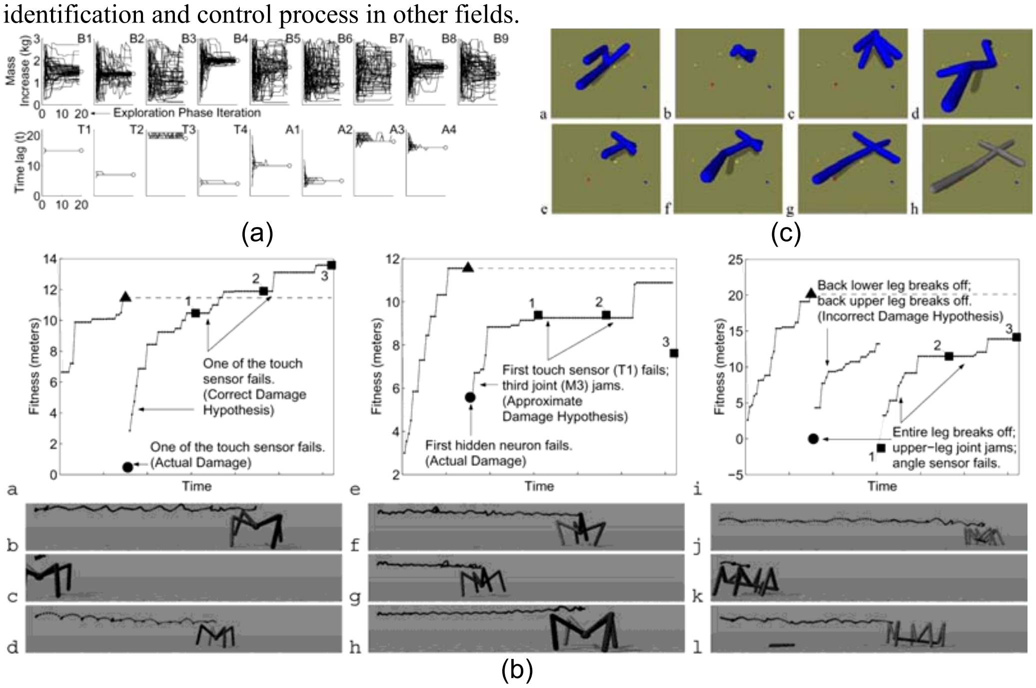


Figure 3: Co-evolving robots and simulators: (a) Convergence toward the physical characteristics of the target robot. Each pass through the estimation phase produces a set of mass changes for each of the nine body parts of the robot (top row) and a set of time lags for each of the eight sensors (bottom row). The open circles indicate the actual differences between the target robot and the starting default simulated robot [1]. (b) Three typical damage recoveries. a: The evolutionary progress of the four sequential runs of the exploration EA on the quadrupedal robot, when it undergoes a failure of one of its touch sensors. The hypotheses generated by the three runs of the estimation EA (all of which are correct) are shown. The dots indicate the fitness of the best controller from each generation of the exploration EA. The triangle shows the fitness of the first evolved controller on the target robot (the behavior of the 'physical' robot with this controller is shown in b); the filled circle shows the fitness of the robot after the damage occurs (the behavior is shown in c); the squares indicate the fitness of the 'physical' robot for each of the three subsequent hardware trials (the behavior of the 'physical' robot during the third trial is shown in d). e-h The recovery of the quadrupedal robot when it experiences unanticipated damage. i-l The recovery of the hexapedal robot when it experiences severe, compound damage. The trajectories in b-d, f-h and j-l show the change in the robot's center of mass over time (the trajectories are displaced upwards for clarity) [2]. (c) The simulator progressively learns the entire robot morphology from scratch. Panels (a-g) are progressive intermediate self-inference stages, panel (h) is the true target system [3].

2. Making morphological changes in hardware

An evolutionary process may require a change of morphology, or production of a new physical morphology altogether. One approach for generating new morphology is to use reconfigurable robots [12]. Reconfigurable robots are composed of many modules that can be connected, disconnected and rearranged in various topologies to create machines with variable body plans. Self-reconfigurable robots are able to rearrange their own morphology, and thus adapt in physical reality. Figure 4a shows one example of a self-reconfiguring robot composed of eight identical cubes [14]. Each cube can swivel around its (1,1,1) axis, and connect and disconnect to other cubes using electromagnets on its faces. Though this robot contains only 8 units, it is conceivable that future machine will be composed of hundreds and thousands of modules of smaller modules, allowing much greater control and flexibility in morphological change.

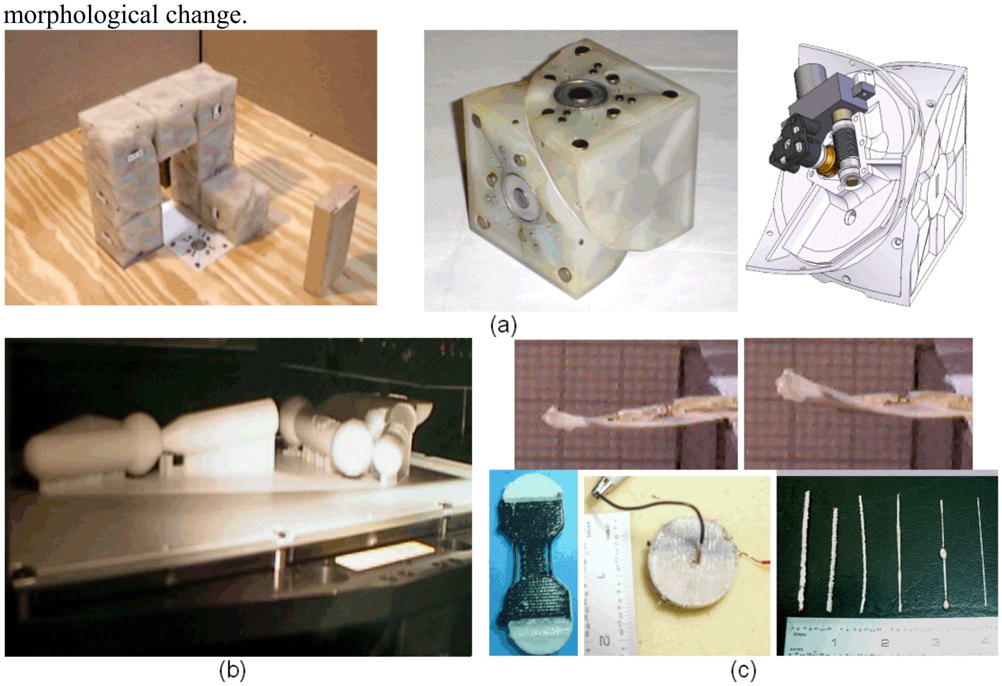


Figure 4: Transferring morphological changes into reality (a) Reconfigurable *molecube* robots [14], (b) Rapid prototyping, (c) Future rapid prototyping systems will allow deposition of multiple integrated materials, such as elastomers, conductive wires, batteries and actuators, offering evolution a larger design space of integrated structures, actuators and sensors, not unlike biological tissue. From [7].

An alternative approach to varying morphology is to produce the entire robot morphology automatically. For example, the robots shown in Figure 2b were produced using rapid prototyping equipment: These are 3D printers, that deposit material layer by layer to gradually build up a solid object of arbitrary geometry, as shown in Figure 4b. This “printer”, when coupled to an evolutionary design process, can produce complex geometries that are difficult to produce any other way, and thus allow the evolutionary search much greater design flexibility. But even when using such automated fabrication equipment we needed to manually insert the wires, logic, batteries and actuators. What if the printer could print these components too? Future rapid prototyping systems may allow deposition of multiple integrated materials, such as elastomers, conductive wires, batteries and actuators, offering evolution an even larger design space of integrated structures, actuators and sensors, not unlike biological tissue. Figure 4c shows some of these printed components [7].

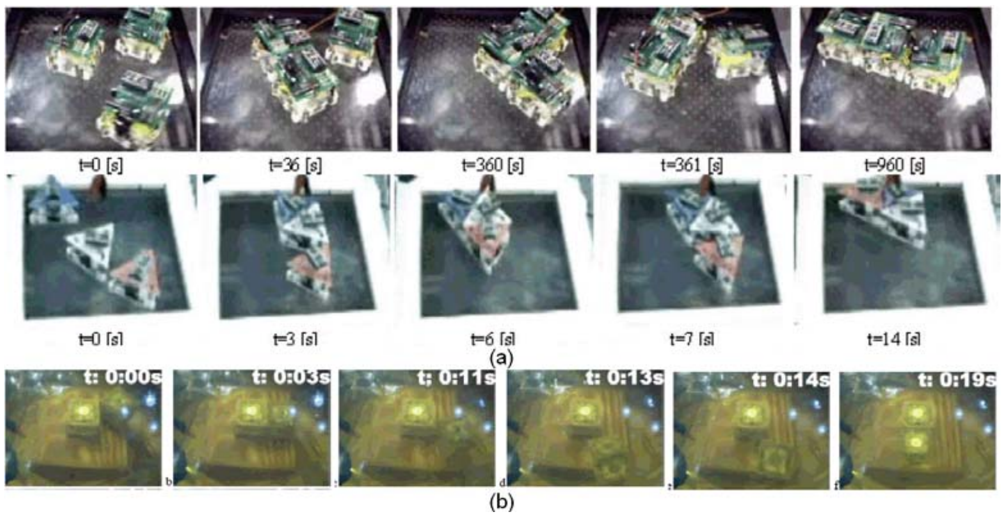


Figure 5: **Macro-scale physical models of stochastic self-assembly**. (a) Stochastic self-assembly and self reconfiguration of 10-cm scale modules on an oscillating air table: Top: units with electromagnets; Bottom: Units with swiveling permanent magnets [10]. (b) Three dimensional stochastic self assembly and reconfiguration of 10-cm cubes in oil [11].

Looking at biology, one would ultimately like to emulate 'growing structures' – structures that can actively move material from one place to another, adapting to needs in situ. As we move to smaller and smaller scales, however, deterministically moving material becomes increasingly difficult. An interesting alternative is to exploit the random 'Brownian' motion of the particles in the environment to assist in stochastic self assembly. Figure 5 shows some macro-scale prototypes of such stochastically reconfiguring systems, both in 2D and in 3D. Implementation of such systems at the micro scale, using many thousands of units, entails many physical as well as computational challenges, involving local actuation, sensing, and control.

3. Conclusions

The transition of evolutionary robotics methods from simulation to reality has met several hurdles: Besides the scaling limits of evolutionary computation itself, we are confronted with the limits of simulation and modeling, the cost, time and risk of training machines in reality, and the technical challenge of adapting morphology in reality. Again we have resorted to inspiration from biology: Better ways to design adaptive simulations, better ways to determine the most useful physical evaluation, and new ways to adapt physical morphology through automatic reconfiguration and material growth, all leading to new ideas for engineering and new engineering insights into biology.

4. References

[1] Bongard J. C., Lipson H., (2004) "Once More Unto the Breach: Automated Tuning of Robot Simulation using an Inverse Evolutionary Algorithm", Proceedings of the Ninth Int. Conference on Artificial Life (ALIFE IX), pp.57-62

[2] Bongard J., Lipson H. (2004), "Automated Damage Diagnosis and Recovery for Remote Robotics", IEEE International Conference on Robotics and Automation (ICRA04), pp. 3545-3550

[3] Bongard, J. and Lipson, H. (2004) "Integrated Design, Deployment and Inference for Robot Ecologies", Proceedings of Robosphere 2004, November 2004, NASA Ames Research Center, CA USA

[4] Hornby G.S., Lipson H., Pollack. J.B., 2003 "Generative Encodings for the Automated Design of Modular Physical Robots", IEEE Transactions on Robotics and Automation, Vol. 19 No. 4, pp 703-719

[5] Jakobi, N. (1997). Evolutionary robotics and the radical envelope of noise hypothesis. Adaptive Behavior, 6(1):131–174.

[6] Lipson H., Pollack J. B. (2000) Automatic design and manufacture of artificial lifeforms. Nature, 406:974–978

[7] Malone E., Lipson H., (2004) "Functional Freeform Fabrication for Physical Artificial Life", Ninth Int. Conference on Artificial Life (ALIFE IX), Proceedings of the Ninth Int. Conference on Artificial Life (ALIFE IX), pp.100-105

[8] Nolfi S., Floreano D. (2004), Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines, Bradford Books

[9] Sims K. "Evolving 3D morphology and behaviour by competition". Artificial Life IV, pages 28–39, 1994.

[10] White P. J., Kopanski K., Lipson H. (2004), "Stochastic Self-Reconfigurable Cellular Robotics", IEEE International Conference on Robotics and Automation (ICRA04), pp. 2888-2893

[11] White P., Zykov V., Bongard J., Lipson H. (2005) Three Dimensional Stochastic Reconfiguration of Modular Robots, Proceedings of Robotics Science and Systems, MIT, Cambridge MA, June 8-10, 2005

[12] Yim, M., Zhang, Y. and Duff, D., "Modular Reconfigurable Robots, Machines that shift their shape to suit the task at hand," IEEE Spectrum Magazine cover article, Feb. 2002

[13] Zykov V., Bongard J., Lipson H., (2004) "Evolving Dynamic Gaits on a Physical Robot", Proceedings of Genetic and Evolutionary Computation Conference, Late Breaking Paper, GECCO'04

[14] Zykov V., Mytilinaios E., Adams B., Lipson H. (2005) "Self-reproducing machines", Nature Vol. 435 No. 7038, pp. 163-164