Embodied Evolution: A Response to Challenges in Evolutionary Robotics

Sevan G. Ficici, Richard A. Watson, and Jordan B. Pollack *

**DEMO Lab, Department of Computer Science, Brandeis University, Waltham Massachusetts USA*
{sevan, richardw, pollack}@cs.brandeis.edu—www.demo.cs.brandeis.edu

Abstract. We introduce Embodied Evolution (EE), a new methodology for conducting evolutionary robotics (ER). Embodied evolution uses a population of physical robots that evolve by reproducing with one another in the task environment. EE addresses several issues identified by researchers in the evolutionary robotics community as problematic for the development of ER. We review results from our first experiments and discuss the advantages and limitations of the EE methodology.

1 Introduction

As an alternative to the hand design of robotic controllers, evolutionary robotics (ER) (Harvey et al., 1993; Cliff et al., 1993; Husbands and Harvey, 1992) has generated a great deal of exuberance. While expected by many to emerge as an important technology (Meyer et al., 1998), ER is currently a new research area in which a number of potentially serious problems are known to exist. Comprehensively summarized by (Mataric and Cliff, 1996), these issues ultimately question whether ER techniques can produce a net savings of human effort when applied to complex robotic domains.

Our work is inspired by the vision of a large population of robots that reproduce with one another and evolve in their task environment. This vision, to our knowledge first described in (Husbands et al., 1992), aspires to an ideal where the robot population evolves in a completely hands-free and autonomous manner; in so doing, it offers intriguing possibilities for the future of evolutionary robotics. Nevertheless, many substantial technological demands are made by this conception of robot evolution, and considerable algorithmic detail must be added before it is implementable.

We have developed this vision into a working implementation and have termed our methodology *embodied evolution* (EE). We define EE as evolution taking place within a population of real robots where evaluation, selection, and reproduction are carried out by and between the robots in a distributed, asynchronous, and autonomous manner. Thus, we distinguish embodied evolution from ER methods that use *embodied trials*—the serial evaluation of candidate controllers on one or a small number of real robots (Harvey et al., 1993; Floreano and Mondada, 1994; Floreano and Mondada, 1996; Nolfi, 1997)—as well as algorithms that maintain and manipulate the specifications of individual agents in a centralized manner. As an intrinsically population-based method where robots adapt in the task environment, EE provides an intersection between evolutionary robotics and *collective* robotics: EE potentially offers an ideal substrate with which to study emergent group behavior and explore mechanisms that adaptively discover problem decomposition. As well as providing a substrate for studying collective agent behavior, the distributed nature of the EE architecture gives its adaptive mechanism both scalability (with respect to the number of robots used) and robustness (with respect to hardware failures).

We review some of the technical and algorithmic details that have enabled the first embodied evolution experiments and discuss the advantages and limitations of our methodology, particularly with respect to the many problematic issues that face the evolutionary robotics community, as identified by (Mataric and Cliff, 1996). In the following sections we outline our implementation of the embodied evolution concept, describe experiments and give results, then enter a discussion of related work and detail the issues raised by Mataric and Cliff that EE does and does not address, and finally give concluding remarks and directions for future work.

2 Implementing EE

One requirement for implementing embodied evolution is the development of an appropriate evolutionary algorithm. The principal components of any evolutionary algorithm are evaluation, selection, and reproduction, and all of these are carried out autonomously by and between the robots in a distributed fashion according to our definition of embodied evolution (the implications of this are discussed below).

Because the process of evaluation is carried out autonomously by each robot, some metric must be programmed into the robots with which they can measure their performance. This can be quite implicit, for example, where failing to maintain adequate power results in "death" (Mondada and Floreano, 1996). Or, it can be explicitly hard-coded, for example, where fitness is a function of objects collected and time. Whatever metric is used, performance against it must be monitored by the robot itself, as no external observer exists to measure a robot's ability explicitly.

Reproduction must also be both distributed and asynchronous in EE. Assuming that we can not really create new robots spontaneously, the offspring must be implemented using (other) robots of the same population. And, if we do not have structurally reconfigurable bodies, reproduction must simply mean the exchange of genetic information that codes control programs.

In general, selection in an evolutionary algorithm may be realized by having more-fit individuals supply genes (i.e., be parents) or by having less-fit individuals lose genes (i.e., be replaced by the offspring) or by a combination of both. The Microbial GA (Harvey, 1996) uses this observation to simplify the steady-state genetic algorithm; rather than pick two (above-average fitness) parents and produce an offspring from the combination of their genes to replace a (below-average) third, the Microbial GA selects two individuals at random and overwrites some of the genes of the less fit (of the two) with those from the more fit. In effect, the less fit of the two becomes the offspring. To achieve decentralized and asynchronous reproduction in EE, we have developed a probabilistic version of the Microbial GA that we call the Probabilistic Gene Transfer Algorithm (PGTA). This algorithm requires minimal inter-agent communication, and eliminates the need to coordinate the communication of each reproduction event.

In the PGTA, reproduction is concurrent with task behavior—there is no "reproduction mode" as such. Each robot maintains a virtual energy level, which reflects the robot's performance at the task, and each robot probabilistically broadcasts genetic information on its local-range communication channel at a rate proportional to this energy level. Each broadcast contains a mutated version of one randomly-selected gene from the robot's genome (i.e., one parameter from the robot's control specification). If another robot receives the broadcast, that robot may allow the received gene value to overwrite its own corresponding gene. The receiving robot will accept the broadcast gene with a probability inversely related to its own energy level. Robots with higher energy thus attempt to reproduce, and resist the reproductive attempts of others, more frequently than do robots with lower energy. Nevertheless, because sending and receiving is probabilistic, and genes are picked at random, the PGTA does not guarantee that a fitter robot will transfer all its genes to a less fit robot. On average robots are left with a mixture of genes in proportion to their relative energy levels. This implements a fitness-proportionate recombinative evolutionary algorithm.

Using the PGTA, each reproduction event requires only unidirectional communication—there is no need for robots to coordinate reproductive acts, for a robot to know the fitness or identity of another robot, or even to know that any robot received its broadcast. The PGTA thus allows the complete decentralization of selection and reproduction. Though the PGTA provides an interesting mechanism for EAs in general, its robustness to genetic information "dropped" in communication makes the PGTA particularly advantageous for implementation in a population of real robots.

3 Experiments and Results

3.1 Setup

Our first experiments in EE used a population of eight of our custom-built robots, which employ the "Cricket" micro-controller board (supplied by the MIT Media Laboratory (Resnick et al., 1997)). Each robot has a 12cm diameter and is equipped with two light sensors, two motors, and an infra-red emitter/detector pair that provides local communication. The transfer of genetic material during reproduction is performed via local broadcasts ('narrowcasts') on the infra-red communications channel. Thus reproduction events occur according to the movements and co-locations of the robots. This limited communication range (approximately one body width in radius) combined with the freely-mixed population of our shared environment essentially implements random selection of mates, as appropriate for the PGTA. The control architecture is a small feed-forward artificial neural network, the weights of which are evolved to perform phototaxis similar to that described in (Braitenberg, 1984). The network consists of two output nodes, one for each of the two motors, one binary-valued input node, which indicates which of the robot's two light sensors is receiving more light, and one bias node that has a constant activation.

The task environment consists of a 130cm by 200cm pen with a lamp located in the middle, visible from all positions on the floor plane. The robot task is to reach the light from any starting point in the pen. An infra-red beacon mounted above the light emits a signal that robots detect when they reach the light source. The beacon signal triggers a built-in reset behavior that moves the robot to a random position and orientation along the periphery of the pen, from where the robot recommences its light-seeking behavior. If a robot's sensor values do not change for some period of time, the robot assumes that it is stuck against a wall and invokes a second built-in behavior that attempts to free the robot by rotating it a random amount. Both of these built-in behaviors operate independently of the evolving neural-network controller.

The virtual energy level maintained by a robot is updated as follows: whenever a robot reaches the light its energy is increased by a fixed amount, up to a maximum energy value; whenever a robot sends a gene for reproduction (regardless of whether another robot receives or accepts the gene) its energy is decreased by a small fixed amount, down to a minimal energy value. Since the robot's rate of sending genes is proportional to its energy level and decrements occur with each send, the rate of broadcasting decays exponentially over the time from its most recent visit to the light. The energy level thus approximates a leaky integral of the robot's performance at its task (i.e., the frequency with which it reaches the light). Experimental details can be found in (Watson et al., 1999).

3.2 Results

Figure 1 shows the frequency with which the light is successfully reached by the robot population over time in each of three experiments. The main experiment evolves the neural-network weights to perform the light-seeking task. The initial condition for the networks is that all weights have a value of zero (this configuration produces no output to the motors and provides a neutral starting point). The other two experiments are controls where the robots do not evolve; in one case the robots' weights are random values, in the other the robots use weights of a hand-designed solution.

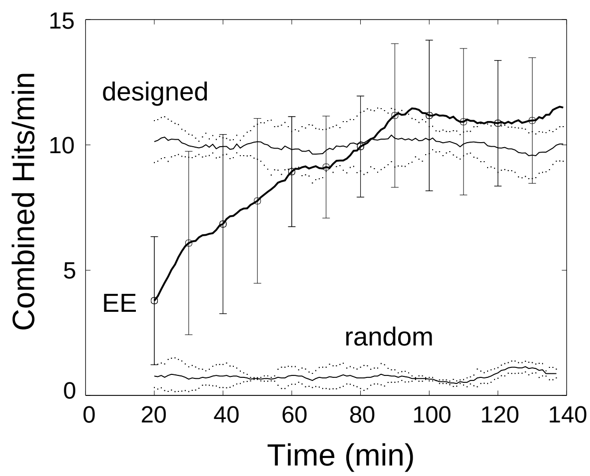

Figure 1: Performance Over Time. Upon detecting the beacon's signal, a robot sends a reply signal to the beacon. These replies are then delivered to a desktop computer where they are time-stamped. Three solid curves show the performance of the robot population using hand-designed (non-evolved), evolved, and random (non-evolved) networks. The data from the hand-designed and evolved experiments are averaged over six runs, while the data from the random-solution experiment are averaged over two runs. Each run lasts 140 minutes and uses eight robots. The vertical axis represents the average rate (in hits per minute) at which the group of robots reaches the light. A time window of 20 minutes is used to compute the instantaneous hit rate for each data point on the graph (hence the first data points appear at Time = 20 minutes). Vertical bars on the evolved run, shown every 10 minutes, and the dotted lines on the control experiments, show +/- one standard deviation. Though the evolved solutions begin with network weights of zero, we see that the robots achieve an average performance of four hits per minute within the first twenty minutes of the experiment and eventually exceed the hand-designed hit rate (the Wilcoxon rank-sum test gives $p = 0.935$).

As Figure 1 shows, the two controls show a broad range of possible performance levels and provide useful references against which to judge the success of the trials where evolution takes place. We see that embodied evolution allows the population of robots to achieve performance favorably comparable to that of our hand-designed solution—the Wilcoxon rank-sum test [1] indicates that EE outperforms the designed solution with probability $p = 0.935$. These results provide the first evidence that a fully decentralized, asynchronous evolutionary algorithm, can operate effectively in a population of physical robots and provide high-quality control programs.

There are several points of interest in this result. First, though the robots learn to approach the light in a multi-robot environment, they are able to perform effectively in isolation, as well (not shown here). Second, the evolved solutions exhibit behaviors that are qualitatively different from our hand-designed solution; evolution appears to favor a "looping" solution, whereas, with our hand-designed solution, the robot "swaggers" to the light. The reasons for this are not known as yet. Finally, the energy level maintained by a robot is an odd representation of its performance compared to the usual meaning of "fitness." In our implementation, the energy level is not reset in a robot when it receives a new gene during a reproductive event—and so, the energy level is related to the performance of the various controllers that have been resident on that robot. In contrast, one would not normally expect the fitness of previously-resident controllers to affect the current fitness of a robot. However, assuming that the offspring is similar to the parent, our method of using inherited energy potentially reduces the number of trials that must take place before the fitness measure of a new controller is reliable.

4 Discussion

Mataric and Cliff's thorough assessment of the challenges that face evolutionary robotics (Mataric and Cliff, 1996) attracted considerable attention in the ER community. It provides a useful framework in which we can consider the contributions made by the embodied evolution methodology. We now revue how EE addresses the major issues that they raise.

[1] The Wilcoxon test is preferred over the Student t-test in this case because the variances of the designed and evolved runs are very different from each other.

4.1 Simulation

- "The difficulty of accurately simulating physical systems is well known in robotics." (Mataric and Cliff, 1996, p. 76)

- "As the complexity of robotic systems grows and the gap between the simulation and the real system widens, the question of the value of investing in a specialized simulation will become increasingly important." (Mataric and Cliff, 1996, p. 76)

Mataric and Cliff, in agreement with (Brooks, 1992), raise the concern that a lack of simulation fidelity can lead to problems of *transference*, where robotic controllers that evolve in simulation are unable to perform effectively when transferred to real robots because they exploit features of the simulator that are not found in the physical world. They argue that this problem is magnified as the system to be modeled becomes more complex. Indeed, even virtual worlds that are not intended to faithfully model the real-world can become complex and difficult to create; the ground-breaking work of (Sims, 1994), for example, has yet to be fully duplicated.

More recently, a method has been devised to provably eliminate transference risks from the design of a simulator, but only provided that the environmental factors responsible for correct behavior are known *a priori* (Jakobi, 1997a; Jakobi, 1997b). This allows the environmental factors that are not salient to the robot to be approximated, which minimizes the computational requirements of the simulation. However, as environments continue to become more complex (for example, if they involve a multiplicity of robots, or robots with high-resolution sensory apparatus such as vision), the critical environmental factors will become more difficult to ascertain, more difficult to model, and more costly to simulate, arguably to the extent that simulation is prohibited.

Because embodied evolution does not use simulation—evaluations are conducted with real robots in a real environment—these issues are avoided. Evaluations have perfect fidelity and the problem of transference is side-stepped completely.

4.2 Time

- "Evolution on physical systems takes prohibitively long." (Mataric and Cliff, 1996, p. 76)

- "[Consider a problem of]...five free parameters, each of which is tested with four different values, [and where]...each trial takes 15 s [seconds] ... With a population of size 100...the 100th generation will finish in roughly five years." (Mataric and Cliff, 1996, p. 80)

The issue of time raised by Mataric and Cliff is in contrast to the supposed speed of simulation. Though simplifying assumptions and stochastic approximations minimize simulator complexity, they do not eliminate it; a sufficiently complex environment can still cause simulation to run slower than real time. Embodied trials, although they may be slower than approximate models of simple domains, are never slower than real time and they have perfect fidelity.

Nevertheless, serial evaluation of candidate robot controllers on a single robot (Floreano and Mondada, 1994; Floreano and Mondada, 1996) can take a very long time, indeed. Parallelization can alleviate this concern for both simulated and embodied approaches. Even in Mataric and Cliff's example scenario above, which they characterize as "exaggerated," a population of 100 robots reduces five years to 18 days (5 years = 1825 days; dividing by population size of 100 gives 18). However, where robots perform interactive tasks in a shared environment, parallelizing simulation is not trivial: for example, collision detection and mutual-sightedness are problematic. Also, a centralized method of evaluation using real robots will potentially suffer from communication bottlenecks (Martinoli et al., 1997). In contrast, the distributed architecture of EE is intrinsically scalable with respect to the number of robots.

Our distributed steady-state PGTA evolutionary algorithm also prevents pathologically long evaluations from delaying other evaluations from starting. These time savings obviously come at the cost of significant hardware duplication. Yet, as large-scale multi-agent domains become increasingly important arenas of research, large numbers of robots will become commonplace; with EE, we merely exploit the ubiquity of hardware as it becomes available.

4.3 Power

- "The unavoidable need to recharge robot batteries further slows down the experimental procedure." (Mataric and Cliff, 1996, p. 76)

The issue of power pervades robotics. Battery-powered robots can run for a period only on the order of hours, usually no more than two or three, before the batteries either run out or require recharging. The use of recharging stations, however, is not transparent to the domain task because it interrupts task behaviors for non-trivial amounts of time. Tethers provide continuous power, but easily tangle if used on more than a few robots.

We have developed and refined a powered-floor technology that transparently provides continuous, untethered power to our robots, without the use of recharging stations. Our powered floor is surfaced with strips of stainless-steel tape, which are alternately connected to the positive and negative poles of a DC power source. Each robot has four contact points on the underside of its body, with which it draws power from the floor. The geometry of the contacts guarantees that at least one point can make contact with each pole of the DC power supply, regardless of the rotation or translation of the robot on the floor. Nevertheless, a contact switches polarity according to the position of the robot on the floor, and so the power is rectified before being delivered to the robot's controller and motors. A rechargeable cell is used to cover intermittentcy in contact.

While building our powered floor, we learned of two other research groups that have built floors of similar construction (Martinoli et al., 1997; Keating, 1998). These parallel achievements attest to the viability and utility of this power supply approach. Other approaches (AAIS, 1998), like earlier prototypes of our own, use a floor-and-ceiling "bumper-car" style set-up. Together these examples demonstrate that the issue of power delivery is not a fundamental restriction to the development of evolutionary robotics, at least in laboratory conditions, if not some industrial settings as well. While technologies such as the powered floor are important to the implementation of our experiments, we do not consider them to be an intrinsic part of the EE methodology.

4.4 Robustness

- "...a robotic system cannot survive the necessary continuous testing..." (Mataric and Cliff, 1996, p. 76)

Robots used in research are rarely endowed with the robustness that is engineered into industrial robots, usually for reasons of economy of development time or expense. As a result, research robots demand almost constant care and attention to keep them in operational order. The robots we built for our experiments are no exception in this respect.

Nevertheless, the population of robots that EE uses is a valuable source of redundancy, which allows the performance of the evolutionary system to degrade gracefully with the number of robot failures. There is even potential for the evolutionary system to learn to avoid destructive behavior. While we concede that physical failures are inevitable for long running times, especially when many robots are involved in physical interaction, the parallelism of embodied evolution reduces the amount of run-time per robot by a factor equal to the size of the robot population.

4.5 Other Issues

Mataric and Cliff raise several other issues that pertain to any method of machine learning, or even to hand-design methods. While our EE methodology is silent on these points, they are important to recognize. One such point in particular concerns the creation of an effective metric of agent success; all automated learning methods, including embodied evolution, require feedback to function. The question of how researchers are to construct good metrics of behavior for autonomous robots, especially as environments become more complex and interactive, is of vital importance and will continue to require special attention.

5 Where is EE suitable?

The core strengths of EE stem from its distributed architecture. Particularly, EE has potential to scale to very large systems (on the order of hundreds or thousands of robots). Embodied evolution is also particularly suited in any of the following circumstances, which we assert will become increasingly prevalent. All but the first item are out of reach for traditional ER approaches. Taken together, the points below provide a strong motivation for a distributed, embodied approach:

- where the task domain cannot be simulated, for whatever reason, or where a simulator is not available.

  Minimally complex simulations of multi-agent systems will eventually run slower than real time as the domain complexity increases, or cause transference problems—there is an unavoidable tradeoff between complexity and accuracy when simulation is used. EE avoids this.

- where a centralized, global coordinator for machine learning is not implementable or is unavailable, or where having parallelized embodied trials makes coordination of generational reproduction difficult.

  Scaling problems, e.g., in the form of communication bottlenecks (Martinoli et al., 1997), will arise for any centralized learning algorithm. EE does not depend on any centralized components.

- where the agents must learn "in the field."

  One can easily imagine applications where learning can only occur once the agents are deployed in the actual task domain, for example if the agents are deployed in a remote region (e.g., Mars or perhaps a Micro Electro-Mechanical Systems (MEMS) substrate). In such a case, a centralized coordinator of agent learning would be both difficult to design and perhaps precarious to use, as it would give the learning system a single point of failure. The distributed nature of EE provides a robust method for adaptation in the field.

- where we are concerned with interactive tasks (e.g., emergent group or team behaviors where we do not have *a priori* knowledge of group size and problem decomposition).

  In complex multi-agent domains, we are likely to not know how best to decompose the problem task into sub-tasks and will therefore require the discovery of a decomposition; note that multi-agent domains that require some form of *centralized task control* are not incompatible with a *distributed learning algorithm*. A minimally biased way to achieve behavioral or functional differentiation, in an evolutionary context, is to put reproductive behavior itself under evolutionary control such that speciation is made possible. EE uses multiple agents interacting in a shared environment and enables integration of reproductive behaviors with task behaviors.

- where the reproductive behavior itself is under adaptation.

  If the robots' choices for reproduction are expressed and determined by their behaviors, then a centralized reproductive algorithm that determines which robots reproduce with which other robots is excluded: reproductive and task-oriented behaviors are not categorically distinct, and a centralized mechanism would require an interpretive process to disentangle reproductive from non-reproductive behaviors, or else be reduced to a proxy for the robot's reproductive choices.

6 Caveats and Peculiar Issues

In spite of our enthusiasm for the embodied evolution approach, we must recognize that it is still a developing methodology, and although EE offers solutions for some issues, it also introduces new difficulties.

- An environment that contains a multitude of robots also includes some amount of robot-to-robot interference (Schneider-Fontán and Mataric, 1996); e.g., our phototaxis environment implicitly requires that each robot also successfully overcome interference. Hence we suggest that EE is more suitable where interaction is native to the task domain.

- Our experiments to date have concerned only simple tasks for which many other learning approaches have also been effective. Though we suggest EE is suited to multi-agent systems, we are only just now designing experiments that involve explicitly interactive tasks.

- Our powered-floor is constrained to research or industrial environments; other applications will require different power technologies.

- Because we eschew centralization, the job of monitoring our experiments and collecting data is made more awkward. With our current robotic hardware we are unable to monitor reproductive activity, for example.

- Because reproduction is based upon the principle of locality in our particular experimental setup, our implementation of EE is susceptible to failure if the agents become physically, and therefore reproductively, isolated. Moreover, the need to prevent reproductive isolation produces a selective pressure that may interfere destructively with the objective of the task. And if, as we suggest, the reproductive mechanism is modified to allow speciation, we can imagine that reproductive behaviors could become quite elaborate, worsening this interference.

- Though embodied evolution appears particularly suited to team tasks, the precise manner in which EE should be applied to team evolution is unclear. The requirement of locality seems to suggest that an awkward overlapping of multiple teams is needed for reproduction to take place, and the mechanism for organizing games between teams is problematic.

Thus we see that a great deal of research effort is still required to meet the long-term goals of EE, and although EE provides advantages in some domains it is not suitable for all applications.

7 Conclusions

We have introduced embodied evolution as a new methodology for evolutionary robotics. In good part, our motivation for the development of EE was simply to see if the artificial evolution of a population of robots could be implemented in the distributed and autonomous manner of natural evolution. But, EE also addresses several significant issues identified by ER researchers as problematic. Specifically, EE eliminates problems of simulation transference, alleviates the slow running time of numerous embodied trials, and provides robustness and scalability. EE also provides a promising substrate for the exploration of evolved group behaviors and provides interest for the Artificial Life community.

We must be careful to separate technological issues from fundamental ones when considering the long term prospects for evolutionary robotics. Our implementation of embodied evolution reveals some concerns, such as power, to be merely technological. Other issues, such as combinatorics and the limits of simulation, are fundamental in nature and can not be solved technologically. Nevertheless, some of these fundamental issues are ameliorated by the parallelism that EE employs.

We are currently running a suite of control experiments to better understand the evolutionary dynamics of the PGTA. These experiments will help us understand more precisely the parameters critical for the operation of EE. Our future work will address advanced robotic platforms, including evolvable hardware and evolvable morphology. In particular, our future task domains will emphasize the goal of evolving multi-agent systems.

Mataric and Cliff rightly point out that the utility of ER is ultimately determined by its ability to save human effort. Our continued investment in developing ER techniques, and the EE methodology in particular, is done with the belief that problem domains of interest will soon be, if they are not already, too difficult for the hand design of solutions; indeed, the seminal work in evolutionary robotics was done with this realization in mind (Husbands and Harvey, 1992). That evolutionary techniques have the potential to find unexpected, yet effective solutions has been made apparent to us several times during the course of our work on embodied evolution. The ability to discover novel, surprising solutions is the real promise of evolutionary techniques.

8 Acknowledgments

Our research would not have been possible without the valuable teamwork of the following people: Miguel Schneider-Fontán, Giovanni Motta, Prem Melville, Paaras Kumar, and members of DEMO especially Greg Hornby and Hod Lipson. We are also grateful for careful review and suggestions from Dave Cliff, and to Fred Martin of the MIT Media Lab for the supply of the Cricket micro-controllers. Finally, we thank the reviewers for their thorough comments.

References

AAIS (1998). *Continuous Power Supply for Khepera.* Applied AI Systems, Inc., Kanata, Ontario, Canada. Product literature.

Braitenberg, V. (1984). *Vehicles: experiments in synthetic psychology.* MIT Press.

Brooks, R. (1992). Artificial life and real robots. In Varela, F. and Bourgine, P., editors, *Proceedings of the First European Conference on Artificial Life*, pages 3–10. MIT Press.

Cliff, D., Harvey, I., and P., H. (1993). Explorations in evolutionary robotics. *Adaptive Behavior*, 2(1):73–110.

Floreano, D. and Mondada, F. (1994). Automatic creation of an autonomous agent: Genetic evolution of a neural-network driven robot. In Cliff, D., Husbands, P., Meyer, J.-A., and Wilson, S., editors, *From Animals to Animats 3*, pages 421–430. MIT Press.

Floreano, D. and Mondada, F. (1996). Evolution of homing navigation in a real mobile robot. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, 26(3):396–407.

Harvey, I. (1996). The microbial genetic algorithm. Submitted.

Harvey, I., Husbands, P., and Cliff, D. (1993). Issues in evolutionary robotics. In Meyer, J.-A., Roitblat, H., and Wilson, S., editors, *From Animals to Animats 2*, pages 364–373. MIT Press.

Husbands, P. and Harvey, I. (1992). Evolution versus design: Controlling autonomous robots. In *Proceedings of the Third Annual Conference on Artificial Intelligence, Simulation and Planning*, pages 139–146. IEEE Press.

Husbands, P., Harvey, I., and Cliff, D. (1992). Central issues in evolutionary robotics. Unpublished manuscript presented at ALife III, Sante Fe.

Jakobi, N. (1997a). Evolutionary robotics and the radical envelope of noise hypothesis. *Adaptive Behavior*, 6(2):325–368.

Jakobi, N. (1997b). Half-baked, ad hoc, and noisy: minimal simulations for evolutionary robotics. In Husbands, P. and Harvey, I., editors, *Fourth European Conference on Artificial Life*, pages 348–357. MIT Press.

Keating, D. (1998). Personal communication.

Martinoli, A., Franzi, E., and Matthey, O. (1997). Towards a reliable set-up for bio-inspired collective experiments with real robots. In Casals, A. and de Almeida, A., editors, *Proceedings of the Fifth Symposium on Experimental Robotics ISER-97*, pages 597–608. Springer Verlag.

Mataric, M. and Cliff, D. (1996). Challenges in evolving controllers for physical robots. *Robotics and Autonomous Systems, Special Issue on Evolutional Robotics*, 19(1):67–83.

Meyer, J.-A., Husbands, P., and Harvey, I. (1998). Evolutionary robotics: A survey of applications and problems. In Husbands, P. and Meyer, J.-A., editors, *Evolutionary Robotics: First European Workshop, EvoRobot98*, pages 1–21. Springer-Verlag.

Mondada, F. and Floreano, D. (1996). Evolution and mobile autonomous robotics. In Sanchez, E. and Tommasini, M., editors, *Towards Evolvable Hardware*, pages 221–249. Springer Verlag, Berlin.

Nolfi, S. (1997). Evolving non-trivial behaviors on real robots: A garbage collecting robot. *Robotics and Autonomous Systems*, 22(3–4):187–198.

Resnick, M., Berg, R., Eisenberg, M., and Turkle, S. (1997). Beyond black boxes: Bringing transparency and aesthetics back to scientific instruments. MIT project funded by the National Science Foundation (1997-1999).

Schneider-Fontán, M. and Mataric, M. (1996). A study of territoriality: The role of critical mass in adaptive task division. In Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., and Wilson, S., editors, *From Animals to Animats IV*, pages 553–561. MIT Press.

Sims, K. (1994). Evolving 3d morphology and behavior by competition. In *Artificial Life IV*, pages 28–39. MIT Press.

Watson, R. A., Ficici, S. G., and Pollack, J. B. (1999). Embodied evolution: Embodying an evolutionary algorithm in a population of robots. In Angeline, P., Michalewicz, Z., Schoenauer, M., Yao, X., and Zalzala, A., editors, *1999 Congress on Evolutionary Computation*, pages 335–342. IEEE Press.