God Save the Red Queen!
Competition in Co-Evolutionary Robotics

Dario Floreano
Center for Neuro-Mimetic Systems
Laboratory of Microcomputing
Swiss Federal Institute of Technology
EPFL, Lausanne, Switzerland
floreano@di.epfl.ch
diwww.epfl.ch/lami/team/floreano/

Stefano Nolfi
Department of Neural Systems
and Artificial Life
National Research Council
Viale Marx 15, Roma, Italy
stefano@kant.irmkant.rm.cnr.it
kant.irmkant.rm.cnr.it/nolfi.html

ABSTRACT

In the simplest scenario of two co-evolving populations in competition with each other, fitness progress is achieved at disadvantage of the other population's fitness. The ever-changing fitness landscape caused by the competing species (named the "Red Queen effect") makes the system dynamics more complex, but it also provides a set of advantages with respect to single-population evolution. Here we present results from an experiment with two mobile robots, a predator equipped with vision and a much faster prey with simpler sensors. Without any effort in fitness design, a set of interesting behaviors emerged in relatively short time, such as obstacle avoidance, straight navigation, visual tracking, object discrimination (robot vs. wall), object following, and others. Although such experiments cannot yet be performed in real-time on populations of robots for technical reasons, the approach seems promising.

1 Competitive Co-Evolution

Competitive co-evolution has recently attracted considerable interest in the community of Artificial Life and Evolutionary Computation. In the simplest scenario of two co-evolving populations, fitness progress is achieved at disadvantage of the other population's fitness. Although it is easy to point out several examples of such situation in nature (e.g., competition for limited food resources, host-parasite, predator-prey), it is more difficult to analyze and understand the importance and long-term effects of such "arms races" on the development of specific genetic traits and behaviors. An interesting complication is given by the "Red Queen effect"[1] whereby the fitness landscape of each population is continuously changed by the competing population. Given the relative lack of empirical evidence for the importance of the Red Queen effect on biological evolution, Artificial Life techniques seem well-suited to study this penomenon (Cliff and Miller, 1995). For example, Ray's "Tierra system" (1991) and Sims' creatures (1994) are based on co-evolutionary competing species; also, several other simulated eco-worlds make use of co-evolving species and competitive fitness schemes (Menczer and Belew, 1993; Yeager, 1994). Some researchers have attempted to provide a theoretical understanding of the underlying complex dynamics; notably among others, Axelrod (1989) in the context of the Iterated Prisoner's Dilemma, Renshaw (1991) by modeling spatially distributed populations, and Kauffman (1992) in the extended framework of his "NKC" class of statistical models of rugged fitness landscape .

From a computational perspective, competing co-evolutionary systems are appealing because the ever-changing fitness landscape, caused by the struggle of each species to take profit of the competitors' weaknesses, could be potentially exploited to prevent stagnation in local maxima. Hillis (1990) reported a significative improvement in the evolution of sorting programs when parasites (programs deciding the test conditions for the sorting programs) were co-evolved , and similar results were found by Angeline and Pollack (1993) on co-evolution of players for the Tic Tac Toe game. Koza

---

[1] The Red Queen is a figure, invented by novelist Lewis Carroll, who was always running without making any advancement because the landscape was moving with her.

(1991, 1992) applied Genetic Programming to the evolution of pursuer-evader behaviors and Reynolds (1994) observed in a similar scenario that co-evolving populations of pursuers and evaders display increasingly better strategies. Cliff and Miller realised the potentiality of co-evolution of pursuit-evasion tactics in evolutionary robotics. In the first of a series of papers (Miller and Cliff, 1994), they provided an extensive review of the literature in biology and in differential game theory and introduced their 2D simulation of simple robots with "eyes". Later, they proposed a new set of performance and genetic measures in order to describe evolutionary progress which could not be otherwise tracked down due to the Red Queen effect (Cliff and Miller, 1995). Recently, they described some of the results where simulated robots with evolved eye-morphologies could either evade or pursue their competitors of several generations earlier and proposed some applications of the approach in biology and in the entertainment industry (Cliff and Miller, 1996).

1.1 Prospects for the Red Queen in Robotics

Despite the promising achievements described above, if one carefully looks at the results described in the literature focusing on competitive co-evolution of pursuit-evasion behaviors, it is easy to notice that co-evolutionary benefits often come at the cost of several thousand individuals per population (Reynolds, 1994), several hundred generations (Cliff and Miller, 1996), or repeated trials of evolutionary runs with alternating success (Sims, 1994). Moreover, since all the experiments have been conducted in simulation, often the results cannot be directly applied to real robots, either because agent descriptions are too abstract or technically unfeasible, or because the fitness function takes into account global information (such as the distance between the competing agents). All these facts seem to greatly limit any prospect of exploiting the Red Queen effect for evolution of robotic controllers in the real world and for engineering purposes. The focus of this paper is an investigation of the feasibility of this approach in more realistic conditions for evolutionary robotics.

We have been attracted to competing co-evolutionary robotics for two main reasons: the prospect of a methodology that does not require specification of complex fitness functions to evolve efficient behaviors and the intrinsic complex dynamics of an elegantly-simple setup. Definition of a fitness function in evolutionary robotics is often a painstaking process (Mataric and Cliff, 1996) which impairs automaticity of controller development and reduces behavioral autonomy of the robot. Complications arise from the tendency of genetic algorithms to exploit minimalist solutions which often do not match the expectations of the experimenter; moreover, if one wishes to conduct the evolutionary run entirely on the robot (or also in the case when only the final phase of evolution is carried out on the real robot (Nolfi et al., 1994)), the fitness function cannot be based on variables other than those directly available in real time to the robotic platform. Although some of us have previously attempted to devise ways of reducing effort in fitness design (Floreano and Mondada, 1996a), those results cannot yet be considered fully satisfactory with respect to this issue. On the other hand, co-evolutionary pursuit-evasion provides a rich set of complex dynamics which are well-suited to study other interesting issues on which we have indulged in our recent research on evolution of neurocontrollers, such as rapid adaptation during life (Floreano and Mondada, 1996b), various schemes of genetic representation (Nolfi and Parisi, 1995; Calabretta et al., 1996), and different network architectures (Nolfi, 1997). However, in this paper we will leave out these latter issues (which are currently being evaluated), and focus on the first set of results.

The main goal of the experiments described here consists in studying the feasibility of co-evolutionary pursuit-evasion for evolving useful neurocontrollers for two Khepera robots in a simple but realistic scenario.

2 Method

We decided to study pursuit-evasion as a metaphor for predator-prey, this being a quite common and suggestive situation in nature. As often happens, predators and preys belong to different species with different sensory and motor characteristics. Thus, we employed two Khepera robots, one of which (the *Predator*) was equipped with a vision module while the other (the *Prey*) had a maximum available speed set to twice that of the predator (Figure 1). Both individuals were also provided with eight infrared proximity sensors (six on the front side and two on the back) which had a maximum detection range of 3 cm in our environment. The two species would evolve in a square arena of size 47 x 47 cm with high white walls so that the predator could always see the prey (if within the visual angle) as a black spot on a white background.

Running co-evolutionary experiments with two or more robots within the same environment causes problems with the cables that connect the robots to the workstation for power supply and information exchange (see (Floreano and Mondada, 1994) for detailed description of this methodology). Therefore, we decided to resort to a particular type of simulation developed and extensively tested on Khepera by some of us: Instead of employing a mathematical model of the sensors and motors (which inevitably causes problems when the evolved controllers are downloaded to the physical robot due to non-uniform noise and different response types of each sensor), it has

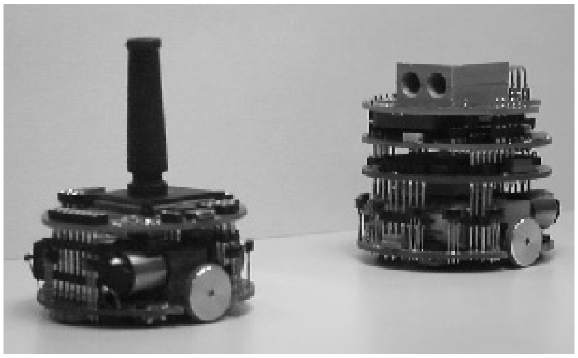

Figure 1. Right: The Predator is equipped with the vision module (1D-array of photoreceptors, visual angle of 36°). Left: The Prey has a black protuberance which can be detected by the predator everywhere in the environment, but its maximum speed is twice that of the predator. Both Predator and Prey are equipped with 8 infrared proximity sensors (max detection range was 3 cm in our environment).

been shown that for geometrically-simple environments one can reduce discrepancies between behaviors in simulation and on the real robot by sampling sensor activity at different distances and angles of the robot from the objects of the world (see (Miglino et al., 1996) for details). We have thus employed the same methodology and sampled infrared sensor activity of each robot in front of a wall and in front of another robot. These values were then separately stored away and accessed through a look-up table depending on the faced object. Displacement of the robots was computed by passing to the simulator a vector of wheel velocities (positive and negative values standing for motion in opposite directions) and calculating the new x, y position as follows

$$y_{t+1} = \left(\frac{L}{2}\frac{V_L + V_R}{V_L - V_R}\right)\sin\left(\frac{V_L - V_R}{L}t\right) \quad (1)$$

$$x_{t+1} = \left[\frac{L}{2}\frac{V_L + V_R}{V_L - V_R}\right]\left[1 - \cos\left(\frac{V_L - V_R}{L}t\right)\right] \quad (2)$$

where V_L, V_R are the velocities applied to the left and right wheel respectively, L is the inter-wheel distance, and t is the amount of time for which the set of wheel speed is maintained. Given the characteristics of the PID controller and physical friction and slippage, the maximum deviation for each wheel at a speed of $100mm/s$ is $\pm 1\%$ with standard deviation $\sigma = 0.5$ (Maechler, 1997). In our simulations we have set the maximum wheel speed in each direction to $80mm/s$ for the predator and to $160mm/s$ for the prey. Since wheel velocities were updated very often ($t = 100ms$), the maximum error that we could expect was $0.008mm$ for the predator and $0.016mm$ for the prey, absolutely negligible values with respect to the sensor characteristics. As a conservative measure, if the simulated robots attempted to rotate against an obstacle, their position was not updated (this solution was already successfully employed by Miglino *et al.* (1996)).

Simulation of the visual input required different considerations. The vision module K213 of Khepera is an additional turret which can be plugged-in directly on top of the basic platform. It consists of a 1D-array of 64 photoreceptors which provide a linear image composed of 64 pixels of 256 gray-levels each, subtending a view-angle of 36°. The optics are designed to bring into focus objects situated at distances between 5cm and 50cm while an additional sensor of light intensity automatically adapts the scanning speed of the chip to keep the image stable and exploit at best the sensitivity of receptors under a large variety of illumination intensities. The K213 vision turret incorporates a private 68HC11 processor which is used for optional low-level processing of the scanned image before passing it to the robot controller. One of these options is detection of the position in the image corresponding to the pixel with minimal intensity (in this case, only one byte of information is transmitted). Therefore, instead of simulating the response of the 1D-array of receptors resorting to complex and time-consuming ray-tracing techniques, we exploited the built-in facility for position detection of the pixel with minimal intensity and divided the visual angle in five sectors corresponding to five simulated photoreceptors (Figure 2, left). If the pixel with minimal intensity was within the first sector, then the first simulated photoreceptor would become active; if the pixel was within the second sector, then the second photoreceptor would become active; etc. We made sure in a set of preliminary measurements that this type of input reduction was largely sufficient to reliably capture and represent all the relevant visual information available to the predator.

In line with some of our previous work (Floreano and Mondada, 1994), the robot controller was a simple perceptron with two sigmoid units and recurrent connections at the output layer. The activation of each output unit was used to update the speed value of the corre-

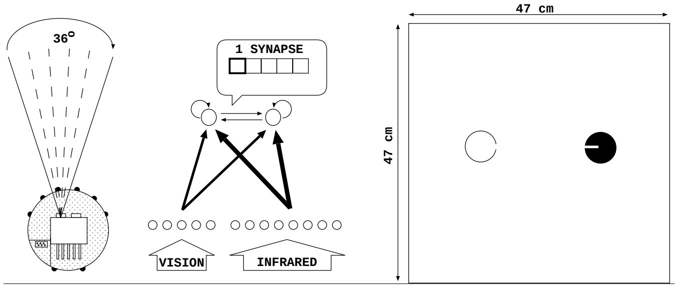

Figure 2. Left and center: Details of simulation of vision, of neural network architecture, and of genetic encoding. The prey differs from the predator in that it does not have 5 input units for vision. Each synapse in the network is coded by five bits, the first bit determining the sign of the synapse and the remaining four its strength. Right: Initial starting position for Prey (left, empty disk with small opening corresponding to frontal direction) and Predator (right, black disk with line corresponding to frontal direction) in the arena. For each competition, the initial orientation is random.

sponding wheel every 100ms. In the case of the predator, each output unit received connections from five photoreceptors and from eight infrared proximity sensors (Figure 2, center); in the case of the prey, each output unit received input only from 8 infrared proximity sensors, but its activation value was multiplied by 2 before setting the wheel speed. This structure, which is well-suited for evolution of Braitenberg-like obstacle avoidance (because of its symmetrical architecture and direct input-output connections), was chosen for being a minimally sufficient architecture to evolve something interesting while maintaining system complexity at a manageable level; for the same reason, the architecture was kept fixed, and only synaptic strengths and output unit threshold values were evolved.

In order to keep things as simple as possible and given the small size of the parameter set, we used direct genetic encoding (Yao, 1993): each parameter (including recurrent connections and threshold values of output units) was encoded on five bits, the first bit determining the sign of the synapse and the remaining four its strength. Therefore, the genotype of the predator was 5 x (30 synapses + 2 thresholds) bits long while that of the prey was 5 x (20 synapses + 2 thresholds) bits long. Two populations of 100 individuals each were co-evolved for 100 generations. Each individual was tested against the best competitors of the ten previous generations (a similar procedure was used in (Sims, 1994; Reynolds, 1994; Cliff and Miller, 1995)) in order to improve co-evolutionary stability. At generation 0, competitors were randomly chosen whithin the same generation, whereas in the other 9 initial generations they were randomly chosen from the pool of available best individuals (2 at generation 3, 3 at generation 4, etc.). For each competition, the prey and predator were always positioned on a horizontal line in the middle of the environment at a distance corresponding to half the environment width (Figure 2, right), but always at a new random orientation. The competition ended either when the predator touched the prey or after 500 motor updates (corresponding to 50 seconds at maximum on the physical robot). The fitness function Φ_c for each competition c did not require any sensor or motor measurement, nor any global position measure; it was simply $TimetoContact$ normalized by the maximum number of motor updates TtC for the predator pr, and $1 - TtC$ for the prey py, further averaged over the number of competitions. Therefore the fitness values were always between 0 and 1, where 0 means worst. Individuals were ranked after fitness performance in descending order and the best 20 were allowed to reproduce. One-point crossover was applied on all randomly paired strings with probability pc, and random mutation (bit switching) was applied to each bit with constant probability $pm = 0.05$.

3 Results and Analyses

We performed 6 evolutionary runs of 100 generations each, three of them with $pc = 0.1$ and three with $pc = 0.9$. We did not attempt to optimize either the number of individuals per population or other genetic operators. Each run took approximately 10 hours on a Sun SparcStation 20, even though this time could be reduced by optimizing the computer code. We did not notice significative differences between sets of runs with different crossover probability and among single runs within each set (with respect to all the measures and analyses reported here). Figure 3, left side, shows the average population fitness for one run with $pc = 0.1$. As in (Sims, 1994, p. 36), a set of oscillations in fitness emerge after an initial short period. Since these oscillations take place at different frequencies and start at different points in different evolutionary runs, averaged measures over multiple runs would not convey any information. Therefore, for clarity of explanation, we give data for a single run. As compared to (Sims, 1994, p. 36), we never observed

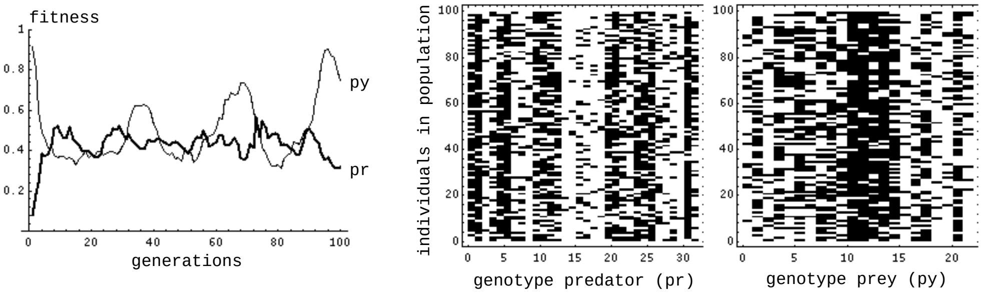

Figure 3. Left: Average population fitness across generations for predator (pr) and prey (py). Data points from a single run. Right: Genetic bitmap of the two final populations. Only the first bit per synapse, which determines the sign of the synapse, is plotted: black means a negative synapse, white a positive synapse. Each line correspond to the genotype of one individual in the population. The first half of each genotype corresponds to the synapses for the left motor neuron, the remaining half to the synapses for the right motor neuron. The genotype for the predator is longer (that is, squares are smaller) because it includes visual inputs too.

dominance of one population against the other in any of our evolutionary runs. However, the fitness for the prey always tended to generate higher peaks due to initial position advantage (even in the case of the worst prey and best predator, the latter will always take some time to reach the prey from its starting position). Hundred generations were always sufficient to evolve the full range of pursuit-evasion behaviors which will be described below.

After 100 generations, each population displays a good degree of convergence (also when $pc = 0.9$), as one can see from the bitmap of the genotypes of all individuals of the last population displayed on the right side of Figure 3. Genetic maps show only the value of the first bit of each synapse which determines the sign of the synapse and thus gives a rough idea of the underlying neural structure. This structure, which is symmetric around its center, indicates that most of the individuals in both populations are a variation of a Braitenberg vehicle (equipped with vision for the predator); more details and analyses will be given elsewhere. However, average fitness measures do not tell us much of what happens at the level of individual tournaments, especially when the populations have not yet converged.

3.1 Individual Tournaments

An individual tournament is one or more competitions between a single individual and one or more of his competitors. Individual tournaments tell us what happens at the microlevel and provide useful clues about the dynamics of competitive co-evolution. However, performance oscillations caused by the Red Queen effect make it difficult to visualize real changes and progress in both populations. Cliff and Miller have developed an interesting set of measuring techniques for individual tournaments in competing co-evolutionary systems (Cliff and Miller, 1995), which we will apply below. The fitness plot of the best individual at each generation against the best competitors of the ten preceding generations (Figure 4, left center) shows a set of oscillations which are approximately in opposite phase. The development of a better strategy by one of the species corresponds to a decrement in performance of the competing species. Major changes in behavioral strategies are reflected at the genetic level. On the top and bottom left of Figure 4 one can see – for the predator and prey, respectively– the Hamming distance between the genotype of the best individuals for each generation. Every square in the matrix represents the normalized Hamming distance between two individuals; the darker a square is, the more different the two genotypes are.The white diagonal line (Hamming distance zero) is the identity comparison. Small white areas, which indicate almost identical genotypes, correspond to periods of similar fitness in the performance graph. Since these areas appear in the neighborhood of the matrix diagonal, only best individuals which are close in evolutionary time have similar genotypes. It also means that individuals that report similar fitness, but are distant in time, such as preys of generation 68 and 98, do not have the same genotype. Small dark lines between white zones indicate an abrupt change of behavioral strategy, whereas a gradual fading of white into gray indicates gradual genetic change (which is quite rare here). But, does genetic diversity among the best distant individuals imply real progress in the behavioral strategies? Cliff and Miller (Cliff and Miller, 1995) have devised a way of monitoring fitness performance by testing the performance of the best individual against the best competing ancestors, which they call CIAO data (Current Individual vs. Ancestral Opponents). In applying this technique to our populations, each individual is tested ten times against each best competing ancestor and the average fitness is plotted as darker squares for higher values (Figure 4, right). CIAO graphs (for

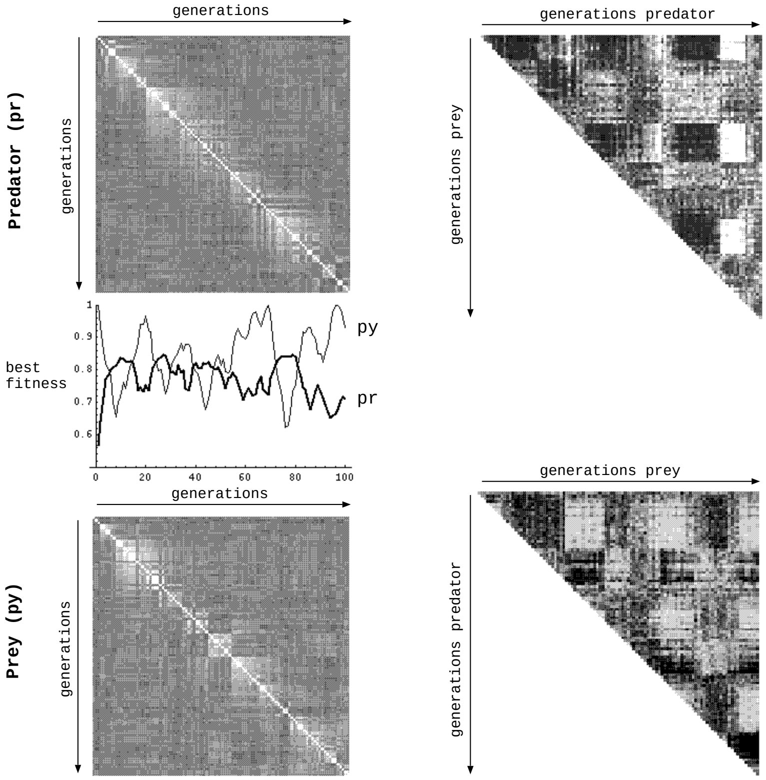

Figure 4. Left, center: Fitness of best individual across generations for predator (pr) and prey (py) (smoothed using rolling average over three data points). Left, top and bottom: Hamming distance between each best individual and all the other best individuals across generations for predator and prey. Black means that all genes between the two individuals are different, white means that the two individuals are identical. Right, top and bottom: CIAO graphs: Average fitness over ten tournaments between the best individual and each of the best competitors of the preceding generations, respectively for prey and predator; more informations on this measuring technique is available on the paper by Cliff and Miller (1995). Black is best, white is worst. Slight asymmetries between the two graphs are due to random initial orientations for each tournament.

predator and prey) show two interesting facts. The first is that individuals of later generation, although they report a high fitness during evolutionary training, do not necessarily score well against competitors of much earlier generations. For example, the best predators of generations 85-95 can hardly catch best preys of generations 0-10, 35-50, and 70-80. This indicates that during generations 85-95, predators developed a behavioral strategy that was tuned to their preceding ten best competitors (let us remind that during co-evolution each individual is tested against the best competitor of the preceding ten generations). Furthermore, the Scottish tartan patterns of these CIAO graphs indicate periods of relative stasis interrupted by short and radical changes of behavior, in accordance with the Hamming graphs on the left of figure 4.

3.2 Master Tournament

The analyses described above have revealed much about the dynamics of our competing co-evolutionary system. For example, for optimization purposes, we know that individuals of the final generations are not necessarily the best individuals of the evolutionary run. The simplest way to know more about our predators and preys, is to organize a Master Tournament where each best individual is tested ten times against each best competitor of all generations. Top of Figure 5 shows the master fitness for each best individual across generations (fitness values are averaged over ten competitions and over hundred tournaments). A Master Tournament tells us two things: At which generation we can find the best prey and the best predator, and at which generation we are guaranteed to observe the most interesting tournaments. The first aspect is important for optimization purposes and applications, the latter for pure entertainment. The best individuals are those reporting the high-

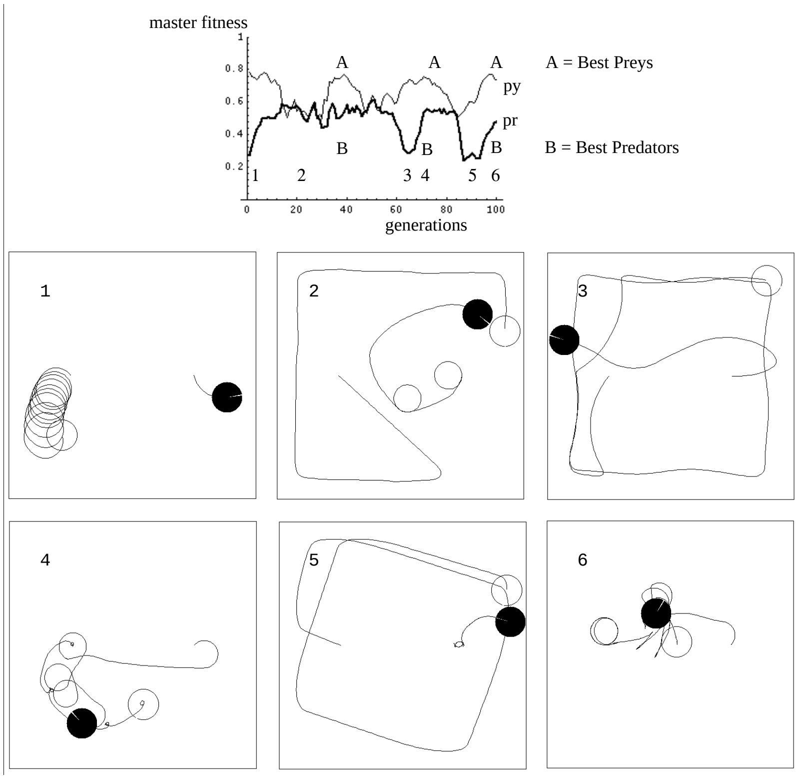

Figure 5. Top: Fitness of best individuals in Master Tournament. Letters indicate position of best preys and best predators. Numbers indicate position of individuals whose competitions are displayed below. Bottom: Behaviors recorded at interesting points of co-evolution, representing typical strategies. Black disk is predator, white is the prey.

est fitness when also the competitor reports the highest fitness (marked by letters A and B in the graph). Instead, the most entertaining tournaments are those that take place between individuals that report the same fitness level, because these are the situations where both species have the same level of ability to win over the competitor. The Master Tournament also shows that some increment/decrement of fitness in the individual tournaments (Figure 4, left center), such as at generation 45 and generation 75, are due to slight opportunistic adjustements between the competitors which do not reflect their overall performance. In the lower part of Figure 5, behaviors of best competitors at critical stages of co-evolution, as indicated by Master Tournament data, give a more intuitive idea of how pursuit-evasion strategies are co-evolved. Initially, the predator tends to stop in front of walls while the prey moves in circles (box 1). Later, the prey moves fast at straight trajectories avoiding walls while the predator tracks it from the center and quickly attacks when the prey is closer (box 2). Interestingly, predators develop the ability to know how distant the preys are by using information on how fast their target moves in the visual field. Decrement of predator performance around generation 65 is due to a temporary loss of the ability to discriminate between walls and preys. As shown in box 3, the predator intercepts the prey, but it misses it crashing against the wall. Around generation 75, we have a typical example of the best prey (box 4); it moves in circles and, when the predator gets closer, it rapidly avoids it. This is quite interesting. Indeed, preys that move too fast around the environment sometimes cannot avoid an approaching predator because they detect it too late (IR sensors have lower sensitivity for a small cylindrical object, like another robot, than for a white flat wall). Therefore, it pays off to wait for the slower predator and accurately avoid it. However, some predators become smart enough to perform a small circle once they have missed the target, and re-attack until, by chance, the prey displays a side without IR sensors. As soon as the preys begin again moving around the environment, the predator develops a "spider strategy" (box 5): it slowly backs until it finds a wall where it waits for the fast-approaching prey. However, this strategy does not pay off when the preys stay in the same place (as indicated by the decrement in master fitness and by the white patterns in the CIAO graphs of Figure 4). Finally, at generation 99 we have a new interesting strategy (box 6): the predator quickly tracks and reaches the prey which quietly rotates in small circles. As soon as the prey senses the predator, it backs and then approaches the predator (without touching it) on the side where it cannot be seen; consequently, the predator quickly turns in the attempt to visualize the prey which rotates around it, producing an entertaining dance.

4 Conclusion

Our work has shown that competing co-evolutionary systems can be of great interest in evolutionary robotics. In the case described here, we have observed spontaneous evolution og obstacle avoidance, visual tracking, object discrimination (prey vs. wall), following, and a variety of other behaviors without any effort in fitness design (as compared to single population experiments (Floreano and Mondada, 1996a)). Similarly, evolutionary time and resources were relatively small and manageable. From an optimization point of view, the drawback due to cycling between alternative behavioral strategies could be easily overcome by devising a strategy for picking out the most appropriate individuals (Master Tournament) for one's own purposes (being it efficiency or fun).

At the same time, we feel that competing co-evolutionary systems are an excellent testbed for studying several important issues in evolutionary computation. One of these is the encoding strategy. Here we have used a very simple strategy that, although perfectly sufficient for evolving interesting behaviors in this particular setup, does not guarantee continuous progress in more complex situations. The motivation for our choice was the effort to keep the system simple and analyzable (further analysis is being currently carried out on a set of similar experiments). On the other hand, if co-evolution between competing species is indeed an important engine of evolutionary progress in nature, a genetic encoding that better cares of already achieved progress, is mandatory. For example, Cliff and Miller (Cliff and Miller, 1995) have used a much more complex encoding strategy which seems to provide continuous evolutionary progress, even though at the cost of much longer time and computational resources. Another interesting issue is learning during life. The situation described here seems to be one which would reward systems capable of some form of quick adaptation during life, and it provides the necessary unpredictable dynamics for testing some of the learning schemes available in the neural network literature.

Although before conducing experiments in real time on more complex robots one should devise a solution for power supply, there seem to be good prospects for the Red Queen in evolutionary robotics.

Acknowledgments

D.F. thanks Philip Mächler for useful discussions on simulating the motor system of Khepera, Francesco Mondada for providing the Khepera robots and the vision module, and LAMI-EPFL for generously supporting the expenses related to presentation of this work at the Genetic Programming conference at Stanford University.

Bibliography

Angeline, P. J. and Pollack, J. B. 1993. Competitive environments evolve better solutions for complex tasks. In Forrest, S., editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 264–270, San Mateo, CA. Morgan Kaufmann.

AxelRod, R. 1989. Evolution of strategies in the iterated prisoner's dilemma. In Davis, L., editor, *Genetic Algorithms and Simulated Annealing*. Morgan Kaufmann, San Mateo, CA.

Calabretta, R., Galbiati, R., Nolfi, S., and Parisi, D. 1996. Two is better than one: A diploid genotype for neural networks. *Neural Processing Letters*, In press.

Cliff, D. and Miller, G. F. 1995. Tracking the red queen: Measurements of adaptive progress in co-evolutionary simulations. In Morán, F., Moreno, A., Merelo, J. J., and Chacón, P., editors, *Advances in Artificial Life: Proceedings of the Third European Conference on Artificial Life*, pages 200–218. Springer Verlag, Berlin.

Cliff, D. and Miller, G. F. 1996. Co-evolution of Pursuit and Evasion II: Simulation Methods and Results. In Maes, P., Mataric, M., Meyer, J., Pollack, J., Roitblat, H., and Wilson, S., editors, *From Animals to Animats IV: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*. MIT Press-Bradford Books, Cambridge, MA.

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., and Wilson, S. W., editors, *From Animals to Animats III: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*. MIT Press-Bradford Books, Cambridge, MA.

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

Floreano, D. and Mondada, F. 1996b. Evolution of plastic neurocontrollers for situated agents. In Maes, P., Mataric, M., Meyer, J., Pollack, J., Roitblat, H., and Wilson, S., editors, *From Animals to Animats IV: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*. MIT Press-Bradford Books, Cambridge, MA.

Hillis, W. 1990. Co-evolving parasites improve simulated evolution as an optimization procedure. *Physica D*, 42:228–234.

Kauffman, S. A. and Johnsen, S. 1992. Co-evolution to the edge of chaos: Coupled fitness landscapes, poised states, and co-evolutionary avalanches. In Langton, C., Farmer, J., Rasmussen, S., and Taylor, C., editors, *Artificial Life II: Proceedings Volume of Santa Fe Conference*, volume XI. Addison Wesley: series of the Santa Fe Institute Studies in the Sciences of Complexities, Redwood City, CA.

Koza, J. R. 1991. Evolution and co-evolution of computer programs to control independently-acting agents. In Meyer, J. and Wilson, S., editors, *From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior*. MIT Press, Cambridge, MA.

Koza, J. R. 1992. *Genetic programming: On the programming of computers by means of natural selection*. MIT Press, Cambridge, MA.

Maechler, P. 1997. Robot odometry correction using grid lines on the floor. In *Proceedings of 2nd International Workshop on Mechatronical Computer Systems for perception and Action*, Pisa, Italy.

Mataric, M. and Cliff, D. 1996. Challenges in Evolving Controllers for Physical Robots. *Robotics and Autonomous Systems*. In press.

Menczer, F. and Belew, R. K. 1993. Latent energy environments. In Belew, R. K. and Mitchell, S., editors, *Plastic Individuals in Evolving Populations*. Addison Wesley, Redwood City, CA.

Miglino, O., Lund, H. H., and Nolfi, S. 1996. Evolving Mobile Robots in Simulated and Real Environments. *Artificial Life*, 2:417–434.

Miller, G. F. and Cliff, D. 1994. Protean behavior in dynamic games: Arguments for the co-evolution of pursuit-evasion tactics. In Cliff, D., Husbands, P., Meyer, J., and Wilson, S. W., editors, *From Animals to Animats III: Proceedings of the Third International Conference on Simulation of Adaptive Behavior*. MIT Press-Bradford Books, Cambridge, MA.

Nolfi, S. 1997. Using emergent modularity to develop control system for mobile robots. *Adaptive Behavior*, 5:in press.

Nolfi, S., Floreano, D., Miglino, O., and Mondada, F. 1994. How to evolve autonomous robots: Different approaches in evolutionary robotics. In Brooks, R. and Maes, P., editors, *Proceedings of the Fourth Workshop on Artificial Life*, pages 190–197, Boston, MA. MIT Press.

Nolfi, S. and Parisi, D. 1995. Genotypes for neural networks. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*. MIT Press, Cambridge, MA.

Ray, T. S. 1992. An approach to the synthesis of life. In Langton, C., Farmer, J., Rasmussen, S., and Taylor, C., editors, *Artificial Life II: Proceedings Volume of Santa Fe Conference*, volume XI. Addison Wesley: series of the Santa Fe Institute Studies in the Sciences of Complexities, Redwood City, CA.

Renshaw, E. 1991. *Modeling Biological Populations in Space and Time*. Cambridge University Press, Cambridge.

Reynolds, C. W. 1994. Competition, Coevolution and the Game of Tag. In Brooks, R. and Maes, P., editors, *Proceedings of the Fourth Workshop on Artificial Life*, pages 59–69, Boston, MA. MIT Press.

Sims, K. 1994. Evolving 3D Morphology and Behavior by Competition. In Brooks, R. and Maes, P., editors, *Proceedings of the Fourth Workshop on Artificial Life*, pages 28–39, Boston, MA. MIT Press.

Yao, X. 1993. A review of evolutionary artificial neural networks. *International Journal of Intelligent Systems*, 4:203–222.

Yeager, L. 1994. Computational Genetics, Physiology, Metabolism, Neural Systems, Learning, Vision, and Behavior or PolyWorld: Life in a New Context. In Langton, C., editor, *Artificial Life III*. Addison-Wesley, Redwood City, CA.