Competitive Co-Evolutionary Robotics: From Theory to Practice

Dario Floreano[*]**, Stefano Nolfi**[**]**, Francesco Mondada**[*]

[*] Laboratory of Microcomputing
Swiss Federal Institute of Technology, CH-1015 Lausanne
{Name.Surname}@epfl.ch

[**] Department of Neural Systems and Artificial Life
National Research Council, Viale Marx 15, Roma, Italy
stefano@kant.irmkant.rm.cnr.it

Abstract

It is argued that competitive co-evolution is a viable methodology for developing truly autonomous and intelligent machines capable of setting their own goals in order to face new and continuously changing challenges. The paper starts giving an introduction to the dynamics of competitive co-evolutionary systems and reviews their relevance from a computational perspective. The method is then applied to two mobile robots, a predator and a prey, which quickly and autonomously develop efficient chase and evasion strategies. The results are then explained and put in a long-term framework resorting to a visualization of the Red Queen effect on the fitness landscape. Finally, comparative data on different selection criteria are used to indicate that co-evolution does not optimize "intuitive" objective criteria.

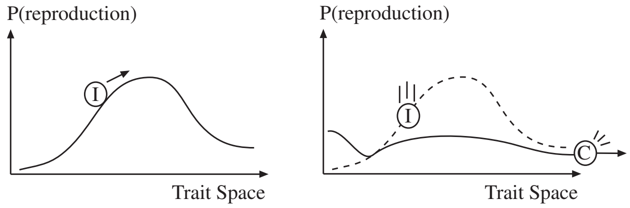


Figure 1 **Left**: Reproduction probability of a single species I under evolution in a static environment. Evolution drives the organisms towards zones (combinations of traits) corresponding to higher reproduction success. **Right**: Reproduction probability of species I under competitive co-evolution. The reproductive value (fitness) of certain trait combinations can be affected by adaptive changes in the competing species C, resulting in a continuous modification of the evolutionary surface. This phenomenon is often called the *Red Queen Effect* (van Valen, 1973).

1. Competitive Co-Evolution

In a competitive co-evolutionary system the survival probability of a species is affected by the behavior of the other species. In the simplest scenario of only two competing species, such as a predator and a prey or a parasite and a host, the performance of each individual is tightly related to the performance of the competitor both on the ontogenetic and on the evolutionary time scale. Changes in one lineage might affect the selection pressure on the other lineage and, if the other lineage also responds with counter-adaptive features, the system might give rise to what biologists call a "a co-evolutionary arms race" (Dawkins and Krebs, 1979).

Within the framework of *artificial evolution*, the reciprocal effects of changes by one species on the selection pressure of the other species introduce novel complexities with respect to the case of a single species evolved in a static environment. In the latter case (figure 1, left), there is a unique relationship between the traits of organisms and its reproduction success, and evolution is often seen as a force driving the population towards combinations of traits that maximize reproduction success (Goldberg, 1989). Instead, in competitive co-evolution the reproduction probability of an organism with certain traits can be modified by the competitors; that is, changes in one species affect the reproductive value of specific trait combinations in the other species (figure 1, right). It might thus happen that progress achieved by one lineage is reduced or eliminated by the competing species. This phenomenon is sometimes referred to as the "Red Queen Effect" (van Valen, 1973) (from the imaginary chess figure, invented by novelist Lewis Carroll, who was always running without making any advancement because the landscape was moving with her).

Under certain symmetry conditions, a competitive co-evolutionary system can display oscillatory dynamics whose outcome and stability are difficult to predict. Initial formal studies of predator-prey dynamics date back to the mid-20's when Lotka (1925) and Volterra (1926) independently developed a simple model composed of two differential equations based on the assumptions described above. The Lotka-Volterra model describes how the population densities N_1 and N_2 of two co-evolving competing species vary in evolutionary time t

$$\frac{dN_1}{dt} = N_1 (r_1 - b_1 N_2), \quad \frac{dN_2}{dt} = N_2 (-r_2 + b_2 N_1),$$

where r_1 is the increment rate of the prey population in the absence of predators, r_2 is the death rate of predators in the absence of prey, b_1 is the death rate of prey due to be eaten by predators, and b_2 is the ability of the predator in catching the prey. Probabilistic variations of this simple model have been used to explain several biological observations, such as the oscillatory dynamics of co-evolving host-parasite populations reported in figure 2 (Utida, 1957), and predict oscillation periods and stability conditions (Renshaw, 1991; Murray, 1993). However, it should be noticed that this model attempts to predict variations in population size assuming fixed performance for both predators and prey across generations.

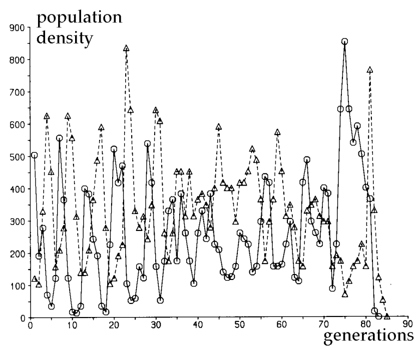


Figure 2 Population density of two biological co-evolving and competing species across generations. △=parasite; ○=host. Termination at generation 85 was caused by external factors. Adapted from (Utida, 1957).

1.1 Computational attractions

From a computational perspective, it would be interesting to know *whether competitive co-evolution might promote performance progress.* In other words, are predators of later generations better at catching prey of previous generations than predators belonging to the same generation of those prey (and viceversa)? Since data on population densities cannot be readily used as indicators of performance, theoretical models and biological observations do not tell us whether competitive co-evolution might be a powerful generator of increasingly better individuals. One possibility is that, since each species is selected only against competitors belonging to the same evolutionary period, there would be no selective pressure to evolve increasingly better performances across generations in either party which would rather display a cyclic alternation of the same strategies every n generations. Another possibility is that the ever-changing fitness landscape, caused by the struggle of each species to take profit of the competitors' weaknesses, might prevent stagnation of the two populations 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. More recently, Rosin and Belew (1997) compared various co-evolutionary strategies for discovering robust solutions to complex games.

In the context of adaptive autonomous agents, Koza (1991, 1992) applied Genetic Programming to the co-evolution of pursuer-evader behaviors, Reynolds (1994) observed in a similar scenario that co-evolving populations of pursuers and evaders display increasingly better strategies, and Sims used competitive co-evolution to develop his celebrated artificial creatures (Sims, 1994). Cliff and Miller realised the potentiality of co-evolution of pursuit-evasion tactics in evolutionary robotics. In a series of papers, they described a 2D simulation of simple robots with evolvable "vision morphology" (Miller and Cliff, 1994) and 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 results where simulated agents with evolved eye-morphologies could either evade or pursue their competitors from some hundred generations earlier and proposed some applications of this methodology in the entertainment industry (Cliff and Miller, 1996).

In recent work (Floreano and Nolfi, 1997b), we have re-evaluated competitive co-evolution in the framework of evolutionary robotics resorting to realistic (in the sense that the same setup could be used for physical robots) computer simulations of two mobile robots, a predator and a prey (figure 3). The results and analyses showed the emergence of a set of different behavioral strategies in very short time (shorter than single-agent evolution) without effort in fitness design, such as obstacle avoidance, straight navigation, visual tracking, object discrimination (robot vs. wall), object following, and others. In this paper we significantly extend those results describing what is, to the best of our knowledge, the first physical implementation of competitive co-evolutionary robotics. We then provide an explanation of the results based on the visualization of the Red Queen Effect on the fitness surface of the experimental data. Finally, we give comparative data on different selection criteria indicating that co-evolution does not "optimize" intuitive objectives.

2. Experimental method

Predators and prey often belong to different species with different sensory and motor characteristics. We thus em-

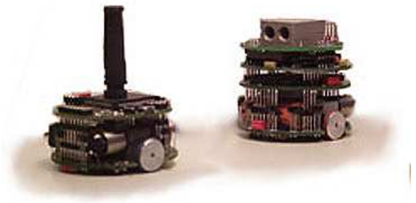


Figure 3 **Right**: The Predator is equipped with the vision module (1D-array of photoreceptors, visual angle of 36°). **Left**: The Prey has a black protuberance which can be detected by the predator everywhere in the environment, but its maximum speed is twice that of the predator. Both Predator and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

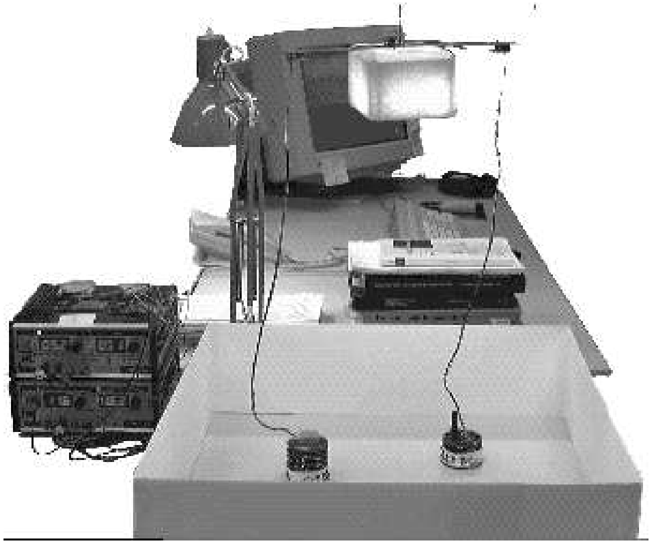


Figure 4 Setup to run co-evolutionary experiments on the physical robots. **Top**: the suspended bar with the three rotating contacts and a white box casting light over the arena. **Left**: two voltage transformers for powering the robots and the halogen lamp. **Background**: the workstation on which the two genetic algorithms run. The robots in the arena are equipped with contact detectors and wrapped in white paper to increase the reflection of infrared light.

ployed 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 3). Both species were also provided with eight infrared proximity sensors (six on the front side and two on the back): a wall could be detected at a distance of 3 cm approx., but the other robot can be detected at only half that distance because it offers a smaller reflection surface to infrared rays. The two species evolved within 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.

The two robots were connected to a desktop workstation equipped with two serial ports through a double aerial cable (figure 4). Aerial cables provided the robots with electric power and data communication to/from the workstation. The two cables ended up in two separate rotating contacts firmly attached to the far ends of a suspended thin bar. Both wires then converged into a single and thicker rotating contact at the center of the bar and ended up in the serial ports of the workstation and in two voltage transformers (on the left of figure 4). The thick rotating contact allowed the bar to freely rotate around its own center while the remaining two contacts allowed free rotations of the two robots. Attached under the bar was also a halogen lamp (20 W output) providing illumination over the arena.[1] Both robots were also fitted with a conductive metallic ring around their base to detect collisions. An additional general input/output module provided a digital signal any time the two robots hit each other (but not when they hit the walls). The motor bases of both robots were also wrapped by white paper in order to improve reflection of infrared light emitted by the other robot (approx. 1.5 cm against 0.5 cm). These two solutions are displayed in figure 4.

The vision module K213 of Khepera is an additional turret which can be plugged 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 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. However, a reliable image at lower illumination comes at the cost of a slower scanning speed of the 64 receptor values. This means that the image would be updated less frequently, thus giving an advantage to the prey (which indeed exploited it during exploratory experiments). Although one might decide to vary ambient illumination as a further independent variable, we decided to keep the halogen lamp switched on over the arena. In the simple environment employed for these experiments, the prey is at the eye of the predator like a valley whose width is proportional to the distance from the predator (figure 5) and whose position indicates the relative position of the prey with respect to the predator.

[1] No special care was taken to protect the system against external light variations between day and night.

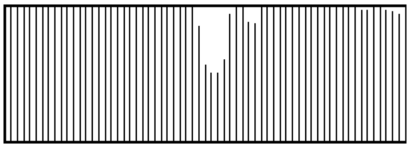


Figure 5 A snapshot of the visual field of the predator looking at the prey (taken at the position illustrated in figure 4). The heights of vertical bars represent the activations of corresponding photoreceptors. The black protuberance of the prey corresponds to a large valley. The small dip nearby corresponds to the cable. In standard illumination conditions, the image was refreshed at a rate of approximately 15-20 Hz.

2.1 Co-evolving neural controllers

In order to keep things simple, both the predator and the prey controllers were simple neural networks of sigmoid neurons. The input signals coming from the eight infrared proximity sensors (and from the vision module for the predator) were fed into two motor neurons with lateral and self-connections. The activation of each output unit was used to update the speed value of the corresponding wheel. Only the connection strengths were evolved. The maximum speed available for the prey was set to twice that of the predator.

In the case of the predator, the input layer was extended to include information coming from the vision module. The activation values of the 64 photoreceptors were fed into a layer of five *center off/surround on* neurons uniformely distributed over the retinal surface. The spatial sensitivity of each neuron was approximately 13° and the center/surround ratio filtered out low contrast features, such as those generated by weak shadows, the cable of the prey, and other imperfections of the walls. Each neuron generated a binary output of 1 when the prey was within its sensitivity field, and 0 otherwise. These five outputs were fed into the motor neurons along with the signals coming from the infrared sensors.

Given the small sizes of the neural controllers under co-evolution, we used *direct genetic encoding* of the connection strengths: each connection (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 bits 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 separate populations of N individuals each were co-evolved for g generations. Each individual was tested against the best competitors from k 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 from the initial population, whereas in the remaining $k-1$ initial generations they were randomly chosen from the pool of available best individuals (2 at generation 3, 3 at generation 4, etc.).

In our previous work (e.g., (Floreano and Mondada, 1996)) both the genetic operators and the robot controllers run on the workstation CPU and the serial cable was used to exchange sensory and motor information with the robot every 100 ms or longer. This method could not work in the current setup because transmission duration of visual signals and serial processing of the controller states for both robots on the same CPU significantly delayed and disturbed the interaction dynamics between the two robots. Here the adopted solution consisted in running the genetic operators on the workstation CPU and the neural controllers on the microcontrollers of the two Khepera robots. Each microcontroller, a Motorola MC68331 equipped with 128K ROM and 128K RAM was largely sufficient to store the set of instructions and variables necessary to handle all input/output routines and neural states. The speed of the sensorimotor cycles was set to approximately 15 Hz for both prey and predator. In the predator image acquisition and low-level visual preprocessing were handled by a private 68HC11 processor installed on the K213 vision turret.[2]

Each neurocontroller architecture, with the connection strengths initialized to zero, was downloaded into the corresponding robot. The two genetic algorithms were then started on the workstation CPU where each genetic string was decoded into a set of connection strengths and sent through the serial cable to the corresponding robot. Upon receipt of the connection strengths, each robot began to move and the internal clock (a cycle counter) of the prey was reset to zero. A tournament ended either when the predator hit the prey or when 500 sensorimotor cycles (corresponding to approximately 35 seconds) were performed by the prey without being hit by the predator. Upon termination, the prey sent back to the workstation CPU the value of the internal clock (ranging between 0 and 499) which was used as fitness value for both prey and predator. Upon receipt of the prey message, the workstation decoded the next pair of individuals and sent them back to both the predator and prey. In order to distantiate the two competitors at the beginning of each tournament, a simple random motion with obstacle avoidance was implemented by both robots for 5 seconds.

The fitness function Φ_i for species i was based only on the average *time to contact* over K tournaments,

$$\Phi_{py} = \frac{1}{K}\sum_{k=1}^{K}\frac{x_k}{500}, \quad \Phi_{pr} = \frac{1}{K}\sum_{k=1}^{K}\left(1-\frac{x_k}{500}\right),$$

[2] More details on this architecture and its relevance for evolutionary robotics are given in (Floreano and Mondada, 1998).

that is the number x_k of sensorimotor cycles performed in tournament k normalized by the maximum number of sensorimotor cycles available (500) in the case of the prey py, and the complement in the case of the predator pr, further averaged over the number of tournaments K. This fitness function rewarded prey capable of resisting longer before being hit by predators, and predators capable of quickly hitting prey. 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 by making an equal number of offspring in order to keep the population size constant. One-point crossover was applied on randomly paired strings with probability $pc = 0.6$, and random mutation (bit switching) was applied to each bit with constant probability $pm = 0.05$.

2.2 Analysis tools

The simple setup described above allowed us to develop a realistic simulation software (figure 6) that was used to perform initial explorations described in (Floreano and Nolfi, 1997b). This software has been used here only as a complementary analysis tool to investigate computationally expensive hypotheses on the dynamics of competitive co-evolution, and help to assess the results of the experiments on the physical robots. More details on the simulator and initial results are given in (Floreano and Nolfi, 1997b). Here it is sufficient to mention that the simulator was based on real sensory values sampled from the two robots (Miglino et al., 1996), not on a mathematical model of the environment.

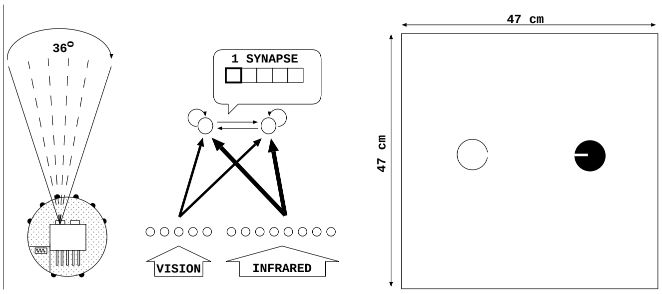


Figure 6 **Left and center**: Details of simulations. The prey differs from the predator in that it does not have 5 input units for vision. Each synapse in the network is coded by five bits, the first bit determining the sign of the synapse and the remaining four the connection strength. **Right**: Initial 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.

3. Results

An exploratory set of experiments were performed in simulation to understand the influence of various parameters, such as the number of tournaments with opponents from previous generations, crossover and mutation probabilities, replicability of the experiments, etc. A detailed analysis of these data is provided in (Floreano and Nolfi, 1997b). Here we provide only a summary of the basic results and compare them to the results obtained with the real robots. Two populations of 100 individuals each were co-evolved for 100 generations; each individual was tested against the best opponents from the most recent 10 generations. Figure 7 shows the average population fitness (left graph) and the fitness of the best individual at each generation. For each generation, the fitness values of the two species do not sum to one because each individual is tested against the best opponents recorded from the previous 10 generations.

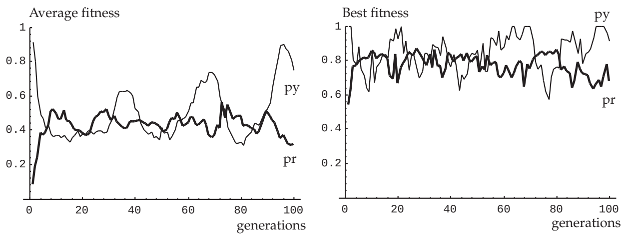


Figure 7 Co-evolutionary fitness measured in simulation. **Left**: Average population fitness. **Right**: Fitness of the best individuals at each generation. pr=predator; py=prey.

As expected, initially the prey score very high, whatever they might do, because the predators are not good at catching them; for the same reason, initially the predators score very low. Very quickly a set of counter-phase oscillations emerge in the two populations, as also reported by other authors (Sims, 1994, p. 36), but we never observed dominance of one population on the other in any of our evolutionary runs (even when continued for 500 generations). However, the fitness for the prey always tended to generate higher peaks due to position advantage (even in the case of the worst prey and best predator, the latter will always need some time to reach the prey). A similar pattern is observed for the fitness of the best individuals (right graph). However, these data cannot be taken as a measure of progress. The only information that they provide is the relative performance of the two species within a (moving) window of ten generations. They indicate that progress in one species is quickly counter-balanced by progress in the competing species, but do not tell us whether evolutionary time corresponds to true progress, or how to choose the best prey and the best predator from the point of view of optimization.

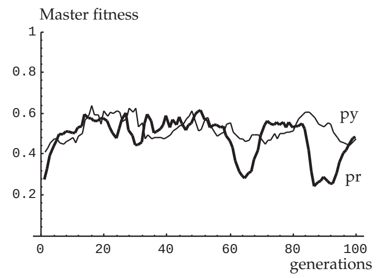


Figure 8 Master Fitness for species evolved in simulation. Each data point is the average fitness of all tournaments of the correponding individual against all the best 100 opponents recorded during co-evolution.

A simple way to learn more about absolute performance of the two species is to organize a Master Tournament where the best individuals for each generation are tested against the best competitors from all generations. For example, the best prey of generation 1 is tested against the 100 best predators and the scores of these tournaments are averaged. The Master fitness values reported in figure 8 indicate that *–in absolute terms–* individuals of the later generations are not necessarily better than those from previous ones. However, these measures can be used to tell *a)* at which generation we can find the best prey and the best predator; *b)* 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 highest fitness when also the competitor reports the highest fitness (here the best prey and predators are to be found at generation 20, 50, and 82). 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 (here the most entertaining tournaments are guaranteed around generation 20 and around generation 50).

The results with the real robot displayed a trend similar to that observed in simulations. Two populations (one for the prey, the other for the predator) of 20 individuals each were co-evolved for 25 generations (P(crossover)=0.6; P(mutation)=0.05 per bit) in approximately 40 hours of continuous operation (time might vary in different replications, depending on the relative performances of the two species). Each individual was tested against the best competitors from the most recent 5 generations. Figure 9 shows the average fitness of the population (left graph) and the fitness of the best individual (right graph) along generations for both species. Very quickly the two scores become closer and closer until after generation 15 they diverge again. A similar trend is observed for the fitness of the best individuals at each generation.

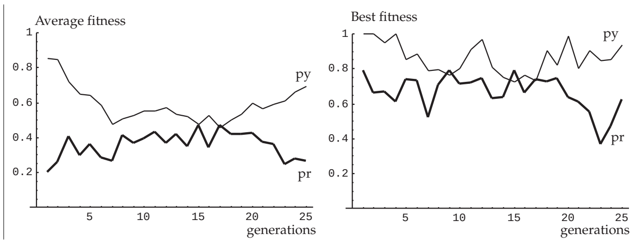


Figure 9 Co-evolutionary fitness measured on the real robots. **Left**: Average population fitness. **Right**: Fitness of the best individuals at each generation. `pr`=predator; `py`=prey.

25 generations are sufficient to display one oscillatory cycle. Once the relative fitness values of the two species reach the same value, one party improves over the other for some generations until the other counter-adapts (the best predators of the last three generations already show a fitness gain). Figure 10 shows the Master Fitness values for the two robot species. The best prey and predators can be found at generation 20 which also hosts the most entertaining tournaments. It can also be noticed that fitness oscillations of the best individuals between generation 9 and 16 (figure 9, right) do not show up in the Master Fitness, indicating that they are due to tight interactions between the two competing species which can amplify the effects of small behavioral differences.

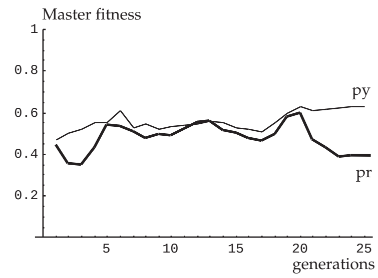
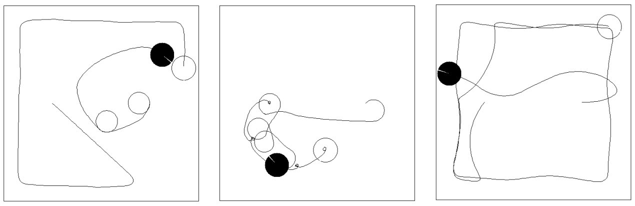


Figure 10 Master Fitness for species evolved on the real robots. Each data point is the average fitness of all tournaments of the correponding individual against all the best 25 opponents recorded during co-evolution.

The behaviors displayed by the two physical robots at significative points of co-evolution (for example, those corresponding to the overall best individuals and to the most entertaining tournaments) are only a subset of those recorded in simulation. The presence of much larger noise in the real environment filters out brittle solutions. Nevertheless, the best strategies displayed by the real robots can be found also in the experiments performed in simulation.[3] Figure 11 shows some typical tournaments recorded from individuals at generation 13, 20, and 22. At generation 13 the prey moves quickly around the environment and the predator attacks only when the prey is at a certain distance. Later on, at generation 20, the prey spins in place and, when the predator gets closer, it rapidly avoids it. Prey that move too fast around the environment sometimes cannot avoid an approaching predator because they detect it too late (infrared sensors have lower sensitivity for a small cylindrical object than for a large white flat wall). Therefore, it pays off for the prey to wait for the slower predator and accurately avoid it. However, the predator is smart enough to perform a small circle after having missed the target and re-attack until, by chance, the prey is caught on one of the two sides (where wheels and motors do not leave space for sensors). The drop in performance



generation 13 generation 20 generation 22

Figure 11 Typical strategies of the best predator and prey in the experiments with real robots. Black disk is the predator, white disk is the prey. Trajectories have been plotted running a tournament with simulated individuals who display the same behavioral strategies observed with the real robots.

of the predator in the following generations is due to a temporary loss of the ability to avoid walls (which was not needed in the few previous generations because the predator soon localized and approached the prey). At the same time the prey resumes a rapid wall following and obstacle avoidance which forces the predator to get closer to walls and collide if the prey is missed (right of figure 11).

4. Discussion

Co-evolution between competing species with a relatively short generational overlap does not seem to guarantee the type of monotonic progress over time expected from an optimization algorithm. This does not mean that competitive co-evolution is not an viable approach for the purpose of evolving efficient behavioral competencies. In fact, the rapid development, variety, and complexity of the behaviors observed, together with appropriate methods to pick them out (such as the Master Tournament), hint at the computational advantages of competitive co-evolution. In a companion paper, we report results showing that predators evolved against a fixed co-evolved prey (from another run) do not reach the same performance levels obtained when both prey and predators are co-evolved (Nolfi and Floreano, 1998). It has also been shown that, by including all the best opponents evolved so far as test cases for each individual (*Hall of Fame* method), co-evolution becomes very similar to a very robust optimization technique (Rosin and Belew, 1997).

However, the basic issue that the results presented above raise is to what extent natural evolution is an optimization process (see also (Gould and Lewontin, 1979)) and to what extent should we use artificial evolution as an optimization technique in a traditional fashion. In contrast to mainstream "evolutionary computation", there is not necessarily one or more fixed maxima to be found in competitive co-evolution. The landscape continuously changes. Furthermore, the way it changes cannot be predicted, and most of the time it will change towards the worst for each competing species. Nevertheless, after an initial period during which the two populations settle into a regime of tight interactions (corresponding to the generation when the two fitness measures become equal), the best individuals of the two populations are always optimal, or almost always, optimal with respect to the environment (competitor) that they are facing. In other words, the optimum is always now.

Other authors have criticized evolutionary computation for its interpretation of evolution as an optimization process (Atmar, 1993) and pointed out its limitations within the framework of incremental evolution (Harvey, 1997). We agree with Harvey that one should not look for the optimization properties of evolution, but rather for its "adaptive improving properties". To this extent, co-evolution of predator-prey scenarios, such as the one described in this paper, not only displays these properties, but also provides a framework for understanding some of the issues brought in by open-ended evolution and testing different methodologies. Indeed, elsewhere (Nolfi and Floreano, 1998) we have shown that under certain conditions "plain competitive co-evolution" can provide better adaptativity than single-agent evolution and methods aimed at improving the (traditional) optimization properties of co-evolution, such as the Hall of Fame mentioned above.

The results described above in simulation and on the real robots are very similar from the point of view of the dynamics of the co-evolutionary system. The main difference is that the larger amount of noise in the real experiments does not allow the persistence of behaviors which instead we observe in simulation. These behaviors are based on the synchronous and time-locked activity of the two individuals which does not hold when each physical robot moves around independently using its own

[3] Although individuals *evolved in simulation* do not behave in the same way when *downloaded into the real robots*.

controller and updates the wheel speed at a slightly different frequency. Noisier conditions may also explain the reason why when the system is run on the physical robots it takes slightly longer to reach the lock-phase where oscillations begin. In effect, noise may be seen as a way of increasing the variety of behaviors "observed" by each opponent. Therefore, individuals must search for more "general solutions". These results, which are in accordance with other data on simulations with high noise levels (Floreano and Nolfi, 1997a), also explain why fitness values recorded during evolution are a better indicator of "true" performance levels (as indicated by the Master Tournament).

As well-known in the Adaptive Behavior community, though, the advantages of running the experiments on the real robots show up as soon as we introduce more complexities, such as a less constrained visual preprocessing (which here was very easy to simulate) and, for example, a vision system for the prey too. At that point not only computer simulations become slower than the physical implementation, but also competitive co-evolution might become a powerful tool to discover solutions that would not be evolvable otherwise. For the time being, the rough similarity between our (careful) simulations and physical implementations allow us to exploit simulations for exploring other computationally expensive features of competitive co-evolution.

4.1 Exposing the Red Queen

The Red Queen effect illustrated in figure 1 is suspected to be the main actor behind the dynamics, complexities, and computational advantages of competitive co-evolution, but how exactly it operates is not known. Capitalizing on the fact that our simple experiment with the robots displayed dynamics similar to those measured in experiments carried out in simulation, we exploited our computer CPUs to study how the fitness surface of one species is affected by the co-evolving competitor.

Given its shorter genotype length, we analyzed how the fitness surface of the prey was changed when confronted with the best predators recorded at different generations. The genotype of the prey was composed of 5 bits x 22 synapses. Assuming that the most significative bits are those coding the sign of the synapses, we are left with 22 bits.[4]. Each of the corresponding 4,194,304 prey was separately tested against the best predators of the first eight generations and against the best predator of generation 20, yelding a total of almost 40 million tournaments. At the beginning of each tournament, both the prey and the predator were positioned at the same location (illustrated on the right of figure 6) facing north. The best predators were selected from the simulation run depicted in figure 7. Figure 12 plots the fitness surface of

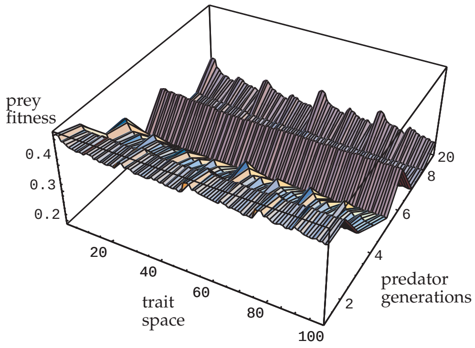


Figure 12 Fitness landscape for the prey when tested against the best predators from generation 1-8 and 20. Each data point is the average over the fitness values reported by 4,194 contiguous individuals.

the prey tested against best predators from generations 1-8 and generation 20. Since there are not enough many pixels on this page to show the results of all tournaments, the fitness values for each generation were grouped into 100 bins of 4,194 values each (discarding remainders) and the average value of each bin was plotted on the graph.

Despite these approximations, one can see that co-evolution of predators during initial generations cause a general decrement of the performance of the prey. However, it should be noticed that these are average values and that for every bin there are always several prey reporting maximum fitness 1.0. The Red Queen effect is clearly visible in the temporary and periodic smoothing of the fitness landscape, as highlighted in figure 13. For example, the best predator of generation 3 causes a redistribution of the fitness values, stretching out the relative gain of some trait combinations with respect to others. This smoothing effect is always temporary and roughly alternates with recovery of a rough landscape.

It should be noticed that some regions corresponding to better fitness remain relatively better also during periods of stretching, whereas others are canceled out. That implies that individuals sitting on these latter regions would disappear from the population. If we view these regions as minima or brittle solutions, our data show the potentials of the Red Queen for optimization problems. Furthermore, it can be noticed that the steepness of the surface around the maxima becomes more accentuated along generations. If we assume that steeper regions are harder to climb, competitive co-evolution might facilitate progressive development of abilities that would be difficult to achieve in the scenario of a single species evolved in a static environment. Although more analysis is to be carried out on these data, especially on the displacement of the prey population on this changing surface, these preliminary results support the hypotheses on the

[4] The remaining 4 bits for each synapse were set at 0101, a pattern that represents the expected number of on/off bits per synapse and also codes for the average synaptic strength.

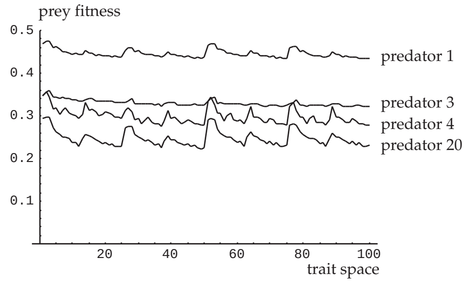


Figure 13 The Red Queen in action. See also figure 12.

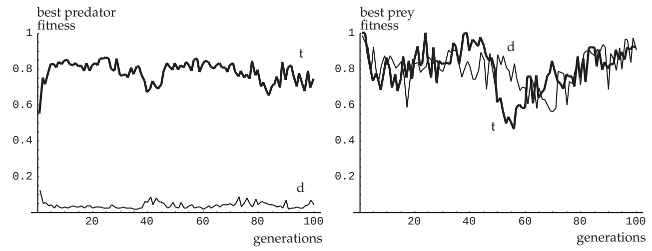


Figure 14 Comparisons between fitness of the best individuals measured as *time to contact* (t) and as *distance*. (d). Both species have been evolved using fitness t. **Left**: Best evolved predators do not attempt to minimize distance (predator fitness is 1 - d). **Right**: Best evolved prey attempt to maximize distance (prey fitness is d).

Red Queen effect outlined in section 1. and indicate its efficacy from an optimization perspective.

4.2 Selection criteria

In artificial evolution the choice of the selection criterion (fitness function) can make the difference between trivial parameter optimization and generation of creative and "life-like" solutions (Floreano, 1997). From an engineering point of view, one might think that it makes sense to pursue an optimization approach by including several constraints in the function in the attempt to describe the expected behavior. However, by doing so one might also include wrong assumptions that derive from an insufficient understanding of the environment and/or of the interactions that might arise. For example, from an engineering point of view, a successful predator should aim at the prey and approach it minimizing the distance, whereas the prey should attempt to maximize this distance, as in (Cliff and Miller, 1996). Did our robots (simulated and real) indirectly optimize this objective although the selection criterion employed was simply *time to contact*? We run a new set of simulations where each individual was selected and reproduced according to the fitness function described in section 2.1, but was also evaluated according to a fitness function based on the *distance* between the two competitors (namely, the *distance* for the prey, and *1 - distance* for the predator).

The fitness values computed according to the two methods (figure 14) did not overlap for the predators, but they did for the prey. In other words, predators selected to hit prey in the shortest possible time did not attempt to minimize the distance from the prey, as it might be expected. On the other hand, the prey did so. The strategy employed by the predators was more subtle. Rather than simply approaching the prey, they tended to "wait for the right moment" and, only then, attack. The behaviors shown in the first two insets of figure 11 are an example of this strategy. The best predator of generation 13 attacks only when the prey is within a certain range, and rotates in place in the other cases (neuroethological analyses showed that the predator infers the distance from the prey by observing how fast the prey moves on its own visual field). Similarly, the best predator of generation 20 does not follow the prey once the latter has moved away; rather, it performs one more turn on itself and re-attacks. By doing so, it has higher probability of hitting the prey on the side of the motors where there are no sensors. Competitive co-evolution is a methodology capable of discovering innovative and unexpected –yet efficient– solutions provided that one does not attempt to force it into specified directions and reduce it to a mere optimization technique.

5. Conclusion

Competitive co-evolution is not necessarily optimization in the sense of solution discovery for a pre-specified problem. The teleological interpretation of artificial evolution that underlies most research in genetic algorithms (Atmar, 1993) and some research in evolutionary theory (Gould and Lewontin, 1979) leaves space to a more complex and richer scenario where robust solutions and true innovation can endogenously arise from simple interactions between parts of the (co-evolutionary) system. In this paper we have shown that this methodology can be applied in evolutionary robotics in order to generate powerful controllers with minimal human effort. At the same time we have tried to give indications that competitive co-evolution can be a viable methodology for developing truly autonomous and intelligent machines capable of setting their own goals to face new and continuously changing challenges. Much work remains to be done to check and further extend the indications that arise from these results. Our current research is focused on the interactions between learning and co-evolution (Floreano and Nolfi, 1997a), and on the extension of formal models of competitive co-evolution that could establish a link between reproduction success and behavioral performance.

Acknowledgements

We would like to thank the two anonymous referees for rich and constructive comments.

References

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.

Atmar, W. (1993). Notes on the Simulation of Evolution. *IEEE Transactions on Neural Networks*, 5:130–148.

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., Matarić, 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.

Dawkins, R. and Krebs, J. R. (1979). Arms races between and within species. *Proceedings of the Royal Society London B*, 205:489–511.

Floreano, D. (1997). Ago Ergo Sum. In Mulhauser, G., editor, *Evolving Consciousness*. Benjamins Press, New York.

Floreano, D. and Mondada, F. (1996). 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. (1998). Hardware Solutions for Evolutionary Robotics. In Husbands, P. and Meyer, J., editors, *Proceedings of the first European Workshop on Evolutionary Robotics*. Springer Verlag, Berlin.

Floreano, D. and Nolfi, S. (1997a). Adaptive behavior in competing co-evolving species. In Husbands, P. and Harvey, I., editors, *Proceedings of the 4th European Conference on Artificial Life*, Cambridge, MA. MIT Press.

Floreano, D. and Nolfi, S. (1997b). God Save the Red Queen! Competition in Co-evolutionary Robotics. In Koza, J., Deb, K., Dorigo, M., Fogel, D., Garzon, M., Iba, H., and Riolo, R. L., editors, *Proceedings of the 2nd International Conference on Genetic Programming*, San Mateo, CA. Morgan Kaufmann.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning.* Addison-Wesley, Redwood City, CA.

Gould, S. J. and Lewontin, R. C. (1979). The spandrels of San Marco and the Panglossian paradigm: a critique of the adaptationist program. *Proceedings of the Royal Society London, B*, 205:581–598.

Harvey, I. (1997). Cognition is not computation; evolution is not optimization. In Gerstner, W., Germond, A., Hasler, M., and Nicoud, J., editors, *Artificial Neural Networks - ICANN97*, pages 685–690, Berlin. Springer Verlag.

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

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.

Lotka, A. J. (1925). *Elements of Physical Biology.* Williams and Wilkins, Baltimore.

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.

Murray, J. D. (1993). *Mathematical Biology.* Springer Verlag, Berlin. Second, Corrected Edition.

Nolfi, S. and Floreano, D. (1998). Co-evolving predator and prey robots: Do "arms races" arise in artificial evolution? *Artificial Life.* To be published.

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.

Rosin, C. and Belew, R. (1997). New methods for competitive co-evolution. *Evolutionary Computation*, 5(1):1–29.

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.

Utida, S. (1957). Population fluctuations. In *Proceedings of the XXII CSHL Symposium on Population Studies: Animal Ecology and Demography*, Cold Spring Harbor Laboratory, NY.

van Valen, L. (1973). A new evolutionary law. *Evolution Theory*, 1:1–30.

Volterra, V. (1926). Variazioni e fluttuazioni del numero di individui in specie animali conviventi. *Memorie dell'Accademia dei Lincei*, 2:31–113. Variations and fluctuations of the number of individuals in animal species living together. Translation in: R. N. Chapman, Animal Ecology. New York: McGraw Hill 1931, pp. 409–448.