1 Co-Evolution and Ontogenetic Change in Competing Robots

Dario Floreano, Stefano Nolfi, and Francesco Mondada

We investigate the dynamics of competitive co-evolution in the framework of two miniature mobile robots, a predator with a vision system and a faster prey with proximity sensors. Both types of robots are controlled by evolutionary neural networks. A variety of efficient chase-escape behaviors emerge in few generations. These results are analyzed in terms of variable fitness landscapes and selection criteria. A new vision of artificial evolution as generation and maintainance of adaptivity is suggested and contrasted with the theory and practice of mainstream evolutionary computation. In a second stage, different types of ontogenetic changes applied to the robot controllers are compared and the results are analyzed in the context of competitive co-evolution. It is shown that predators benefit from forms of directional changes whereas prey attempt to exploit unpredictable behaviors. These results and their effect on co-evolutionary dynamics are then considered in relation to open-ended evolution in unpredictably changing environments.

1.1 Introduction

In a competitive co-evolutionary system the survival probability of a species is affected by the behavior of 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 survival probability of an individual is tightly related to the behaviors of the competitors both on the ontogenetic and on the evolutionary time scale. Behavioral changes in one lineage might affect the selection pressure on the other lineage and, if the other lineage responds with counter-adaptive features, one might observe what some biologists call a “a co-evolutionary arms race” [4]. Consider for example the well-studied case of two co-evolving populations of predators and prey [25]: the success of predators imply the failure of prey and viceversa. Evolution of a new behavior in one species represents a new challenge for the other species which is required to evolve new strategies. The continuation of this co-evolutionary process may produce increasingly higher levels of complexity in the behavioral strategies of the two competing species (although this is not guaranteed).

On the ontogenetic time-scale, it has been argued that pursuit-evasion contests might favor the emergence of “protean behaviors”, that is behaviors which are adaptively unpredictable [5]. For example, prey could take advantage of unpredictable escape behaviors based on short sequences of stochastic motor actions. On the other hand, predators could take advantage of adaptation abilities that occur on a faster time-scale than generations.

The purpose of this research is to investigate the dynamics of competitive co-evolution and the effects of ontogenetic adaptive changes from the perspective of artificial evolution. The testbed chosen for our experiments is a predator-prey scenario where two mobile robots, each representing an evolving population, compete with each other to – respectively – chase or escape the other robot. The fact of using physical robots constraints us to use only local computation, simple assumptions on the rules of the game, and a very general (simple) fitness function. It will be shown that these "physical constraints" have significant effects on the outcome of the experiments. In order to accelerate the exploration of different experimental conditions and run multiple experiments for statistical analysis, we have resorted to realistic simulations of the physical setup.[1]

We shall show that competitive co-evolution alone (without ontogenetic adaptive changes) can quickly develop efficient chase and evasion strategies in both robot species. A retrospective analysis of the fitness landscape after a co-evolutionary run will give some hints on the incremental aspects of co-evolutionary dynamics. We will also see that using a simple fitness function allows predators to evolve non-trivial behaviors. However, after some time (generations) co-evolutionary dynamics fall into a limit cycle where predators and prey rediscover over and over again the same class of strategies. After explaining the adaptive power of this solution, we will investigate how the introduction of ontogenetic adaptation will affect the co-evolutionary dynamics and show that predators and prey exploit different types of ontogenetic adaptive changes. Finally, we will discuss the role of ontogenetic adaptation in the perspective of co-evolving individuals.

The Red Queen

From the computational perspective of *artificial evolution*, the reciprocal effects of changes induced 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 1.1, left), there is a stable relationship between the traits of an organism and its reproduction success.

[1] The software allows us to switch between physical and simulated experiments by toggling the value of a single variable.

In this case evolution is often seen as a force driving the population towards combinations of traits that maximize reproduction success [15].

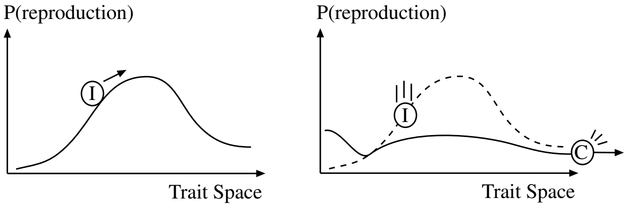

Figure 1.1
Left: Reproduction probability of a single species *I* under evolution in a static environment. Evolution drives the organisms towards zones (combinations of traits) corresponding to higher reproduction success. **Right**: Reproduction probability of species *I* under competitive co-evolution. 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* [34].

Instead, in competitive co-evolution the relationship between traits and reproductive success of the organism can change over time. Behavioral changes in one species could make some traits of the competing species no longer useful for reproductive success (figure 1.1, 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” [34] (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).

From a computational perspective, it would be interesting to understand whether the Red Queen Effect can seriously limit the incremental adaptive characteristics of competitive co-evolution. Theoretical models of competitive co-evolution (based on Lotka-Volterra equations [23, 35]) study how population density (i.e., the number of individuals) varies as a function of pre-defined abilities of the two competing species [26]. These models cannot help us to predict whether artificial competitive co-evolution can be exploited for the purpose of evolving increasingly more complex behaviors. Therefore, at the current stage, experimental work still seems to be a powerful tool to investigate these issues.

Related Work

Prospects of continuous and incremental progress have triggered several attempts to apply competitive co-evolution to difficult problems. Hillis [19] reported a significant 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 on co-evolution of players for the Tic-Tac-Toe and Nim games [1, 31]. Recently, Funes et al. [14] have studied co-evolution of machines and humans competing over the Internet and reported strategy improvement in both populations over time.

In the context of adaptive autonomous agents, Koza [20, 21] applied Genetic Programming to the co-evolution of pursuer-evader behaviors, Reynolds [30] 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 [33]. 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" [25] 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 [2]. 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 [3]. However, these experiments revealed more difficulties than expected and not all experimental runs produced successful pursuers and evaders. Also other authors have stressed that competitive co-evolution strongly depends on the scoring criterion and can easily degenerate in mediocre solutions [6].

In this article we address the realization of physical mobile robots that co-evolve in competition with each other (figure 1.2). Although the results presented in the following pages are based on both real and simulated experiments, the physical implementation constrains our design in ways that significantly affect the outcome of the experiments and allow us to explore a variety of issues related to the autonomy and adaptation of artificial organisms.

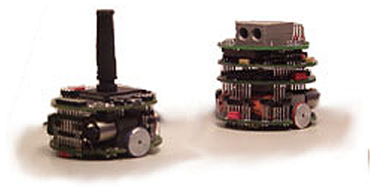

Figure 1.2
Right: The Predator is equipped with the vision module (1D-array of photoreceptors, visual angle of 36°). **Left**: The Prey has a black protuberance which can be detected by the predator everywhere in the environment, but its maximum speed is twice that of the predator. Both Predator and Prey are equipped with 8 infrared proximity sensors.

1.2 Experimental set-up

We employed two Khepera robots, one of which (the *Predator*) was equipped with a vision module while the other (the *Prey*) did not have the vision module, but its maximum speed was set twice that of the predator (figure 1.2). Both robots were also provided with eight active (emitting-measuring) infrared proximity sensors (six on the front side and two on the back). These sensors could detect a wall at a distance of approximately 3 cm and another robot at a distance of approximately 1 cm because of its smaller reflection surface. The two robots 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 (figure 1.3). The two robots were connected to a desktop workstation equipped with two serial ports through a double aerial cable. 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 1.3). 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 (figure 1.4). Attached under the bar was also a halogen lamp (20 W output) providing illumination over the

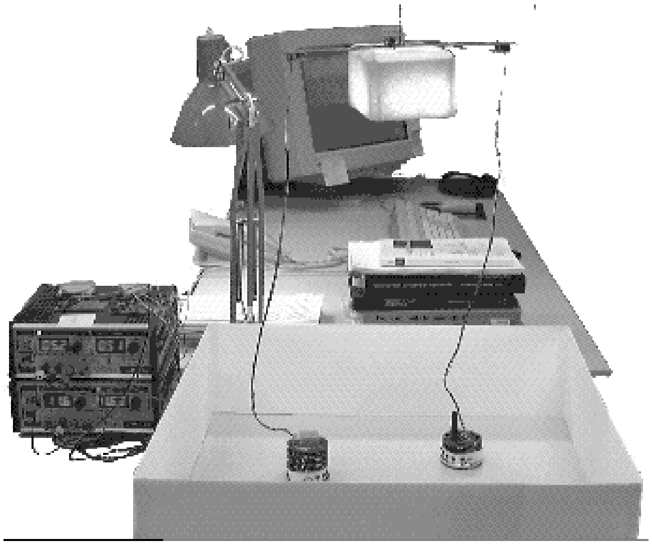

Figure 1.3
Setup to run co-evolutionary experiments on the physical robots. **Top**: the suspended bar with the three rotating contacts and a white box casting light over the arena. **Left**: two voltage transformers for powering the robots and the halogen lamp. **Background**: the workstation on which 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.

arena.[2] 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 (figure 1.3).

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 providing 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 70cm 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

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

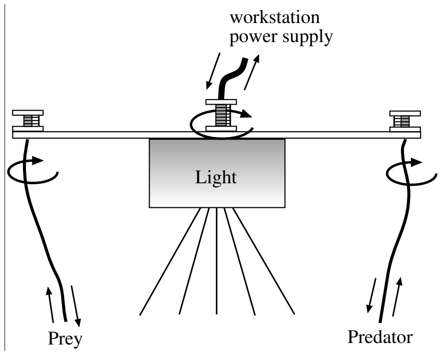

Figure 1.4
The suspended bar with the three rotating contacts and a white box casting light over the arena.

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). This is the reason why we added the halogen lamp to the rotating bar over the arena. In the simple environment employed for these experiments, the projection of the prey onto the artificial retina of the predator looks like a valley (top of figure 1.5) that indicates the relative position of the prey with respect to the predator.

Controller architecture, evolutionary algorithm, and method

Both 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 (forward from 0.5 to 1.0, backward from 0.5 to 0.0). Only the connection strengths were evolved. The maximum speed available for the prey was set to twice that of the predator.

The input layer of the predator 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 (bottom of figure 1.5). The spatial sensitivity of each neuron was approximately 13° and the center/surround ratio filtered

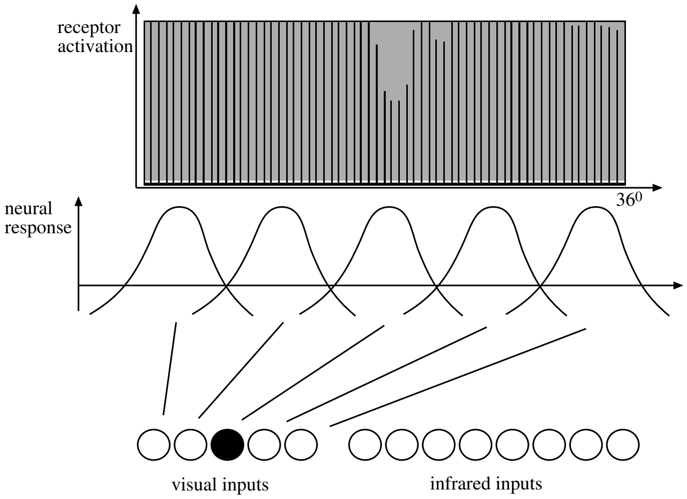

Figure 1.5
Top: A snapshot of the visual field of the predator looking at the prey. The heights of vertical bars represent the activations of the 64 photoreceptors. The black protuberance of the prey looks like a large valley. The small dip on the right of the valley corresponds to the cable. In the illumination conditions of the experiment, the image is refreshed at a rate of approximately 15 Hz. **Bottom**: Visual filtering with five *center/off surround/on* neurons. A neuron is maximally activated when the projection of the prey falls within its receptive field. The most active neuron is set to 1, all the remaining neurons are set to 0 in a Winner-take-All fashion.

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 size of the neural controllers under co-evolution, we used *direct genetic encoding* of 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 times [30 synapses + 2 thresholds] = 160* bits long while that of the prey was *5 times[20 synapses + 2 thresholds] = 110* 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 [33, 30, 2]) in order to improve co-evolutionary stability.

At generation 0, competitors were randomly chosen from the initial population, and later on they were randomly chosen from the pool of best individuals from previous generations (2 at the 3rd generation, 3 at 4th generation, . . . , 49 at 50th generation, etc.).

In our previous work (e.g., [8]) both the genetic operators and the robot controllers were run on the workstation 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 times and serial processing of the controller states for both robots on the same workstation CPU significantly delayed and disturbed the interaction dynamics between the two robots.

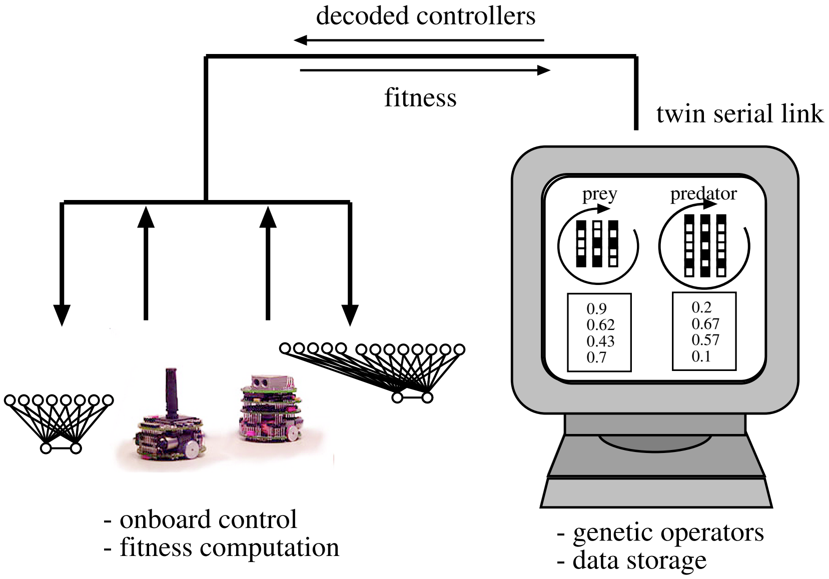

Figure 1.6
The genetic operators run on the main workstation, which also manages data storage and analysis; the neural controllers are automatically downloaded on the microcontrollers of the robots through the serial link. In the predator, an additional microprocessor on the vision module performs visual pre-processing and sends data at 15 Hz frequency to the main microcontroller.

Therefore, we splitted the computational load by running the genetic operators on the workstation CPU and the neural network on the microcontroller of each Khepera robot, a Motorola MC68331 equipped with 128K ROM and 128K RAM. The speed of the sensorimotor cycles was set to approximately 15 Hz for both prey and predator. For the predator, image acquisition and low-level visual preprocessing were handled by a private 68HC11 processor available on the K213 vision turret (see [11] for more details of this modular architecture).

At the beginning of a co-evolutionary run, the two neurocontrollers, with the connection strengths initialized to zero, were downloaded into the corresponding robots. Two genetic algorithms (with the same parameters in these experiments) 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, the internal clock (a cycle counter) of the prey robot was reset to zero and each robot began to move. 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 measure 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 reposition 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 (where i indicates the species) was based only on the average *time to contact* over K tournaments,

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

where x_k is the number of sensorimotor cycles performed in tournament k. This value is 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, and further averaged over the number of tournaments K for both robots. This fitness function rewarded prey capable of resisting longer before being hit by predators, and predators capable of quickly hitting prey. This simple function differs from those used by Cliff and Miller who included distance between the two individuals in order to evolve pursuit and escape behaviors. In our physical setup it was impossible to know the exact distance between the two robots. It will be shown that this subtle difference significantly affects the evolved behaviors.

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 $pm = 0.05$.

Software model

A software model of this physical setup was developed in order to test different experimental conditions and several replications with different random initializations, and to carry out explorations of the fitness landscape. In order to ensure a good match between simulations and physical experiments, we used a sampling technique proposed by Miglino *et al.* [24].

Each robot was positioned close to a wall of the environment and performed a full rotation by steps of 10°. At every step, all sensor values were recorded and stored in a table. The robot was then positioned 2 mm from the wall and the same procedure was applied again. This technique was repeated every 2 mm up to a distance of 4 cm (more than the maximum sensitivity of the infrared sensors). The final table stored all sensor values for 20 distances from the wall and, at each distance, for 180 uniformly-spaced orientations. Another table was built in the same way by positioning the robot in front of the other robot. Notice that the combination of these two tables is sufficient to represent every possible situation that a robot might encounter in this environment (flat wall, corner, other robot, and combinations of them). These tables were used to retrieve appropriate sensor values from the robot current position in the simulator. The activation of the five vision neurons was computed using trigonometric functions. The main difference between the software model and the physical implementation was that in the former case the initial position of the two robot was always the same (in the center of the arena, 5 cm apart from each other), but at different random orientations, whereas in the latter case the two robots moved randomly for five seconds avoiding obstacles.

In a set of preliminary comparisons, it was found that results obtained with the software model did not differ significantly from the results obtained with physical robots both in terms of co-evolutionary dynamics and in terms of behavioral strategies. More details will be given in the next section.

1.3 Co-evolutionary 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 [12]. 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 1.7 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.

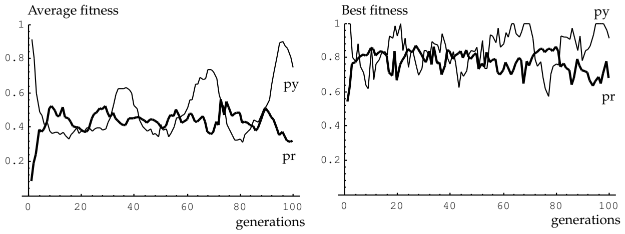

Figure 1.7
Co-evolutionary fitness measured in simulation. **Left**: Average population fitness. **Right**: Fitness of the best individuals at each generation. pr=predator; py=prey.

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. In less than 20 generations a set of counter-phase oscillations emerge in the two populations, as reported by other authors [33, p. 36] too, 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 of the prey always tended to generate higher peaks than that of the predator due to the position advantage (even in the case of the worst prey and best predator, the latter will always take some time to reach the prey). A similar pattern is observed for the fitness of the best individuals (right graph).

These data are not sufficient to assess whether there is real 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 generates true progress, or how to choose the best prey and the best predator out of all evolved individuals.

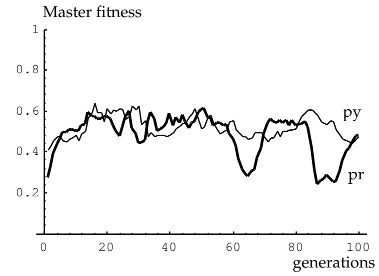

Figure 1.8
Master Fitness for species evolved in simulation. Each data point is the average fitness of all tournaments of the correponding individual against all the best 100 opponents recorded during co-evolution.

A simple way to learn more about absolute performance of the two species consists of organizing 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 100 best predators and the scores of these tournaments are averaged. If there is true progress, i.e. if a species develops strategies capable of defeating an increasing number of competitors, the Master Fitness should display an ascending trend. However, Master fitness values for these experiments (figure 1.8) indicate that *–in absolute terms–* individuals of later generations are not necessarily better than those from previous ones. In other words, individuals of early generations have strategies that can defeat individuals of later generations.

Master Fitness 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, 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).

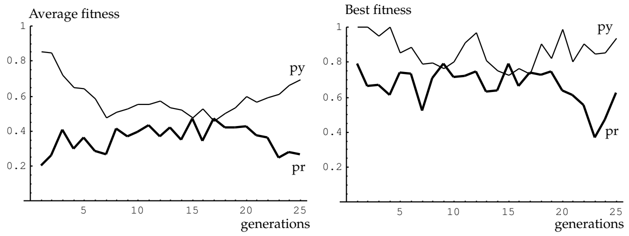

Figure 1.9
Co-evolutionary fitness measured on the real robots. **Left**: Average population fitness. **Right**: Fitness of the best individuals at each generation. `pr`=predator; `py`=prey.

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 1.9 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.

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 1.10 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 1.9, right) do not

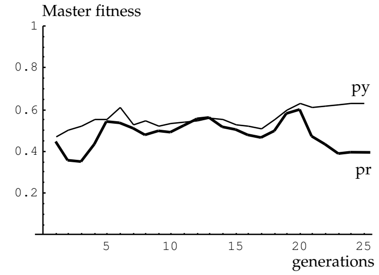

Figure 1.10
Master Fitness for species evolved on the real robots. Each data point is the average fitness of all tournaments of the correponding individual against all the best 25 opponents recorded during co-evolution.

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.

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 that are instead stable in simulations. Nevertheless, *all* strategies displayed by the real robots can be found also in the experiments performed in simulation.[3] Figure 1.11 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 of the predator in the following generations is due to a

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

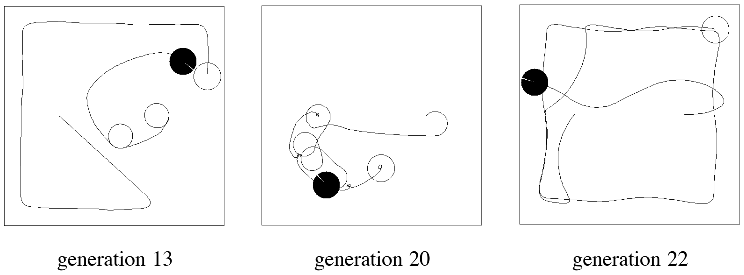

Figure 1.11
Typical strategies of the best predator and prey in the experiments with real robots. Black disk is the predator, white disk is the prey. Trajectories have been plotted running a tournament with simulated individuals who display the same behavioral strategies observed with the real robots.

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 1.11). A description of additional behaviors obtained in simulations is given in [12].

Machine Learning and natural adaptation

The results described above indicate that co-evolution between competing species with a relatively short generational overlap does not necessarily display the type of monotonic progress over time expected from the optimization-oriented approach that characterizes traditional Machine Learning.

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 discovery, variety, and complexity of behaviors observed, together with appropriate methods to pick them out (such as the Master Tournament), hint at the computational advantages of competitive co-evolution. For example, in a companion paper we report results obtained in different environmental conditions where 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 [29]. In other words, under certain circumstances competitive co-evolution can indeed produce more powerful solutions that standard evolution. Furthermore, 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 displays monotonic progress and becomes very similar to a very robust optimization technique [31, 29]. Finally, other results obtained on variations of the basic physical setup described in this article display ascending trends on the Master Tournaments [29].

However, the basic issue here is to what extent natural evolution should be interpreted as an optimization process (see also [17, 16] for a critical review) and to what extent should we use artificial evolution as an optimization technique in the spirit of traditional Machine Learning. Competitive co-evolution is closer to biological evolution in that the challenges faced by evolving individuals continuosly and unpredictably change over time. Under these circumstances, the classic notion of optimal solutions seems inadequate because at every point in time it is impossible to know what the next challenge will be.

With respect to mainstream "evolutionary computation" devoted to searching for peaks on multidimensional and complex, but static, fitness landscapes [15], in a competitive co-evolutionary scenario the landscape continuously changes. Not only one cannot predict the way in which it changes, but very often it will change towards the worst from the point of view of each species (see the right portion of figure 1.1). As a matter of fact, for a co-evolving species there is not a fitness landscape to climb. In other words, at every point in time and space there is not a gradient that can be reliably followed. Therefore, fitness landscapes are useful only for *a posteriori* observations, but are not actually used by a system evolving in unpredictable dynamic environments.

Within this framework, artificial evolution should not be conceived as a method for finding optimal solutions to predefined problems, but as a mechanism for encouraging and developing adaptivity. For example, generation and maintainance of diversity might be an adaptive solution. Another adaptive solution could be that of selecting genotypes whose mutants correspond to different but all equally viable phenotypes. In both cases, the species can quickly respond to environmental changes. Yet another way of evolving adaptivity would be that of encouraging ontogenetic adaptive changes. None of these notions of evolution implies continuous progress as usually defined in the machine learning literature.

Revisiting the experiments described above in this new light, after an initial period during which the two populations settle into a regime of tight interactions (when the two fitness measures become equal), the best individuals of the two populations are always optimal, or almost always, with respect to the environment (competitors) that they are facing. In other words, the optimum is always now.

Exposing the Red Queen

The Red Queen effect illustrated in figure 1.1 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 the workstation CPU to study how the fitness landscape of one species is affected by the co-evolving competitor. Let us remember that here the notion of fitness landscape is valid only as an observation metaphor using data collected during a co-evolutionary run.

Given the shorter genotype length of the prey, we analyzed how the its fitness surface changed when the prey was confronted with the best predators saved from successive generations. Let us recall that the genotype of the prey was composed of 5 bits x 22 synapses (see subsection 1.2). Assuming that the most significative bits are those coding the sign of the synapses, we are left with 22 bits.[4]. The combination of these 22 genetic traits corresponds to 4,194,304 prey. Each 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. In order to facilitate the comparison, at the beginning of each tournament, both the prey and the predator were positioned at the same location facing north. The best predators were selected from the simulation run depicted in figure 1.7. Since there is not enough space on this page to plot the fitness values of all the prey against each predator, the 4,194,304 fitness values were grouped into 100 bins of 4,194 values each (discarding remainders) and the average value of each bin was plotted on figure 1.12.

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 1.12. For example, the best predator of generation 3 causes a redistribution of the fitness values, stretching out the relative gain of some combinations of traits with respect to others. This smoothing effect is always temporary and roughly alternates with recovery of

[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.

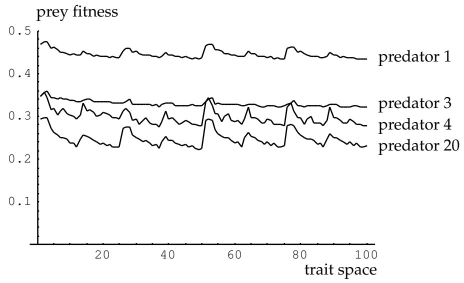

Figure 1.12
The Red Queen in action. Fitness landscape for the prey when tested against the best predators saved from generation 1, 3, 4, and 20. Each data point is the average over the fitness values reported by 4,194 prey close on the genotype space.

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 selecting robust solutions. 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 more difficult to reach, 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. In other words, at least during the initial generations, coevolution might spontaneously generate a set of tasks of increasing difficulty, a sort of “pedagogical series of challenges” [31] that might favor fast emergence of complex solutions.

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 [7]. From an engineering point of view, it might seem reasonable to pursue an optimization approach by designing a detailed fitness function that attempts to develop behaviors that are partially pre-defined. 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 between the robot and its environment [28]. For example, one might think that a successful predator should aim at the prey and approach it minimizing the distance, whereas the prey should attempt to maximize this distance. Consequently, she would design a fitness function that includes –respectively–distance minimization and maximization for predators and prey, as in [3].

Although the fitness function employed in our experiments was simply *time to contact*, it is worth asking whether our robots (simulated and real) indirectly optimize this objective. A new set of simulations was run where each individual *was selected* and reproduced according to the usual fitness function described in section 1.2, 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).

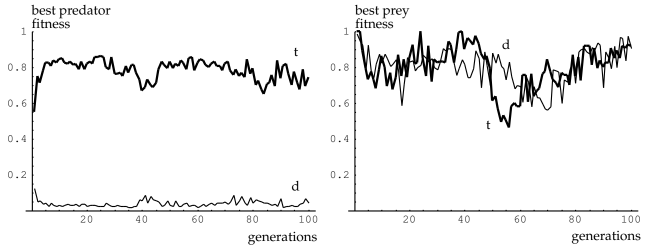

Figure 1.13
Comparisons between fitness of the best individuals measured as *time to contact* (t) and as *distance*. (d). Both species have been evolved using fitness t. **Left**: Best evolved predators do not attempt to minimize distance (predator fitness is 1 - d). **Right**: Best evolved prey attempt to maximize distance (prey fitness is d).

The fitness values computed according to the two methods (figure 1.13) 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 one might expect. On the other hand, in general prey attempt to maximize the distance. 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 1.11 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.

These results indicate that competitive co-evolution can discover innovative and unexpected –yet efficient– solutions provided that one does not attempt to force it into pre-defined directions. If a co-evolutionary system in externally channelled in certain directions, it might loose its ability to generate adaptivity and explore unpredictable solutions. We might speculate that to the limit, a strongly constrained co-evolutionary system will converge to trivial solutions, if such a solution exist for the specific architecture evolved, or will degenerate in random search.

1.4 Ontogenetic adaptive change

In subsection 1.3 we have mentioned that competitive co-evolution might favor the emergence of ontogenetic adaptive changes (i.e., changes during the “life” of an individual robot) as a way to improve adaptivity. Although most of the co-evolutionary systems described in the literature include some type of ontogenetic changes in the form of small random changes to the parameters, it is difficult to say whether this form of noise plays an important role on the specific dynamics of co-evolving species. Moreover, all the results presented so far are based on single-run studies and do not include statistical comparisons between different adaptation techniques.

The aim of this section is that of presenting initial results on the effect of ontogenetic adaptive changes in co-evolving competing species. In particular, we want to address the following questions: Does protean behavior affect evolutionary dynamics? Do competing species exploit different types of protean strategies, and how does this affect the competitor’s behavior? In the attempt to investigate these issues in very simple settings, we have compared co-evolution of competing species equipped with different types of simple adaptive controllers with results from the experiments described in section 1.3 above. In order to obtained data from multiple runs of the same condition for statistical analysis, all the experiments conducted in this section have been carried out with the software technique described in subsection 1.2.

For sake of comparison, all the neural networks had the same architecture, the same genotype length (5 bits per synapse), and used a comparable

Table 1.1
Genetic encoding of synaptic parameters for each co-evolutionary condition. 1: Genetically-determined controllers; 2: Adaptive-noise controllers; 3: Directional-change controllers.

	Bits for one synapse				
Condition	1	2	3	4	5
1	sign	strength			
2	sign	strength		noise	
3	sign	Hebb rule		rate	

encoding technique. We distinguished three evolutionary conditions, each one corresponding to a different type of controller (Table 1.1). In all conditions, the first bit of each synapse coded its sign (whether excitatory or inhibitory).

In the first condition, which is that described in section 1.2 above, the remaining four bits coded the synaptic strength as a value in the range $[0, 1]$. Since no changes take place during the life of the individuals, let us call this condition *genetically-determined controllers*.

In the second condition, only two bits coded the synaptic strength (again, in the range $[0, 1]$), and the remaining two bits coded the level of random noise applied to the synaptic value *at each time step*. Each level corresponded to the lower and upper bounds of a uniform noise distribution: 0.0 (no noise), ± 0.337, ± 0.667, and ± 1.0. For every new sensor reading, each synapse had its own newly-computed noise value added to its strength (with a final check to level out sums below 0.0 or above 1.0). We shall call this condition *adaptive-noise controllers* because each species can evolve the most appropriate noise level for each synapse (including the possibility of not having noise, which corresponds to condition one).

In the third condition, two bits coded four Hebbian rules and the remaining two bits the learning rate (0.0, 0.337, 0.667, and 1.0). Four variations of the Hebb rule were used: “pure Hebb” whereby the synaptic strength can only increase when both presynaptic and postsynaptic units are active, “presynaptic” whereby the synapse changes only when the presynaptic unit is active (strengthened when the postsynaptic unit is active, and weakened when the postsynaptic unit is inactive), “postsynaptic” whereby the synapse changes only when the postsynaptic unit is active (strengthened when the presynaptic unit is active, and weakened when the presynaptic unit is inactive), and “covariance” whereby the synapse is strengthened if the difference between pre- and post-synaptic activations is smaller than a threshold (half the activation level, that is 0.5) and is weakened if the difference is larger than such threshold. After decoding a genotype into the corresponding controller, each synapse was randomly initialised to a value in the range $[0, 1]$ and modified at each time step according to the corresponding hebbian rule and learning rate. In another article, we have shown that this evolutionary scheme in a single-agent static environment can develop stable controllers which quickly develop navigation strategies starting from small random synaptic strengths [10]; interested readers will find more details in that paper. Flotzinger replicated those results (from a previous preliminary report [9]) and studied in more detail the synaptic dynamics, showing that continuously changing synaptic values reflect to a certain approximation input and output states of the controller [13]. Therefore, let us call this condition *directional-change controllers*, simply indicating that synaptic changes depend on sensory activation and motor actions.

Experimental results

For each condition, six different evolutionary runs were performed, each starting with a different seed for initializing the computer random functions. A set of pairwise two-tail *t*-tests of the average fitness and best fitness values along generations among all the six runs, performed to check whether different seeds significantly affected the experimental outcomes, gave negative results at significance level 0.05. Therefore, for each condition below, we shall plot only data referring to seed 1 (arbitrarily chosen), but the statistical tests reported will be based on all the runs.

In order to compare the results between the three co-evolutionary conditions, a *relational measure* of performance was developed. It consisted in computing an index of *relative performance* r_i^c by counting how often one species reports higher fitness than the competing species at each generation for each separate run i (where i stands for a replication of the same experiment with a different random seed) in a specific condition *c*. In our co-evolutionary runs which lasted 100 generations, such index will be in the range $[-100, 100]$, where -100 means that prey always outperformed predators, 0 means that both species were equally better or worse than the competitors, and 100 means that predators always outperformed the prey in 100 generations. As compared to the Master Fitness, this value does not tell us whether there is progress in the population, but it allows us to compare co-evolutionary coupled dynamics between different conditions.

In the condition of genetically-determined controllers $c = 1$ (data given in section 1.3), the average value over six repeated runs is $\overline{R^1} = 16.67$ with standard deviation of the sample mean $\sigma = 38$, indicating that the two species

species did not significantly differ in the number of wins and losses.

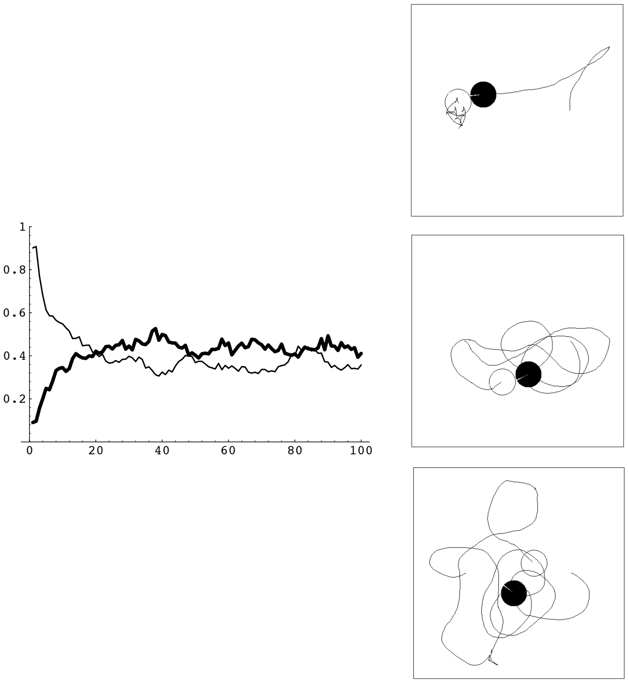

Figure 1.14
Adaptive-noise controllers. **Left**: Average fitness across generations for predator (thick line) and prey (thin line). **Right**: Examples of behaviors (black disk is predator, white disk is prey). Top: generation 20. Center: generation 50. Bottom: generation 80.

The condition with evolutionary adaptive noise ($c = 2$) displayed an average relative performance $\overline{R^2} = 11.66$ with standard deviation of the sample mean $\sigma = 32.5$ which was not statistically different from that of the condition of genetically-determined controllers (probability value was 0.83 for t-test of the difference of the means between the two conditions, i.e. much bigger than significance level 0.05 typically used for rejecting the equality hypothesis). The oscillatory patterns observed on the fitness values for condition 1 took place in condition 2 too, but were much smoother (figure 1.14, left). Furthermore, in all cases it took roughly twice as many generations –as compared to condition 1– to lock into oscillatory dynamics (marked by the generation when the two fitness lines cross). We then compared between the two species the average noise level used by all individuals in the population and by the best individuals at each generation. Two separate t-tests for measuring differences of average noise level and of noise level of the best individuals both displayed a significant difference ($p \ll 0.1$). Prey clearly used much higher noise levels than predators.

The hypothesis that the prey exploited noise to develop unpredictable controllers (that is, not improving much on initial random controllers) while the predator tried to develop more stable pursuit strategies was reflected by the behaviors of the two species (figure 1.14, right). Prey displayed unpredictable manoeuvres while the predators attempted to track them. The prey trajectory was often changing while retaining sufficient obstacle-avoidance abilities (it sometimes stopped near a wall for a few instants and later moved away). The predator behaviors were more predictable. In general, they were sufficiently good at keeping the prey within the visual field.

Relative performance of the two species in the third condition, (directional-change controllers) significantly differed from condition 1 (and from condition 2). Relative performance was $\overline{R^3} = 72$ with standard deviation of the sample mean $\sigma = 15.39$, $p < 0.01$ for a two-tailed t-test of the difference of the means. In all six repeated runs predators reported higher average and best fitness values than prey, except for short temporary oscillations (figure 1.15). In all runs, the average fitness of the predator population was more stable than that of prey.

Predators always display very good chasing abilities across generations: once the prey has been locked in its visual field, it quickly accelerates to maximum speed until contact. As a matter of fact, for the predator it is sufficient to get the sign of the synapses right. Then, independently of their initial random values, the synapses from active sensors will be increased causing an acceleration in the right direction. As compared to condition 1, where predators tended to efficiently track only in one direction, here they can turn in both directions at equal speed depending where the prey is. In condition 1 proper tracking in both directions would have required accurate settings of all synaptic strengths from visual inputs. Here, instead, since synapses are temporarily increased depending on active units [9, 13], individual adjustments of synapses take place when and where required depending on current sensory input. The trajectory displayed in the center image of figure 1.15 shows another example of synaptic adjustment. While the prey rotates always around the same circle, the predator performs three turns during which synaptic values from the visual units

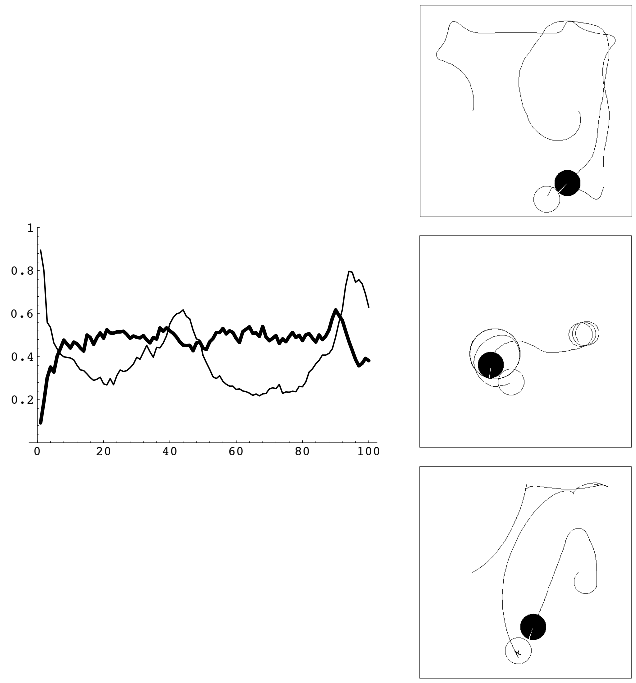

Figure 1.15
Directional-change controllers. **Left**: Average fitness across generations for predator (thick line) and prey (thin line). **Right**: Examples of behaviors (black disk is predator, white disk is prey). Top: generation 20. Center: generation 70. Bottom: generation 95.

are gradually increased; at the fourth turn, the synaptic values are sufficiently strong to initiate a straight pursuit (eventually, the prey will try to avoid the predator without success). The temporary drop in performance of the predator after generation 90 is due a more precise tracking combined with a slower motion (bottom image of figure 1.15). Such behavior was probably developed because prey were also slower and more careful in avoiding obstacles (including the predator). Although activity-dependent synaptic change are exploited by the far-sighted predator, not the same happens for the prey. Prey move faster than in conditions 1 and 2, especially when turning near walls (where IR sensors become active and synapses temporarily strengthen), but they cannot increase their behavioral repertoire with respect to condition 1. Not even can they improve it because volatile changes of the synaptic values imply that most of the time they must re-develop on-the-fly appropriate strengths; although this is alright for avoidance of static obstacles, it is a problem when facing a fast-approaching predator.

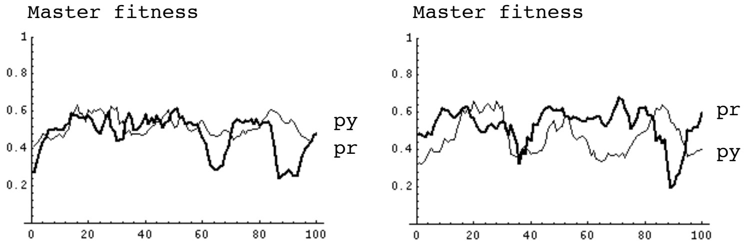

Figure 1.16
Left: Master Tournament between species co-evolved in condition 1 (genetically-determined controllers). **Right**: Master Tournament between predators evolved in condition 3 (directional-change controllers) and prey evolved in condition 1 (right). In the latter case, predators win more often. See text for further implications of these results.

In order to check whether predators' superior performances in condition 3 were due to a real advantage of the predator rather than to some difficulties of the prey to cope with directional-change controllers, we compared Master Fitnesses for predators and prey co-evolved in condition 1 (figure 1.16, left; replicated from figure 1.8 for sake of comparison) with Master Fitnesses for predators evolved in condition 3 againts prey evolved in condition 1 (figure 1.16, right). Directional-change predators win more often against genetically-determined prey ($\overline{R} = 42$) than genetically-determined predators do. If the advantage reported by predators co-evolved with directional-change prey had been caused only by difficulties of prey to evolve suitable directional-change controllers, the Master Tournament between species evolved in different conditions (figure 1.16, right) should have not generated differential performances.

Finally, we run a new set of experiments where each synapse could decide whether to change according to condition 2 (adaptive noise) or to condition 3 (directional change). The genetic code of all synapses of both species was augmented by one bit. If this bit was on, the previous four bits were interpreted as described for condition 2, otherwise as described for condition 3 (notice that the first bit always codes the sign of the synapses in all conditions). The results clearly indicated that predators always won over the prey (figure 1.17,

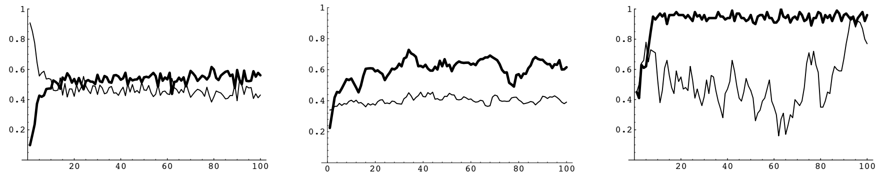

Figure 1.17
Co-evolution of controllers that can exploit either adaptive noise or directional change (thick line = predator; thin line = prey). **Left**: Average population fitness recorded during co-evolution. **Center**: Master Fitness. **Right**: Amount of directional-change synapses in the population. Figures are obtained by counting the number of directional-change synapses per individual (divided by total number of synapses) and further averaging over all individuals in the population.

left) and Master Fitness values even revealed a slightly ascending trend for predators, but not for prey (figure 1.17, center). Analysis of the genotypes showed that all predators consistently evolved directional-change synapses, but prey oscillated between preference for adaptive noise and directional change without any consequence on the outcome of the tournaments that were almost always won by predators.

These results indicate that directional-change provides superior adaptivity with respect to the other two conditions only if coupled with the sensory abilities of the predator. The addition of visual information provides long-range information that is suitable for fast adaptation of chasing strategies. On the other hand, the limited sensory information of the prey cannot capitalize on this form of ontogenetic adaptation. The best the prey can do is to maximize behavioral unpredictability and diversity.

1.5 Conclusive discussion

The experiments described in this article indicate that competitive co-evolution can generate very quickly efficient chasing and escaping strategies in the two species. Consider for example that a simple chasing strategy (consisting in locating the prey, going towards it, while avoiding the walls) emerged in less than half generations required for evolving a simple obstacle avoidance behavior for the same robot and a similar control architecture in standard evolution (single robot in static environment) [8]. This might be explained by the fact that co-evolution, at least in the initial phases, is analogous to incremental evolution. In other words, competitive co-evolution generates solutions to difficult problems by generating a series of challenges of increasing difficulty. This is reflected by a posteriori analysis of the modification of the fitness landscape described in subsection 1.3. However, after some generations the two populations re-discover old behaviors and endlessly alternate between them. In a companion paper [29] we show that this apparent problem can be solved in two ways: either by testing each individual against best competitors extracted from all previous generations (Hall-of-Fame method), or by making the environemnt and sensory system of the two species more complex.

The point is that alternating between behavioral strategies is not a problem, but a smart solution for coping with competitors that rapidly change their strategy. The alternative to this result would be to evolve behaviors that display greater and greater generality, i.e. behaviors that can defeat a larger number of opponents. This is possible only if such behaviors exist or are not too rare on the space of all possible behaviors that can be generated with a given control architecture. If such behaviors do not exist, the best that co-evolution can do is to evolve individuals that can quickly re-adopt a previous strategy as soon as the current strategy is no longer successful (because the opponent has changed his behavior). This implies that co-evolution finds individuals that sit on areas of the genotypes characterized by steep peaks, that is individuals whose mutants are significantly different, but still viable and successful. This observation bears analogies with the hypothesis of "life at the edge of chaos" advanced by Langton [22].

We think that it is necessary to reconsider artificial evolution in a new perspective, different from mainstream evolutionary computation oriented towards function optimization (see also [18] for a similar view). In unpredictably changing environments evolution of optimal solutions is not an appropriate approach because optimality cannot be defined a priori. A more suitable metaphor is *evolution of adaptivity*, that is evolution of solutions that can cope with rapidly changing environments. We argue that artificial evolution should strive for evolving adaptive solutions, rather than solution for a fixed set of predefined problems, if we want to evolve and understand artifacts that display life-like properties. In section 1.3 above, we have outlined at least three ways in which artificial evolution could achieve increased adaptivity: *a)* generation and maintainance of diversity (which gives higher probability that some individuals will be able to reproduce in changed environments); *b)* selection of genotypes whose mutant neighbors correspond to different but all equally viable phenotypes (which allows rapid generational switch between different strategies); *c)* exploitation of ontogenetic adaptation (which can quickly cope with environmental changes without requiring genetic changes).

In this research we have explored in more detail the third option (and only given some hints about the second option which is further discussed in another article [29]). Co-evolving predators consistently exploit the possibility of ontogenetic synaptic change with Hebbian rules and report higher performance over the prey in three different conditions: against prey co-evolving under the same directional-change condition, against prey co-evolved under genetically-determined condition, and against prey that are free to evolve under adaptive-noise or directional-change conditions. In all cases ontogenetic adaptation affects co-evolutionary dynamics by reducing the amplitude and frequency of oscillations in fitness values. This suggests that ontogenetic changes can cope with modifications of prey strategies without requiring much philogenetic change. For example, it was shown that rapid synaptic adaptation allows the network to ontogenetically switch between different strategies (chase to the left or chase to the right, depending on the prey behavior) that would otherwise require generational evolution given this specific control architecture.

The prey, given their limited sensory information, did not benefit from the types of ontogenetic changes considered in this research. However, when co-evolved in adaptive-noise conditions, they consistently increased the level of synaptic noise significantly more than predators. This corresponded to unpredictable, rapidly changing, manoeuvres. Also in this condition the amplitude and frequency of oscillations of fitness values was decreased. In fact, by displaying a continuously changing behavior, the prey offered the predators a larger set of different behaviors forcing them to adopt more general behavioral solutions. These latter results highlight the strong relationship between physical aspects of the machine and the type of learning system adopted. Ideally, sensory motor morphologies and learning algorithms should be co-evolved.

The experiments described in this article have been carried out (partly) on physical robots and (mostly) on a realistic software model in order to quickly explore different hypotheses and obtain repeated measures for statistical analysis. The fact of having co-developed hardware and software has been a determining factor for several decisions that had to be taken early on during the preparation of the experimental set-up. The most important decision regarded the formulation of the fitness function. Considering that our robots could not reliably measure the distance between each other, we resorted to a simpler and more general fitness function based on time to contact. By "simple function" we mean a function that does not force evolution towards predefined behaviors based on human preconceptions derived from a *distal description* of the robot behavior.[5] In this case, we did not ask predators to chase by minimizing distance and prey to escape by maximizing it, as other authors did (e.g., see [3]). This was only one of the possible options that the two co-evolving species could adopt. The analsyis described in section 1.3 showed that predators indeed adopted different and more subtle strategies. Whether or not the choice of a selection criterion is so crucial to significantly affect the global co-evolutionary dynamics [6], constraining co-evolution reduces the amount of emergent behaviors that both populations could potentially discover. Within the context of artificial evolution, intended as an open-ended endeavour to develop increased levels of adaptivity, simpler and more intrinsic fitness functions leave more space to autonomous self-organization of intelligent forms of artificial life.

Acknowledgements

This chapter is the reprint of an article published on the journal *Robotics and Autonmous Systems*, vol. ??, pp. ??, 1999.

References

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

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

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

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

[5]P. Driver and N. Humphries. *Protean behavior: The biology of unpredictability*. Oxford University Press, Oxford, 1988.

[6]S. Ficici and J. Pollack. Coevolving Communicative Behavior in a Linear Pursuer-Evader Game. In R. Pfeifer, B. Blumberg, J-A. Meyer, and S. Wilson, editors, *From Animals to Animats V: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*. MIT Press-Bradford Books, Cambridge, MA, 1998.

[7]D. Floreano. Ago Ergo Sum. In G. Mulhauser, editor, *Evolving Consciousness*. Benjamins

[5] See [32, 27] for a distinction between proximal and distal descriptions of behavior and its implications for design and analsyis of autonomous systems.

Press, New York, To appear.

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

[9]D. Floreano and F. Mondada. Evolution of plastic neurocontrollers for situated agents. In P. Maes, M. Matarić, J-A. Meyer, J. Pollack, H. Roitblat, and S. Wilson, editors, *From Animals to Animats IV: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*, pages 402–410. MIT Press-Bradford Books, Cambridge, MA, 1996.

[10]D. Floreano and F. Mondada. Evolutionary Neurocontrollers for Autonomous Mobile Robots. *Neural Networks*, 11:1461–1478, 1998.

[11]D. Floreano and F. Mondada. Hardware Solutions for Evolutionary Robotics. In P. Husbands and J-A. Meyer, editors, *Proceedings of the first European Workshop on Evolutionary Robotics*. Springer Verlag, Berlin, 1998.

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

[13]D. Flotzinger. Evolving plastic neural network controllers for autonomous robots. Msc dissertation 9580131, COGS, University of Sussex at Brighton, 1996.

[14]P. Funes, E. Sklar, H. Juillè, and J. Pollack. Animal-Animat Coevolution: Using the Animal Population as Fitness Function. In R. Pfeifer, B. Blumberg, J-A. Meyer, and S. Wilson, editors, *From Animals to Animats V: Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior*. MIT Press-Bradford Books, Cambridge, MA, 1998.

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

[16]J. L. Gould. *Full House: The Spread of Excellence from Plato to Darwin*. Random House (2nd Edition), New York, 1997.

[17]S. J. Gould and R. C. Lewontin. 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, 1979.

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

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

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

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

[22]C. G. Langton. Life at the edge of chaos. In C.G. Langton, J.D. Farmer, S. Rasmussen, and C. Taylor, 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, 1992.

[23]A. J. Lotka. *Elements of Physical Biology*. Williams and Wilkins, Baltimore, 1925.

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

[25]G. F. Miller and D. Cliff. Protean behavior in dynamic games: Arguments for the co-evolution of pursuit-evasion tactics. In D. Cliff, P. Husbands, J. Meyer, and S. W. Wilson, editors, *From Animals to Animats III: Proceedings of the Third International Conference on*

Simulation of Adaptive Behavior. MIT Press-Bradford Books, Cambridge, MA, 1994.

[26]J. D. Murray. *Mathematical Biology*. Springer Verlag, Berlin, 1993. Second, Corrected Edition.

[27]S. Nolfi. Using emergent modularity to develop control system for mobile robots. *Adaptive Behavior*, 5:343–364, 1997.

[28]S. Nolfi. Evolutionary robotics: Exploiting the full power of self-organization. *Connection Science*, 10:167–183, 1998.

[29]S. Nolfi and D. Floreano. Co-evolving predator and prey robots: Do “arms races” arise in artificial evolution? *Artificial Life*, 4:30–55, 1998. page numbers not exact.

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

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

[32]N. E. Sharkey and N. H. Heemskerk. The neural mind and the robot. In A. J. Browne, editor, *Current perspectives in neural computing*. IOP Publishing, London, 1997.

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

[34]L. van Valen. A new evolutionary law. *Evolution Theory*, 1:1–30, 1973.

[35]V. Volterra. Variazioni e fluttuazioni del numero di individui in specie animali conviventi. *Memorie dell’Accademia dei Lincei*, 2:31–113, 1926. 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.