NEUROEVOLUTION OF AUTO-TEACHING ARCHITECTURES

EDWARD ROBINSON & JOHN A. BULLINARIA
School of Computer Science, University of Birmingham
Edgbaston, Birmingham, B15 2TT, UK
e.robinson@cs.bham.ac.uk

This paper explores the idea that auto-teaching neural networks with evolved self-supervision signals can lead to improved performance in dynamic environments where there is insufficient training data available within an individual's lifetime. Results are presented from a series of artificial life experiments which investigate whether, when, and how this approach can lead to performance enhancements, in a simple problem domain that captures season dependent foraging behaviour.

1. Introduction

Neural networks with reinforcement or supervised learning algorithms are known to be excellent at acquiring complex behaviours if they are given a large enough amount of training data. However, humans and other animals often need to operate in environments in which there is insufficient information available to enable an individual to configure their behaviour mechanisms appropriately. This can render such lifetime learning techniques inadequate, despite their generally good potential.

Learning, however, can take place over two rather different time-scales: lifetime adaptation during the life of an individual, and evolutionary adaptation over much longer periods. If the learning task remains constant across many generations of evolution, then it is possible that the necessary behaviours could become specified innately, making the need for lifetime learning unnecessary. If lifetime learning has a cost, such as periods of relatively poor performance, then it is best avoided if possible. It is certainly known that the Baldwin Effect can lead to the genetic assimilation of appropriate learned behaviours (Baldwin, 1896; Bullinaria, 2001). However, if the behaviour to be learned changes within the individual lifetimes, or varies between individual environments, or changes too quickly for evolution to track it, or is too complex for easy genetic encoding, then evolution will not be able to help in this way.

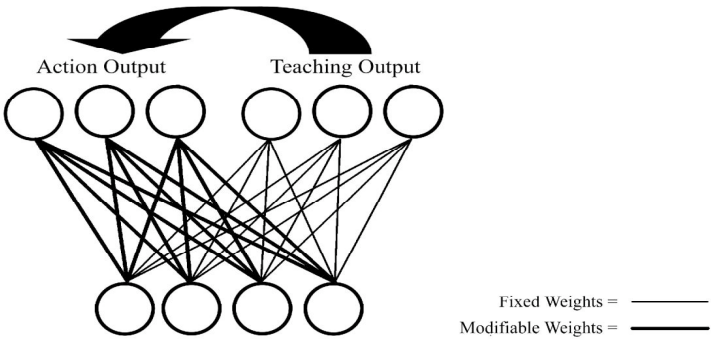

Figure 1: Simple auto-teaching neural network architecture.

Another approach that has been suggested for resolving the problem of insufficient training data during a lifetime involves employing *auto-teaching networks*. These are neural networks that can generate their own "supervision signal", based on evolved innate connection weight parameters (Nolfi & Parisi, 1993, 1996). The idea is that evolution can lead to neural structures that do not completely encode particular behaviours themselves, but instead they facilitate the lifetime learning of appropriate behaviours. Previous research in this area has shown, using a number of different learning tasks, how simulated individuals using these teaching architectures can out-perform similar individuals that do not have an "auto-teaching" mechanism.

The previous experiments involved simple artificial life agents evolved to perform a foraging task. Their neurocontrollers had a feed-forward architecture, in which the inputs were environmental stimuli and the outputs specified behaviour in the environment. This paper presents further simulations following a similar approach, but explores the evolution of auto-teaching networks more systematically, and assesses the performance of this approach on new and more difficult tasks in a novel dynamic environment.

The remainder of this paper will begin with a review of past work on auto-teaching networks, and then describe in detail how the new simulations were set up. This is followed by a presentation of the key simulation results, and some discussion and conclusions.

2. Auto-Teaching Neural Networks

Auto-teaching networks have traditional connection weights w_{ij} that are updated during the individual's lifetime, but instead of the usual external training signal, which will not be available in many realistic scenarios, they have an internal training signal coming from an additional set of neurons driven by the same

sensory inputs via evolved fixed teaching weights W_{ij} (Nolfi & Parisi, 1993). A simple feed-forward auto-teaching network architecture is shown in Figure 1. For each input pattern, the outputs are computed as in a standard feed-forward network, but rather than updating the weights w_{ij} to minimize the output errors relative to externally provided target outputs, they are updated using targets provided by the internal teaching outputs. The idea is that evolution provides an appropriate teaching signal depending on the current sensory inputs, as a substitute for the feedback information lacking in the environment.

The aim of this paper is to explore whether, when and how this architecture performs better than traditional networks. Nolfi & Parisi (1993) devised a simple foraging task within a 'grid-world' artificial life environment to do this, and used a genetic algorithm to evolve agents that were good at finding food items to consume. They used a multi-layer perceptron with sensory inputs that provided the direction and distance to the nearest food item, and action outputs that specified the agent's next movement. After each movement, the teaching outputs were used to update the modifiable weights using a standard gradient descent algorithm. Nolfi & Parisi (1993) presented results indicating that their auto-teaching networks performed better than networks in which the main network weights were evolved but fixed during an individual's lifetime. However, Williams & Bounds (1993) analysed a similar task and found that simple perceptrons performed better than the more complex multi-layer networks with auto-teaching architectures.

The environment used by Nolfi & Parisi (1993) had randomly distributed food items, but was of the same form for each new agent, and static during an agent's lifetime. It seems likely that, for any predictable static environment, any useful search policy that evolves to be encoded in fixed network weights would remain suitable throughout each agent's life, and there will consequently be no advantage to having an auto-teaching mechanism too. Whether auto-teaching might usefully interact with evolution to facilitate the acquisition of better evolved behaviour in such circumstances is less clear (Nolfi & Parisi, 1993; Williams & Bounds, 1993).

Nolfi & Parisi (1997) designed a more challenging changing environment, and used simple perceptrons with auto-teaching to control simulated Kherpera robots. These robots had to move to a randomly pre-defined target in a finite environment, and to do well (i.e. complete the task quickly) the robots had to explore the environment as efficiently as possible. However, there were two distinct environments, and each new agent was placed randomly into one of them, which meant that two distinct search policies were needed. Evolved robot controllers that could auto-teach during their lifetime were found to outperform

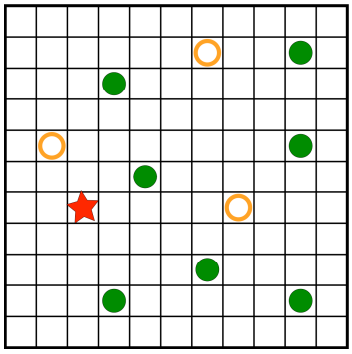

Figure 2: Grid-world environment, with agent ★, food ● and buried food O.

those with fixed weights, because they could adapt to the right policy rather than rely on some inferior compromise policy. In general, it is easy to see how limited sensory information could be used with auto-teaching to switch at birth to the appropriate one of many evolved behaviours that have been acquired though exposure to many potential environments over evolutionary history.

What remains to be explored is the dynamic case in which the form of the environment is the same for all agents, but its nature changes during the agent's lifetime, possibly many times.

3. Simulations with Dynamic Environments

To explore auto-teaching in dynamic environments, agents were simulated that had to forage for randomly placed food items in a grid-world environment of the form shown in Figure 2, as in the Nolfi and Parisi (1993) study. Adaptation is required because the availability of food varies throughout the agents' lifetimes. Each time-step is one agent *day*, and 600 days is one agent *year* made up of two *seasons*: first summer with plenty of food items, and then *winter* with none (Figure 3). Agents must bury some food during the summer to consume later to survive through the winter. Simulation results will be presented for a 20×20 cell world, with agents that start with 100 energy units and expend energy at the rate of one unit per time-step. Each food-item consumed is worth another 35 energy units, but agents can only store 100 units of energy at any time. Given appropriate sensory information, this task is linearly separable (Williams & Bounds, 1993), so a simple perceptron controller was adequate.

At each time-step, the agents' simple perceptron networks (as in Figure 1) have their weights and activations updated and then have their output actions implemented. The agents had different sensory input information available depending upon the experimental condition. They always had the normalized

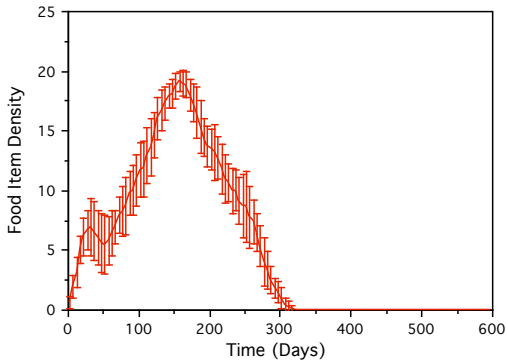

Figure 3: Density of food items in the environment at different times of year.

coordinate distances to the nearest food item. In the first set of simulations, they also had a sinusoidal representation of the time of year (i.e. the season), and their current normalized energy level, but these were absent in the second set of simulations. Each of the agents' three output unit activations specified one of their three potential actions: namely move left/right, move up/down, and consume/bury food. These outputs were all thresholded, so that activations in the range [–0.2, +0.2] meant no action, while higher or lower values meant unit action in the appropriate direction. The consume/bury action always takes precedence over the movements. If an agent is situated in the same cell as a food-item, and the consume/bury output neuron has output +1/–1, the agent will consume/bury the food-item, and not move. Only if the consume/bury output is 0, or the agent is not situated on a food-item, will the agent perform the action corresponding to its movement outputs.

In line with the previous studies described in Section 2, an evolutionary algorithm was used to evolve populations of fit agents. The genotype to be evolved consisted of the network connection weights and biases, and, when lifetime auto-teaching was allowed, the learning rate. For tasks with n input units and m output units, that means $(n+1)m$ evolvable parameters for the fixed networks and $2(n+1)m + 1$ for the auto-teaching networks – consisting of the $(n+1)m$ main initial weights w_{ij}, $(n+1)m$ auto-teaching weights W_{ij}, and one learning rate. At the beginning of each simulation, a population of 100 random individuals were generated, with network weight values taken from a uniform distribution with the range [–1, +1]. Fitness was taken to be the length of time survived, which provided a measure of the agent's foraging skills. To smooth the statistical variations inherent in the simplified task, each agent's fitness was averaged over three independent runs. A steady state genetic algorithm with tournament selection was used, replacing the least fit fraction of the population

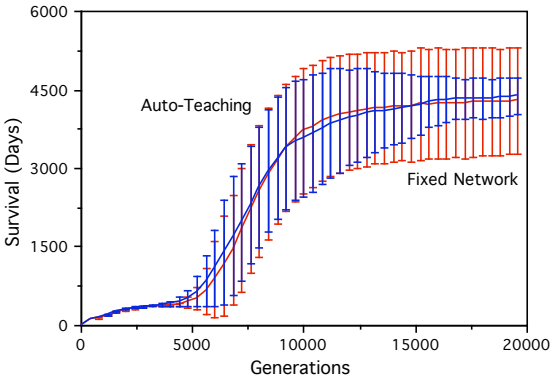

Figure 4: Evolution of performance when all the sensory information is available.

at each stage by children of the fitter individuals (Eiben & Smith, 2003). A recombination process produced the children by crossover and mutation. Crossover randomly took parameters from both parents, but with an increased probability (of 66%) that sets of weights corresponding to a single output were kept together. That minimized the disruption to existing correctly functioning output units. Then, with 1% probability, an individual would have one of its parameters randomly mutated with a Gaussian distribution. This process was repeated over many generations until the performance level stabilized.

4. Simulation Results and Analysis

Three distinct sets of experiments were run to elucidate the advantages, if any, of auto-teaching networks in the artificial life environments described above.

4.1 Simulations With Full Set of Sensory Information

First the performance was evaluated for auto-teaching architectures that had all the sensory-information available to them: coordinate distances to nearest food-item, time of season, and internal energy levels. Although this environment is different to that used by Nolfi & Parisi (1993), it is qualitatively similar in that the agents have all the information necessary to perform well on the task. Thus it can again be asked whether auto-teaching networks provide an advantage over standard fixed neurocontrollers.

Figure 4 presents the survival times, with means and variances over twenty evolutionary runs, with foraging episodes of up to 5000 days. This shows that there is no statistically significant increase in foraging performance when auto-teaching networks are employed on this task. Contrary to the Nolfi & Parisi (1993) study, it is found here that, in this case where the task is linearly

separable, a good behavioural policy can be specified via evolution alone, with simple fixed perceptrons performing just as well as those that adapt during their lifetimes, and there is no difference in the rate of evolution. The overall lifetime performance will always be better when the agents can behave correctly from birth, rather than at some later point in life after learning, so it follows that auto-teaching networks are of no benefit here.

Observing the behaviour of individual agents confirmed that those with fixed networks possess the correct behavioural policy from birth. It was useful to determine, at this point, exactly how evolution had specified the ability to solve the task. The agents could easily evolve the capacity to quickly approach food via the most optimum path, because the available distance information inputs trivially indicate the direction to move at each stage. How the agents evolved to deal with the issue of when to bury or consume food-items, or indeed when to neither bury nor consume them, was not so straightforward. It was found that the best evolved agents used a combination of time-of-year and internal-energy sensory inputs to decide on the appropriate action at each time-step. The main criterion for what an agent should do with a food-item was the time of season. During the winter months, because of the lack of new food-items available for burying, agents only consumed food-items. During the summer months, agents only buried items, unless the winter period was imminent or their internal energy became low. This meant that they generally maximised the number of food-items buried during the summer months, and only consumed food-items when necessary. Another use for the internal-energy sensory input was to maximise the length of time during the winter months when the agent was not consuming its previously buried food-items. Agents did this by remaining still until their energy became low, in which case they headed to a previously buried food-item and consumed it.

4.2 Simulations With Limited Sensory Information

The first objective of this work has been achieved by establishing, in this scenario, that auto-teaching networks do not always perform better than fixed networks in artificial-life environments. Next the simulations were varied to determine under what conditions auto-teaching networks can outperform fixed evolved networks. Analysis of the experimental results so far indicated that the evolved agents relied heavily on the internal-energy and seasonal-time sensory inputs. With these two sensory stimuli, a fixed-network can display a wide enough range of actions to survive indefinitely in the environment. However, if these stimuli were removed, the agents would need another mechanism to change their behaviour during their lifetime in order to survive longer than the

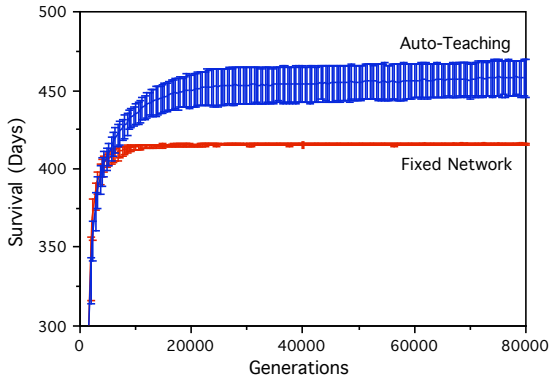

Figure 5: Evolution of performance when only food location input is available.

initial summer months. This is because, with only the position information available, an agent's network input will always be the same whenever it is positioned on a given food-item. Therefore, for the network outputs to be different, e.g. –1 for bury or +1 for consume, the network connection weights must be different, which means the agent needs to adapt during its lifetime.

However, given the nature of the neural learning mechanism being used, it is not obvious whether the necessary adaptation is possible, given that the external teaching signals are limited, and the way an agent teaches itself is fixed during its life (because the teaching weights are fixed). In particular, the agents must use the environment to generate appropriate auto-teaching stimuli, and if the environmental stimuli are limited to the position information of the nearest food-item, it is not clear whether that can lead to appropriate consume/bury signals. To explore this issue, the simulations were re-run with only the food location information as sensory inputs. The results, presented in Figure 5, show that all the performance levels are considerably lower than the previous experiment, due to the increased difficultly of the task resulting from the lack of sensory stimuli. However, there is now a clear and significant improvement in the performance of the auto-teaching networks over fixed networks.

The fixed networks evolve to simply consume food-items whenever they find them, and that allows them to survive until shortly after the end of summer-time, i.e. ~410 simulated days. It is clear from the improved mean performance of ~450 days that agents with an auto-teaching architecture are changing their behaviour (with regard to what action to perform when on a food-item) during the task. They are now, at birth, burying food-items whenever they encounter them. However, at some point during their life, the agents change their behavioural policy to one that involves consuming the food-items whenever they are encountered. This allows them to bury some food-items to

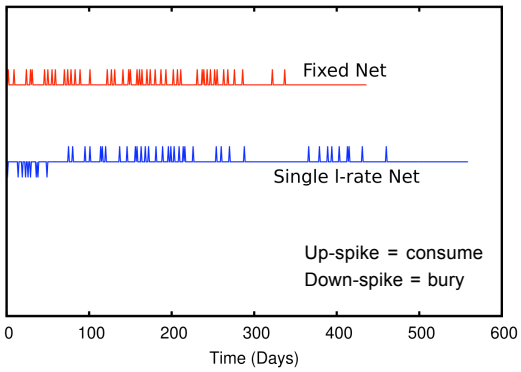

Figure 6: Actions taken by typical evolved agents during the foraging task.

be consumed during the winter part of the task when there are no other food-items available. Since the agents no longer have any sensory information about the time of year, nor their internal energy levels, the evolutionary process is managing to select for teaching-weights that ensure this behaviour change happens at the most appropriate point in time, given the stochastic nature of the task. Figure 6 illustrates how these behaviours take place in typical evolved agents, with the up/down spikes representing successful consume/bury actions. This graph also illustrates the difference in resultant life-times.

From these results, it is clear that the auto-teaching networks do not evolve sophisticated continuous learning – the agents only change their consuming/ burying behaviour once during their lifetime. To do better at the task, they would need to be able to switch between the two behaviours multiple times during a lifetime spanning many years. In effect, the evolved auto-teaching mechanism needs to encode the fact that there are seasons of particular lengths, and adjust the agents' behaviours at appropriate times of each year. The current framework is such that evolution cannot find such a mechanism, probably because the sensory inputs provide insufficient information to allow it to take place with the learning mechanisms available.

4.3 Simulations With Multiple Learning Rates

To give the auto-teaching mechanism more flexibility, and to provide evolution with a better chance of finding improved behaviours, more sophisticated neural networks were considered. The auto-teaching signal provides target outputs, and the single evolved learning rate specifies the rate at which the adaptable weights change to achieve those targets. If numerous different rates of change were allowed in the network, it is likely that a wider range of dynamics would be evolvable. Consequently, the previous approach was extended to allow the

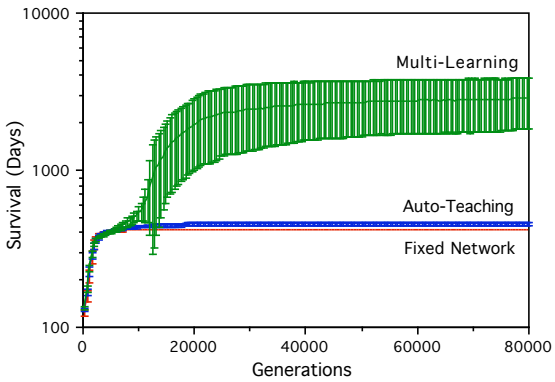

Figure 7: Evolution of performance for multiple learning rate agents.

possibility of a different learning rate to evolve for each adaptable network weight. This would allow (if it proved beneficial) independent weights to change at different speeds during the life of an agent, possibly with some being fixed, and this could allow more complex behaviours to emerge.

The previous simulations were thus repeated with multiple learning rates. For networks with n input units and m output units, that meant increasing the size of the agent genotypes to $3(n+1)m$ – consisting of $(n+1)m$ initial values for the adaptable weights w_{ij} and $(n+1)m$ auto-teaching weights W_{ij} as before, but now with $(n+1)m$ learning rates instead of only one. All the other simulation details remained exactly the same as before, with a similar pattern of crossover and mutation employed for the learning rates as for the network weights. Figure 7 shows how the new multiple learning rate auto-teaching networks significantly outperform both the previous single learning rate auto-teaching architectures and the non-learning fixed networks.

The new multiple learning rate architectures are able to modulate their behaviour during life, allowing them to change between consuming and burying policies multiple times. It is clear, from the average population fitness, that agents using this new architecture are surviving well beyond a single simulated year, which is encouraging because the previous auto-teaching networks were not capable of this. In fact, the maximum foraging time of 5000 days is once again being regularly achieved, and it is now actually possible for agents to survive indefinitely in the environment.

The season dependent behaviour is seen more clearly in Figure 8, which presents a plot of a typical agent's actions when encountering a food-item, for their first 2500 days. Analyzing the details of the evolved networks reveals that there are actually many different mechanisms that can emerge to enable this improved behaviour. The auto-teaching signals always vary rapidly during the

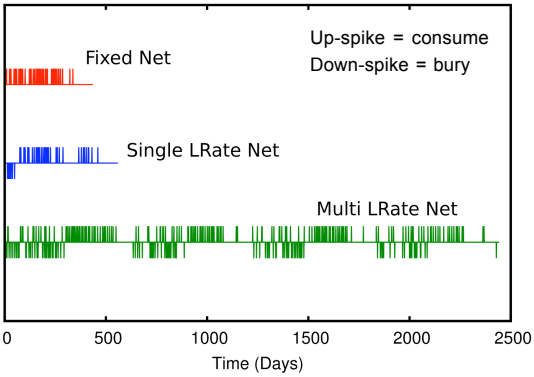

Figure 8: Actions taken by typical evolved multiple learning rate agents.

agents' lifetimes, and these interact with complex patterns of learning rates to modulate the rates of consuming and burying at different times of each year. Exactly how this is achieved is found to vary considerably across individuals from independent evolutionary runs, with different sub-sets of the weights w_{ij} becoming fixed in different individuals. It is certainly clear from Figure 8 that the evolved behaviour is more complex than the simple switch between burying and consuming at appropriate times each year that one might think would be the easiest way to survive over many years. Burying only takes place during the summers, but apart from that, the choice between burying and consuming appears random. Determining the precise range of mechanisms that allow this improved behaviour will require considerable further study. The important issue here is that, within this relatively simple framework, it is possible for evolution to reliably find such mechanisms.

5. Discussion and Conclusions

The study presented in this paper has used artificial life simulations, in a simple foraging domain, to test the effectiveness of auto-teaching neural networks for agents operating in dynamic environments. It showed that, contrary to previous indications in the literature, simpler fixed weight networks can exhibit equally good performance on some tasks. However, if the sensory input information is sufficiently limited on the same tasks, then auto-teaching networks can show significant performance improvements over fixed networks. In general, if the task is predictable, but varies during an individual's lifetime, it will depend on the nature of the task and the available sensory inputs whether auto-teaching will be beneficial. Fortunately, the evolutionary approach adopted in this paper is easily applicable to determine the efficacy of auto-teaching neural networks for any novel problem domain.

For the simple foraging task studied, the performance enhancement provided by standard auto-teaching neural networks, with a single learning rate for all the modifiable weights, was relatively limited. However, by allowing a slightly more general auto-teaching mechanism, with a potentially different learning rate for each modifiable weight, it was demonstrated that auto-teaching architectures can, in certain scenarios, evolve to modulate behaviour successfully in line with the environmental dynamics, and thus out-perform the standard auto-teaching networks by a large margin.

While it is clear that evolved auto-teaching networks can perform better on the simple foraging task considered than evolved networks with fixed weights, it remains to be seen how useful auto-teaching is for more complex behaviours that require more complex neural networks. For example, for the dynamic task studied here, a neural network sophisticated enough to allow the implementation of a timer or oscillator would quite likely allow the evolution of a fixed network that could allow individuals to survive over many years just as easily as the auto-teaching network. It remains a topic for future work to establish the limits of the auto-teaching approach, and to determine if and when it has advantages over other approaches for other tasks. If auto-teaching allows simpler networks to solve a particular task, it is quite possible that evolution will favour it as a mechanism for implementing the appropriate behaviour.

References

Baldwin, J.M. (1896). A new factor in evolution. *American Naturalist*, **30**, 441-451.

Bullinaria, J.A. (2001). Exploring the Baldwin Effect in evolving adaptable control systems. In R.M. French & J.P. Sougne (Eds), *Connectionist Models of Learning, Development and Evolution,* 231-242. Springer.

Eiben, A.E. & Smith, J.E. (2003). *Introduction to Evolutionary Computing*. Berlin, Germany: Springer-Verlag.

Nolfi, S. & Parisi, D. (1993). Auto-teaching: networks that develop their own teaching input. In: J.L. Deneubourg, H. Bersini, S. Goss, G. Nicolis & R. Dagonnier (Eds), *Proceedings of the Second European Conference on Artificial Life*. Brussels.

Nolfi, S. & Parisi, D. (1996). Learning to adapt to changing environments in evolving neural networks. *Adaptive Behavior*, **5**, 75–98.

Williams, B. & Bounds, D. (1993). Learning and evolution in populations of backprop networks. In: J.L. Deneubourg, H. Bersini, S. Goss, G. Nicolis & R. Dagonnier (Eds), *Proceedings of the Second European Conference on Artificial Life*. Brussels.