Non-Homogeneous Classifier Systems in a Macro-Evolution Process

Claude Lattaud
Laboratoire d'Intelligence Artificielle de Paris5
UFR Math-Info
Université René Descartes
45, rue des Saints-Pères
75006 Paris
Lattaud.Claude@math-info.univ-paris5.fr

Abstract

The synthesis of artifacts reproducing behaviors and properties of living beings is one of the main goals of Artificial Life. These artificial entities often evolve according to algorithms based on models of modern genetics. Evolutionary algorithms generally produce micro-evolution in these entities, by mutation and crossover applied on their genome. The aim of this paper is to present Non-Homogeneous Classifier Systems, NHCS, integrating the process of macro-evolution. A NHCS is a derived type of classical classifier systems, CS. In a standard CS, all classifiers are built on the same structure and own the same properties. With a NHCS, the behavior of an artificial creature is defined by the co-evolution between several differently structured classifiers making its organism. These agents, moving in a 2D discreet environment with obstacles and resources, must adapt themselves and breed to build viable populations. Finally, ecological niches and particular behaviors, individual and collective, appear according to initial parameters of agents and environment.

1 Introduction

In A-Life, (Langton, 1989), artificial creatures like animats, (Wilson, 1985), generally evolving with a classifier system, (Holland, 1975), use only one type of classifiers and one structure for them. Genetic Algorithms, (Goldberg, 1989), or Evolution Strategies, (Bäck & Schwefel, 1993), controlling the evolution of a standard CS produces micro-mutations and crossovers inside classifiers. A Non-Homogeneous Classifier System allows agents to own several kinds of classifiers with different structures. Moreover, creatures can undergo a structural mutation : the macro-mutation, (Lattaud, 1998). It acts first on the genotype of the agent by modifying the classifiers architecture, and secondly on the phenotype by adding/removing sensors or other capacities.

In the agent classification written by Patrick Cariani, (Cariani, 1991), the only forms of adaptive agents he observed are natural animals. These agents adapt their behavior according to their environment, but they also adapt their interactions modules, like sensors and effectors. If each type of classifier in a NHCS is in relation with a particular sensor of a creature, then with a macro-evolution process on the NHCS, this agent can be defined as adaptive in the Cariani sense.

This paper presents the structure of the NHCS of artificial creatures and the associated macro-evolution process. The first part shows an overview of the agent model ETIC, (Lattaud, 1997), used to develop these animats. The second section defines the concept of NHCS and the evolution methods using the macro-mutation operator. The next chapter describes experimental results obtained on an artificial life application. Then, the conclusion discusses results and presents future works about NHCSs.

2 ETIC overview

A classification is necessary to implement a deep evolution of agents structure. The reactive/cognitive model, (Ferber, 1995), is not enough flexible for this kind of evolution. Agents build with the External Temporal Internal Classification, ETIC, can change from one class to another, and evolve *progressively* from simple to complex. Each of these classes gives particular functions to agents. The three base classes of ETIC are :

- E : External, an agent belonging to one or more subclasses of E possesses capacities to get information from its environment.
- T : Temporal, this kind of class proposes to agents memory and planning functionalities.
- I : Internal, the subclasses of I determine if agents have a knowledge of some of their internal states, but also if agents can modelize other agents.

The application testing the NHCS method uses mainly five ETIC subclasses :

- Epn : This class represents a non-oriented local perception. An Epn agent perceives all items in its surrounding environment in a range of n[1].
- Epon : The perception of Epon agents is local and oriented. Epon agents perceive only the environment in front of them, according to their direction.
- Edon : Agents belonging to this class have an oriented fuzzy perception. They cut the environment in front of them in four dials and they perceive only the number of each kind of objects in each dial. This perception can be wider than the first two.
- Eck : This class allows agents to communicate a part of their knowledge to other agents. A communication process can occur if an Eck agent perceives at less one other Eck agent. This communication is direct, point-to-point, between these two agents. The transmitted information is the best classifier of each other.
- Ipenergy : An agent with this class codes its energy level[2] in their classifiers. So, it has an internal representation of its energy and its behavior depends on it.

The Figure 1 shows the perception of three types of agents exploiting E subclasses : Ep1 agent, Epo3 agent facing east and Edo5 agent facing west.

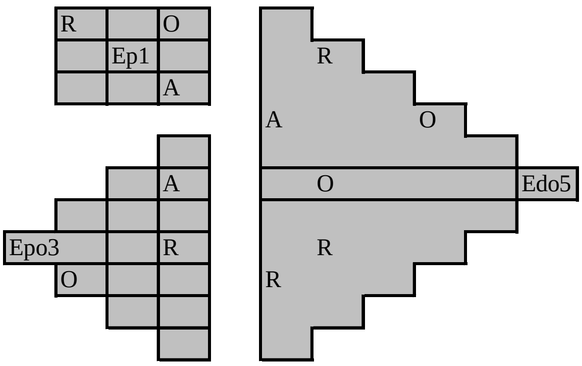

Figure 1 : Ep1, Epo3 and Edo5 agents.

These classes are cumulative, agents can be Ep1Epo3, Epo2Eck, Edo5Ipenergy … The macro-evolution process uses NHCS by adding/removing classes or increasing/decreasing class parameter. This evolution allows a deep modification of agents structure, from a phenotypic and a genotypic point of view. Agents can adapt their morphology and their behavior to the environment.

3 Non-Homogeneous Classifier System

A Non-Homogeneous Classifier System is an hybridization of classical classifier systems defined by John Holland. In a standard CS, generally, all classifiers have the same structure and same properties. In a NHCS, several types of classifiers coexist and are merged. They are build with different structures and owns different properties.

For example, an Ep2Edo7 agent can use either its local precise perception or its fuzzy dial perception to produce an action in a giving situation. Such agent possesses in its artificial organism, after some learning steps, Ep2 and Edo7 classifiers. However, if the agent has undergone a macro-evolution, it could also combine other classifiers like Ep1, Edo1, Edo2…Edo6. These classifiers are totally different and must co-evolve to develop adapted behaviors.

The process of macro-evolution is strongly linked to the macro-mutation operator. This operator is added to the three classical operators of GAs : Mutation, crossover and selection. It gets its own cycle, determining the frequency which the macro-mutation can occur, and the following parameters :

- Pmac : The macro-mutation rate.
- Pevol : The evolution rate, representing the probability of evolution/devolution of the agent.
- Pclass : The class-evolution rate, defining if the evolution/devolution process acts on an entire ETIC class or only on the parameter of an existing class of the agent.

The Figure 2 shows the algorithm for the macro-mutation operator.

[1] In a discreet environment, n is a distance of perception defined by the minimum number of elementary cells between two points.

[2] In this artificial life problem, animats have an energy parameter defining their health level and their fertility for reproduction. If energy falls under 0, the creature dies.

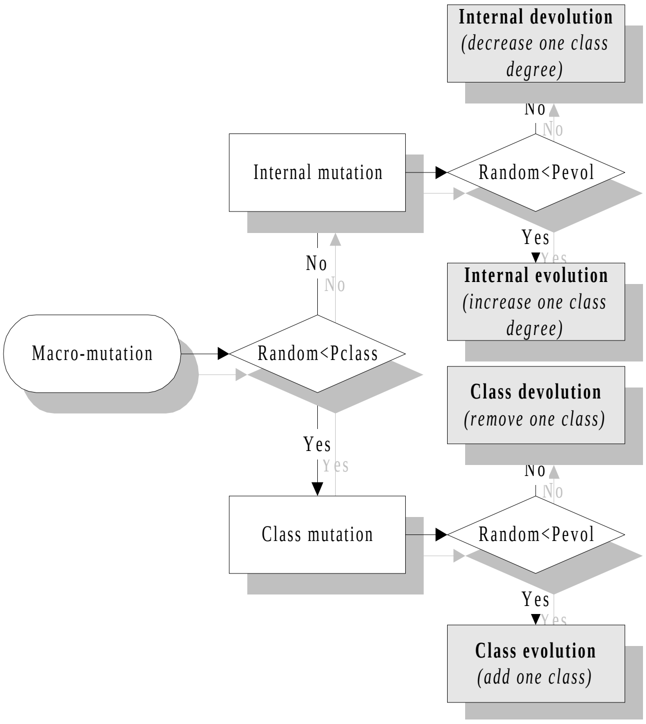

Figure 2 : Macro-mutation algorithm.

For example, if an agent is Ep1Epo3 and undergoes a macro-mutation, four cases exist :

- Rand1<=Pevol and Rand2<=Pclass : An evolution occurs on an entire class. So, the agent adds a new class, it can become Ep1Epo3Edo1, Ep1Epo3Eck or Ep1Epo3Ipenergy.
- Rand1<=Pevol and Rand2>Pclass : In this case, the evolution takes place on an existing class, the agent evolves to Ep2Epo3 or Ep1Epo4.
- Rand1>Pevol and Rand2<=Pclass : The devolution is on a complete class. The agent loses a class and mutates to Ep1 or Epo3[3].
- Rand1>Pevol and Rand2>Pclass : This is a classless devolution, the agent decreases the parameter of one existing class, but neither below 1, so the unique solution is Ep1Epo2.

The macro-evolution described previously is performed according to a cycle, but another way to proceed it exists : During an asexual reproduction. If the reproduction of agents is asexual, two fertile[4] agents are necessary to create a third one. But, if two agents with different classes can mate and give birth to a viable child, how the organism of this offspring is built ? In this system, no constraint prevents this reproduction and a macro-evolution on the child NHCS occurs. This resulting NHCS is a combination of parents NHCSs with a potential macro-mutation, with Pmac probability. The extract of the organism shows in Figure 3 belongs to a Ep1Epo3Edo5Ipenergy[5] agent after several steps of learning and evolution.

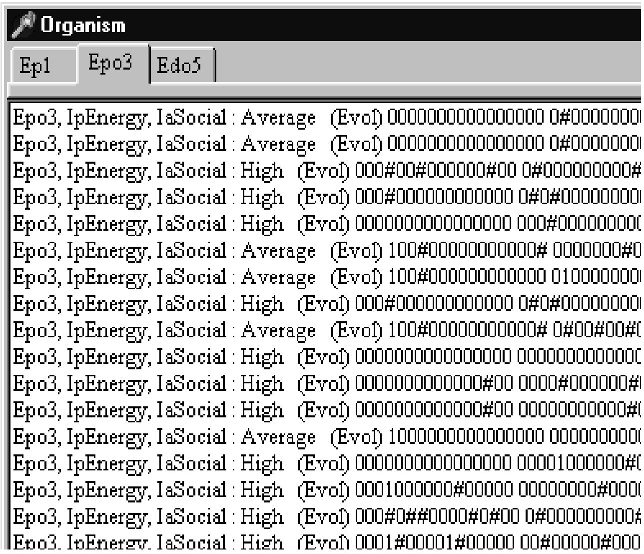

Figure 3 : Organism of a Ep1Epo3Edo5Ipenergy Agent.

Transmission of classifiers from one generation to the next generation follows two simplified models of the genetic :

- Darwinist model : Classifiers, acquired knowledge by individuals, isn't transmit to offspring. Only the NHCS, in relation with the morphology of the creature, is used to define the child capacities.
- Lamarckist model : All knowledge of parents is transmitted to offspring. Classifiers of parents are combined by the GA to obtain classifiers of the new born. Two processes are used, a (λ,μ) or a $(\lambda+\mu)$ reproduction, (Hoffmeister & Bäck, 1991).

The next part brings results concerning these reproductions and others about the co-evolution of differently structured classifiers in a NHCS.

4 Results

The aim of this A-Life application is the survival of populations of autonomous agents, (Maes, 1995), in a dynamic environment. They must adapt their behavior,

[3] If the agent has only one class, it can't lose it.
[4] Age and energy conditions.

[5] The class Iasocial isn't studied in this paper. For information, this agent have the capacity to detect differences between altruists and egoists agents. Altruists can give a part of their energy to other agents, while egoists can't.

by learning and GA, and their structure, by macro-evolution on their NHCS. At the beginning of a simulation, agents have no knowledge, they follow the Aristotian principle of the *tabula rasa*. Then, they learn by reinforcement according to the results of their actions. Thought this paper focuses only on parameters linked to macro-evolution and NHCS, several parameters are opened to users. The GA rates are standard :

- Crossover rate = 0.20.
- Mutation rate = 0.05.
- Selection mode is roulette wheel with stochastic reminder and eventually elitist adjustments, (Goldberg & Deb, 1991).

The environment evolves by adding/deleting/moving obstacles and resources. Its size is fixed to 30*30, and it is toroidal[6]. Two statistical measures are used : Mean and standard deviation. Results are obtained by meaning ten trials of 4.000 steps each. Simulations with a standard deviation too high are not considered in this paper.

4.1 POPULATION VIABILITY

The first experiment consists to show the viability of different initial populations of agents according to their ETIC classes. Agents own, at the beginning of a simulation, the class Ipenergy and one of the subclasses of Epn, Epon and Edon[7]. With these combinations, one value for n seems to be more efficient for each subclass : Ep2, Epo3 and Edo5. The Figure 4 gives the evolution of these populations during 4.000 steps.

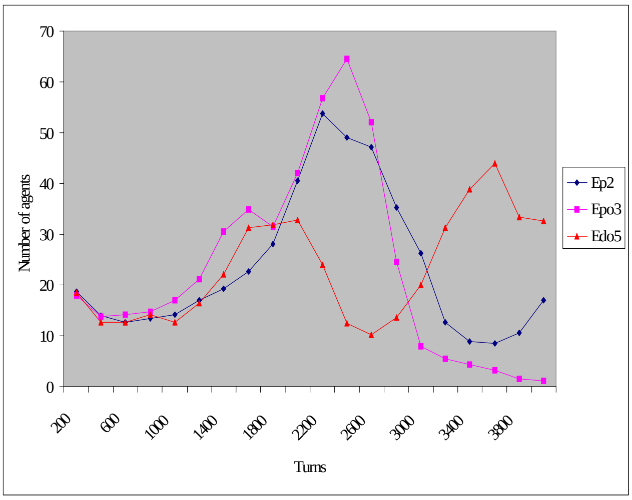

Figure 4 : Ep2, Epo3 and Edo5 agents populations.

[6] Agents going out by one side, return in the world by the opposite side.
[7] With n varying from 1 to 7.

The phenomenon appearing in this figure is the same described by Ginger Booth, (Booth, 1997) : The trophic cascade, the inversely proportional co-evolution between agents and resources. In the beginning of a simulation, agents take time to learn basic behaviors, like 'feed with a resource when it's on the same square' or 'approach a resource if possible', to maintain their energy level over 0. The resource quantity increases during this phase, then when resources are large and agents more efficient, they mate quickly and build a wide population. Resources disappear quicker, causing the fall of agents. Only the most adapted animats can survive this starvation, they are few but very efficient. Finally, resources invade again the environment and this process repeats cyclically in a trophic cascade.

The main difference between a Lamarckist and a Darwinist evolution is the quickness of the population growth. Agents have more trumps in the Lamarckist one, they develop themselves quicker and contribute to a ampler trophic cascade. In a Darwinist evolution, agents have more difficulties due to the knowledge transmission that doesn't exist during the reproduction. But globally, results are mor stable and an equilibrated cycle is reached.

4.2 ECOLOGICAL NICHES

The populations showed in the Figure 4 evolve and their agents also undergo macro-evolution on their NHCS. From generation to the next generation, they mutate to constitute more or less homogeneous populations. Often, according to initial conditions, artificial species emerge from the system and dominate the global population. The most significant case is the emergence of the communication in the NHCSs : 90% of the final populations include agents with the class Eck. These agents appear progressively along the evolution of species and finally they often invade the global population.

Other kinds of combinations emerge after several thousands steps. Generally, these agents have an efficient co-evolution in their NHCS : Ep2Epo4EckIpenergy, Epo3Edo7EckIpenergy, Ep1Edo7EckIpenergy. All of these have communication abilities and an internal representation of their energy level. The first also combines a precise short perception of its local environment and a medium precise oriented perception. The second has a medium precise oriented perception and a long range fuzzy perception. And, the third owns a very short precise non-oriented perception and a large fuzzy perception. These combinations can be found in natural animals. Most of them uses two or more sensors to detect food, predators and other members of their specie, like eye, ear, sonar, infrared, vibration sensors… However, these niches are strongly dependant of the initial parameters of the environment. If the conditions are extreme, very few resources and many obstacles, the emerging niche is Ep1Ipenergy. These agents can't lost time in communication and must take resources as soon as they detect them. As Dave Cliff said : "In the ethology literature, an adaptive behavior is any behavior which, if exhibited by an animal, increase the chance that the animal will survive long enough in its ecological niche to produce viable offspring. Underlying this definition is the assumption that, if the animal does nothing, it will die before it has a chance to reproduce", (Cliff, 94). Agents belonging to emerging niches in this application are precisely very active and they learn particular efficient behaviors, individual as well as collective.

4.3 EMERGING BEHAVIORS

Three main behaviors, cf. Figure 5, appear in this application :

- Obstacle avoidance : The agent at the right of the figure shows how it can avoid obstacles. After learning, it avoids obstacles to reach resources in the quickest way.
- Obstacle following : The same agent follows obstacles to get a resource hide behind them. Agents Epn and Epon, which have a local precise visibility, develop generally this two kind of behaviors. Edo agents, with a fuzzy perception, can't avoid and follow obstacles, unless if they are combined with Epn or Epon subclasses.
- Regrouping : When few resources exist in the environment, surviving agents, those at the left of the figure, are generally more efficient. If they perceive a resource, they try to reach it rapidly. If many of them are on the resource in the same time and feed together, their energy level can be sufficient to become fertile. So, they reproduce and give birth to very efficient agents. This regrouping behavior, logic and coherent, is interesting due to its total emergence : Nothing codes it in this application, neither code nor classifier reinforcement helps the appearance of this behavior.

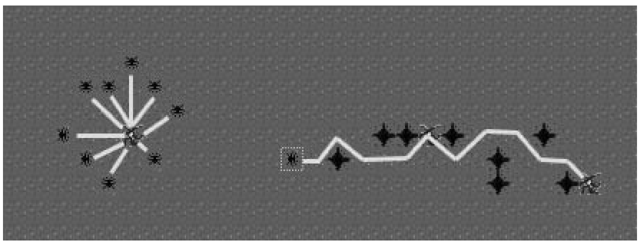

Figure 5 : Emerging behaviors.

All of these behaviors spontaneously emerge during most of simulations. Maybe other kinds of behaviors, less visible but also significantly, could emerge from this system.

4.4 PERFORMANCES WITH WOODS MODEL

To test the validity of GA used in this application, simulations based on the Woods environment, (Wilson, 1985), have been performed. The animat is modeled by a Ep1 agent, and only its learning and evolution capacities are kept. No macro-evolution occurs because it must stay Ep1 during all the time of a simulation. The Figure 6 shows the acquisition time of a resource according to the number of simulation turns. This result is based on the mean of ten trials.

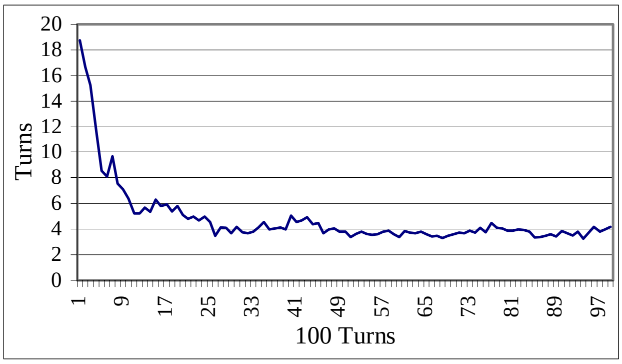

Figure 6 : Woods simulation.

In comparison of the results obtained by Stewart Wilson, the convergence to an efficient classifier base runs at the same speed. No gain is verified at short term. But, at medium and long term, performances of Ep1 agent are better than those found by the animat. After 8.000 turns, the animat moves 4.5 steps to reach the resource, while the Ep1 agent reaches the resource in 3.6 steps after 5.000 turns.

If the macro-evolution is activated in these simulations, then the Ep1 agent can evolve to more complex classes. But, as in the extreme case described in 3.2, these niches are less adapted to this environment. In fact, the simple Ep1 agent is very adapted to the Woods world.

5 Conclusion and future works

The aim of this work was to prove the adaptation of agents population using a Non-Homogeneous Classifier System combined with a macro-evolution process. Generally, agents adapt their behavior and their structure, in ETIC classes terms, to initial conditions of environment. In few cases, with the most extreme values, agents can't adapt and population dies. But with a large number of initial values, agents find the most efficient classifiers and the most adequate morphology. Tests show that creatures survive better with combination of sensors allowing them several perception properties. Moreover, agents mutating and discovering communication, with a knowledge transmission, often invade populations. Finally, several behaviors emerge totally from the system without the intervention of a particular coding constraining the reinforcement of specific classifiers.

A NHCS is necessary to implement an efficient macro-evolution process. Effectively, previously, simulations have been performed without NHCS and classifiers contained all information due to ETIC classes. So, a

Ep1Epo4Edo7 agent possesses huge classifiers with non-separate parts Ep1 and Epo4 and Edo7. If a classifier has an inefficient part, then this classifier isn't efficient either if the two other parts are correct. A NHCS allows to share the knowledge from one classifier to several classifiers. The inefficient part, transformed in a complete classifier, can be remove from the base. Moreover, the NHCS method allows a modular development of A-Life applications where each agent class can be implemented progressively.

Though several A-Life systems and Multi-Agents systems obtain similar results, NHCS is a new approach allowing co-evolution between classifiers.

Several directions can be followed for future works. First, with the large number of parameters include in this system, other simulations could de performed and new behaviors or new niches could appear. Secondly, this application also can be enriched with another kind of object : Tools. Agent using tools would improve their consumption of resources and these tools would lead to the concept described in (Zannoni & Reynolds, 1997), the cultural learning. But, the main goal is to develop an A-Life online system integrating NHCS in artificial creatures. This virtual laboratory would allow agents, like the critters of Karl Sims, (Sims, 1994), to grow and evolve in 3D real time with strong interactions with humans. Agent behaviors could be learn either by imitation of humans or by a reinforcement/macro-evolution process.

The final step of this work, in a very long term, is its implementation on real artificial creatures like robots. But, as underlined by Rodney Brooks, (Brooks, 1991), most of efficient algorithms used in virtual creatures doesn't run correctly for real robots. Several other parameters must be considered : Fuzzy information given by sensors, unpredictable effects performed by effectors… The adaptation of NHCS should be studied deeply to stay efficient in real conditions !

References

T. Bäck, H.P. Schwefel (1993), *An Overview of Evolutionnary Algorithms for Parameter Optimization*, in *Evolutionnary Computation Vol. 1*, MIT Press, 1-23.

G. Booth (1997), *Gecko : A Continuous 2D World for Ecological Modelling*, in *Artificial Life* **3**(3):147-164, MIT Press.

R. Brooks (1991), Artificial Life and Real Robots, in Towards a practice of Autonomous Systems, in Proceedings of the 1st European Conference on Artificial Life, F. Varela & P. Bourgine Eds., MIT Press, Paris.

P. Cariani (1991), *Emergence and Artificial Life*, in *Proceedings of Artificial Life II*, SFI Studies in the Sciences of Complexity, Vol. X, C.G. Langton, C. Taylor, J.D. Farmer & S. Rasmussen Eds., Addison-Wesley, 775-797.

D. Cliff (1994), *AI and A-Life : Never Mind the Blocksworld*, in *Proceedings of the 11th European Conference on Artificial Intelligence*, 799-804, A.G. Cohn Ed., Amsterdam, August 8-12.

J. Ferber (1995), *Les Systèmes Multi-Agents : vers une Intelligence Collective*, Inter-Editions.

D. Goldberg (1989), *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley.

D. Goldberg, K. Deb (1991), *A Comparative Analysis of Selection Schemes used in Genetic Algorithms*, in Proceedings of the 1st Workshop on Foundations of Genetic Algorithms, 69-93, G. J. E. Rawlins Ed., Morgan Kaufmann.

F. Hoffmeister, T. Bäck (1991), *Genetic Algorithms and Evolution Strategies : Similarities and differences*, in *Proceedings of the 1st Workshop on Parallel Problem Solving from Nature*, 455-469, H.P. Schwefel & R. Männer Eds., Springer.

J. Holland (1975), *Adaptation in Natural and Artificial Systems*, MIT Press.

C. Langton (1989), *Artificial Life : Proceedings of the 1st Workshop on the Synthesis and the Simulation of Living Systems '1987*, C. Langton Ed., Addison-Wesley.

C. Lattaud (1997), *A Classification for the control of the evolution of adaptive agents*, in *Proceedings of the 10th International Florida Artificial Intelligence Research Symposium*, 449-454, Daytona.

C. Lattaud (1998), *A Macro-Mutation Operator in Genetic Algorithms*, in *Proceedings of the 13th European Conference on Artificial Intelligence*, 323-324, Brigthon, August 25-28.

P. Maes (1995), *Modelling Adaptive Autonomous Agents*, in *Artificial Life : An Overview*, 135-162, C. Langton Ed., MIT Press / Bradford Books.

K. Sims (1994), *Evolving 3D Morphology and Behavior by Competition*, in *Artificial Life IV : Proceedings of the 4th International Conference on Artificial Life*, 28-39, R. Brooks and P.Maes Eds., MIT Press.

S. Wilson (1985), *Knowledge Growth in an Artificial Animal*, in *Proceedings of the 1st International Conference on Genetic Algorithms and their Applications*, 16-23, J.J. Grefenstette Ed.

E. Zannoni, R. Reynolds (1997), *Learning to Control the Program Evolution Process with Cultural Algorithms*, in *Evolutionnary Computation* **5**(2):181-211.