The Animat Approach: Simulation of Adaptive Behavior in Animals and Robots

Jean-Arcady Meyer
AnimatLab
Ecole Normale Supérieure
46 rue d'Ulm
75230 Paris Cedex 05
France
meyer@wotan.ens.fr

In various application areas there is an urgent need for autonomous artifacts that should be able to adapt their behavior to changing circumstances. Given some overall mission in a given, unpredictable, and eventually hostile, environment - such as seeking a specific information within the Internet or performing a specific task on the surface of Mars - one wishes such artifacts to be able to cope with unforeseen circumstances in the absence of any human intervention and to take any initiative that makes it possible to fulfill their objective. Although animals are living examples of such fully autonomous and adaptive agents, it turns out that artifacts with equivalent properties yet remain to be synthesized. However, several progresses have been recently made in this respect by people who attempt to devise so-called *animats*, i.e., simulated animals or real robots whose structure and functionalities are inspired by those of animals and that exhibit some variety of adaptive behavior (Blumberg et al., 1998; Maes et al., 1996; Husbands et al., 1994; Meyer et al., 1993; Meyer and Wilson, 1991).

An animat is usually equipped with sensors, with actuators, and with a behavioral control architecture that relates its perceptions to its actions and that allows it to "survive" - at least temporarily - in its environment. In this context, survival depends upon some *essential variables* that must be monitored and maintained within a given *viability zone*, an ability that can be enhanced, should the animat be capable of learning which actions elicit a positive or negative reward from the environment (Figure 1) (Meyer, 1995, 1996, 1997).

The control architecture of a given animat can be *innate* - because it is programmed or wired in by a human designer - or *acquired* - because it results from a learning process, which occurs at an individual or at a population level.

1 Subsumption architecture

Many animats exhibit adaptive behaviors because they have been purposely programmed or cabled this way. For instance, several robots have been built by Brooks and his students at the Artificial Intelligence Laboratory of MIT, whose sizes, morphologies, and missions vary, but which are all controlled by the same *subsumption architecture* (Brooks, 1986). Essentially, this architecture consists in superimposing layers of networks of finite-state machines, augmented with various timers and registers. Each layer connects sensors to actuators and implements a control system which achieves a certain level of competence. Higher-level layers can subsume the roles of lower levels by suppressing their outputs. However, lower levels continue to function

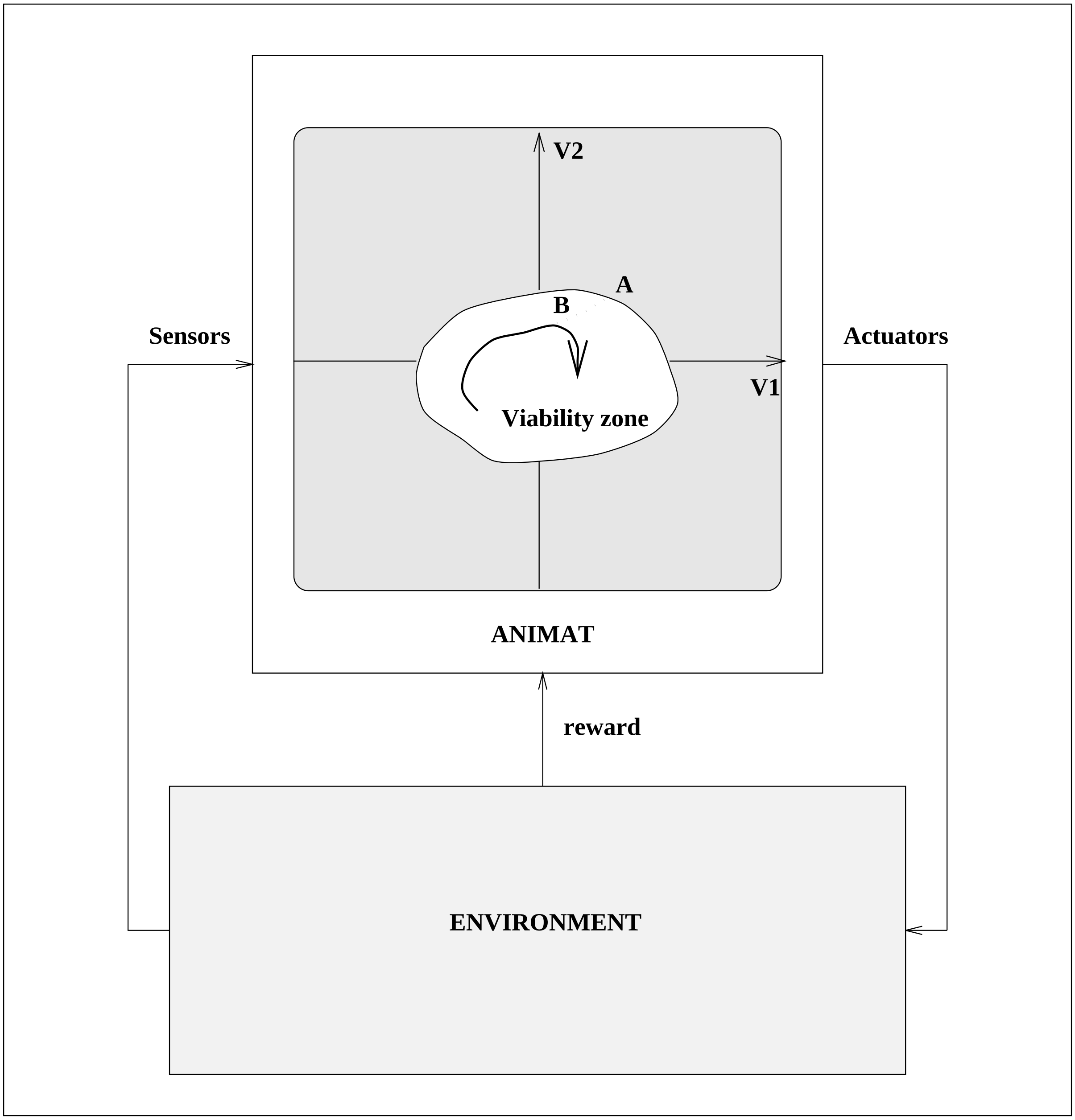

Figure 1. The interactions betwen an animat and its environment. The behavior of this animat is said to be adaptive because a corrective action has been taken at point B so as to avoid crossing at point A the viability zone associated with the two essential variables V1 and V2.

as higher-level layers are added and higher levels of competence are achieved. Such a design mobilizes no central control module responsible for analyzing the environmental situation at each instant, then for deciding to activate one specific behavior module or another. On the contrary, each competence module reacts in parallel to the environmental characteristics concerning it, and the collective behavior is an *emergent property* resulting from the interactions between instantaneous behaviors of the modules. The result is a robust and flexible robot-control system that exhibits several advantages in comparison with more traditional approaches (Figure 2).

For example, its subsumption architecture allows the robot Genghis to chase infrared sources over rough terrain. Likewise, it permits Squirt - "the world's largest one-cubic-inch robot" - to act as a "bug", hiding in dark corners and venturing out in the direction of noises only after the noises are long gone.

As far as biology is concerned, it is possible to account for the behavior of the coastal snail *Littorina* by supposing that this behavior is controlled by a subsumption architecture (Figure 3)(Connell, 1990).

One can indeed regard this behavior as depending on two basic competence modules: UP, which tells the snail always to crawl against the gravitational gradient, and DARK, which tells it to avoid light by crawling directly away from the source. However, DARK subsumes UP, i.e., if a very bright light source is present, the snail will crawl away from it even if it means going downward. On Figure 3a, this interaction is shown by a circle with an arrow entering it, suggesting that the output of DARK replaces the output of UP. In a similar manner, it can be supposed that a competence BRIGHT subsumes DARK because, if one turns the snail upside down, instead of avoiding light, it will now head toward bright areas. However, because this light-seeking behavior occurs only underwater, another competence module must be added to the control architecture: DARKER, which takes precedence over all the other light-sensitive behaviors when the snail is out of water. Finally, a last module, STOP, halts the snail when it encounters a dry surface and thus keeps it from wandering too far inland. Figure 3b shows how this collection of competence modules and their interactions aids the snail in its pursuit of food and how it allows it to arrive at the region of maximum algae concentration - i.e., in cavities near the surface of the water - even if it has to negotiate major obstacles along the way.

2 Swarm intelligence

Another biological metaphor underlies much work on animats, i.e., that of insect society or of *swarm intelligence*. Here, the idea is to use a colony of animats with individual behaviors that may be quite simple but, due to the fact that they interact with each other, can exhibit an emergent collective behavior that is relatively complicated and adaptive. In particular, this collective behavior can be maintained even in the event of the dysfunction of one or more individual animats.

As an example, the work of Reynolds (1987) leads to interesting applications in the field of computer animation. It illustrates how a collective flocking behavior can emerge from the interactions of very simple bird-like creatures - called *birdoids* or *boids* for short - each characterized by a direction of motion, a velocity and minimal visual abilities that allow it to detect nearby obstacles or conspecifics. Each boid obeys the following three rules:

- maintain a minimum distance from surrounding boids
- match its velocity with that of the boids in the neighborhood

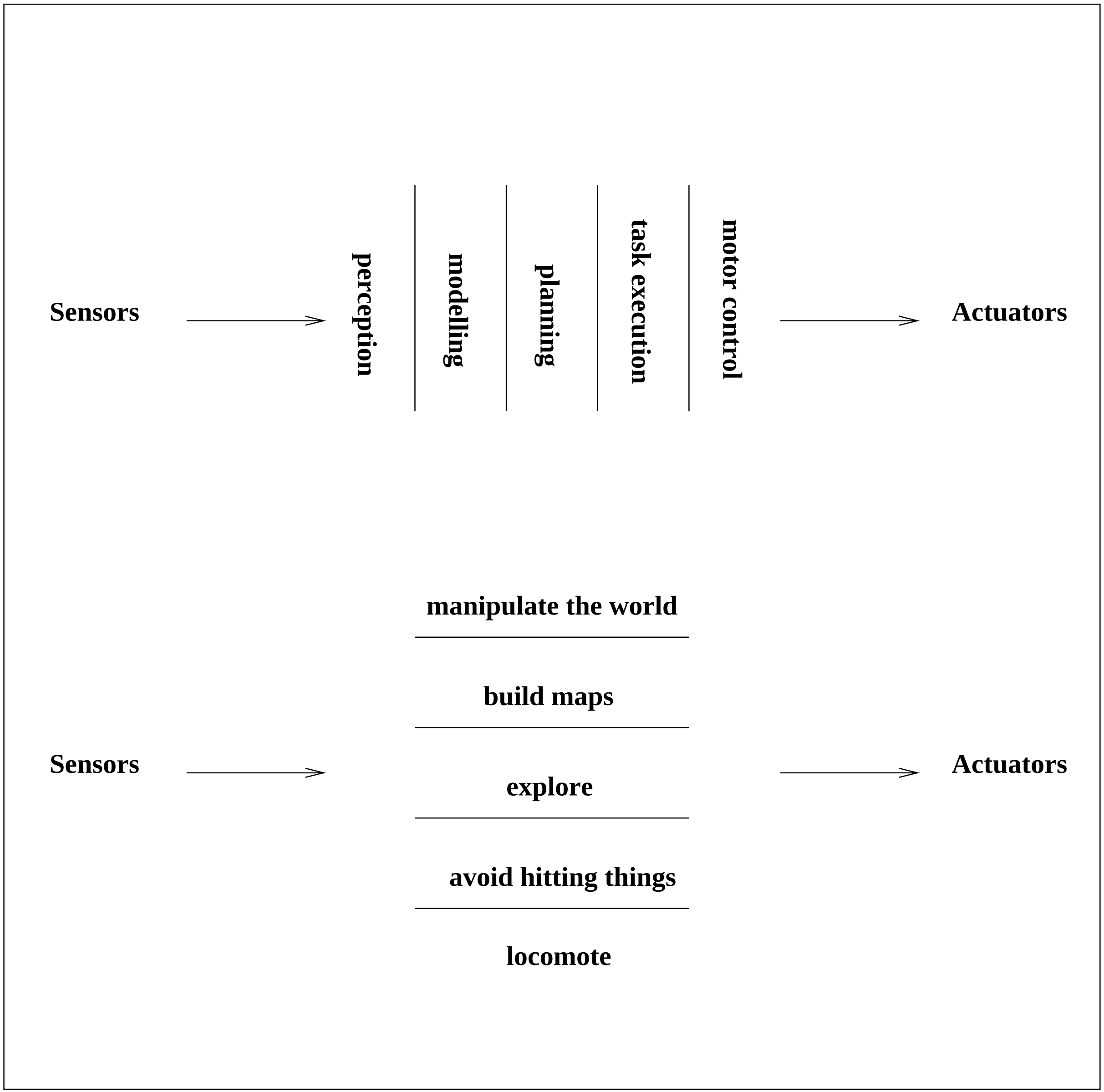

Figure 2. The traditional decomposition of the control architecture of an autonomous robot is to break processing into a chain of information processing modules proceeding from sensors to actuators (top). The failure of a single module in the chain entails the failure of the whole architecture. In the subsumption architecture (bottom), the decomposition is in term of competence modules, each of which connects sensors to actuators. If some module fails, there is a chance of surviving by relying on the modules that are still functional.

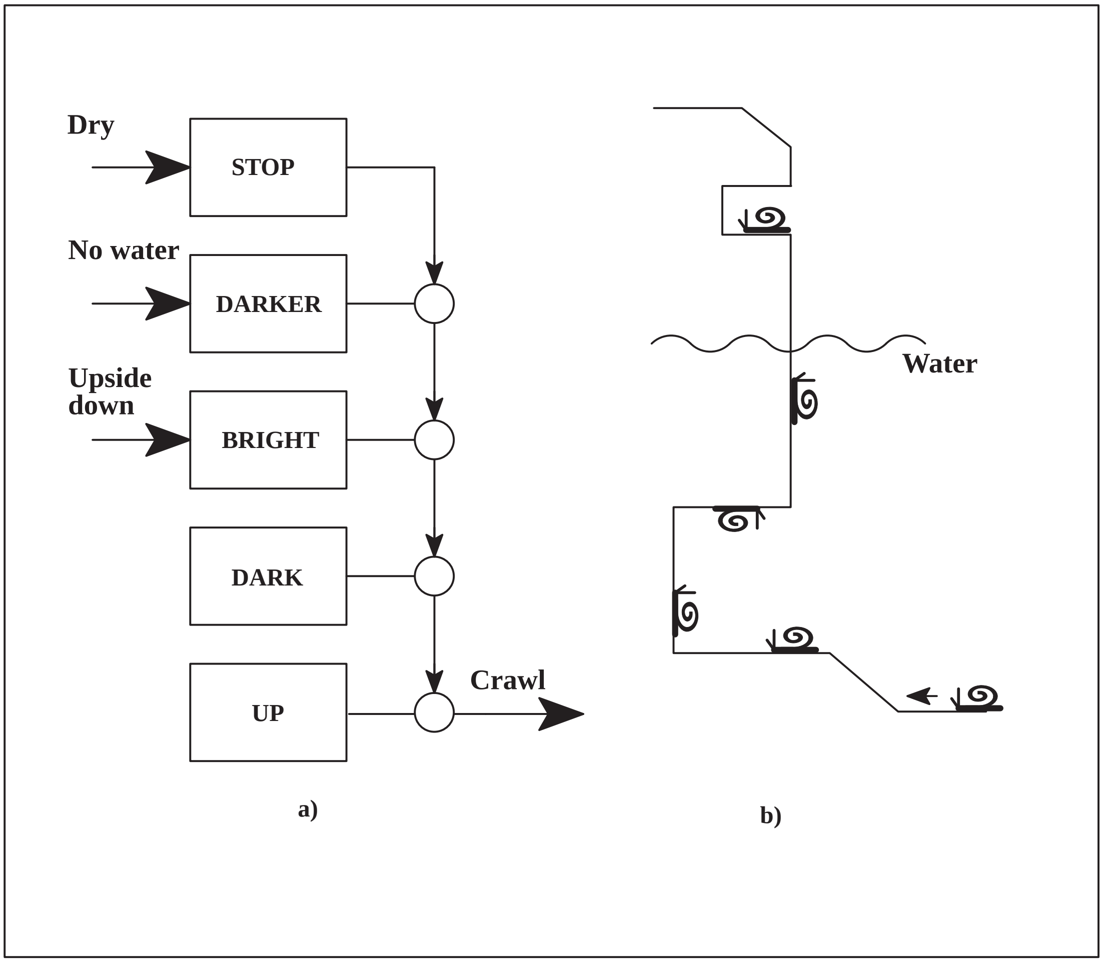

Figure 3. a) The subsumption architecture controling Littorina's behavior. b) Littorina's adaptive food seeking behavior.

- move towards the perceived center of mass of the boids in the neighborhood.

Although such rules rely on purely local knowledge - no boid has a global view of the positions of each obstacle in the environment, nor of the velocities and positions of all its conspecifics - the resulting behavior of the whole community is that of a flock. Such behavior appears extremely coordinated and realistic, because boids do not collide with each others and because, when the trajectory of an individual eventually diverges from that of the rest of the flock, this individual soon speeds up in the direction of its nearest neighbors and rejoins the group. Moreover, in the presence of an obstacle, the flock eventually gracefully splits into two subgroups, each skirting around one side of the obstacle (Figure 4). Such behaviors are emergent because nothing in the corresponding simulation program expressly codes for them.

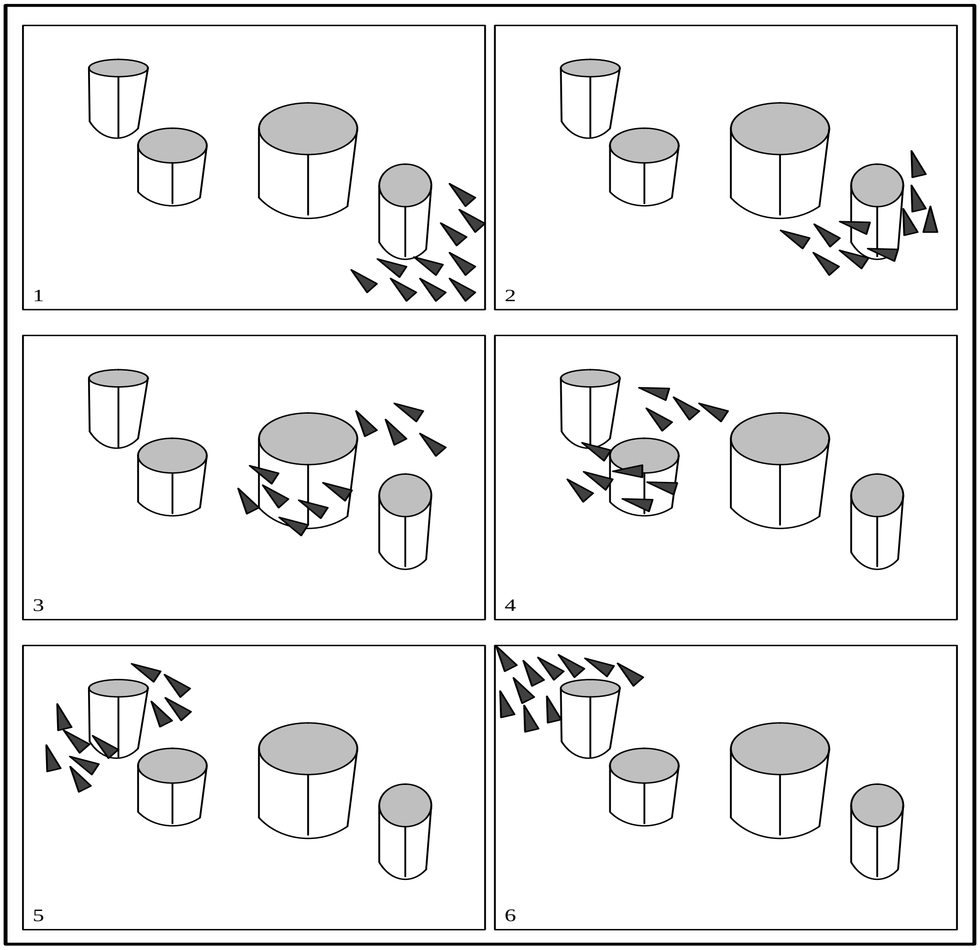

Figure 4. Boids skirting around obstacles. Adapted from Boids Demo (Video from Symbolics Graphics Division)

Another application of the swarm intelligence metaphor has been realized by Colorni et al. (1992) and relies on the observation that every ant lays down traces of pheromone on the paths it travels and that these traces incite other ants, that otherwise would wander randomly, to follow this same path. These new ants in turn lay down pheromone traces on the path in question and reinforce its attractive force. These individual behaviors generate an interesting emergent collective property: that of making it possible to identify the shortest path around an obstacle, as shown on Figure 5.

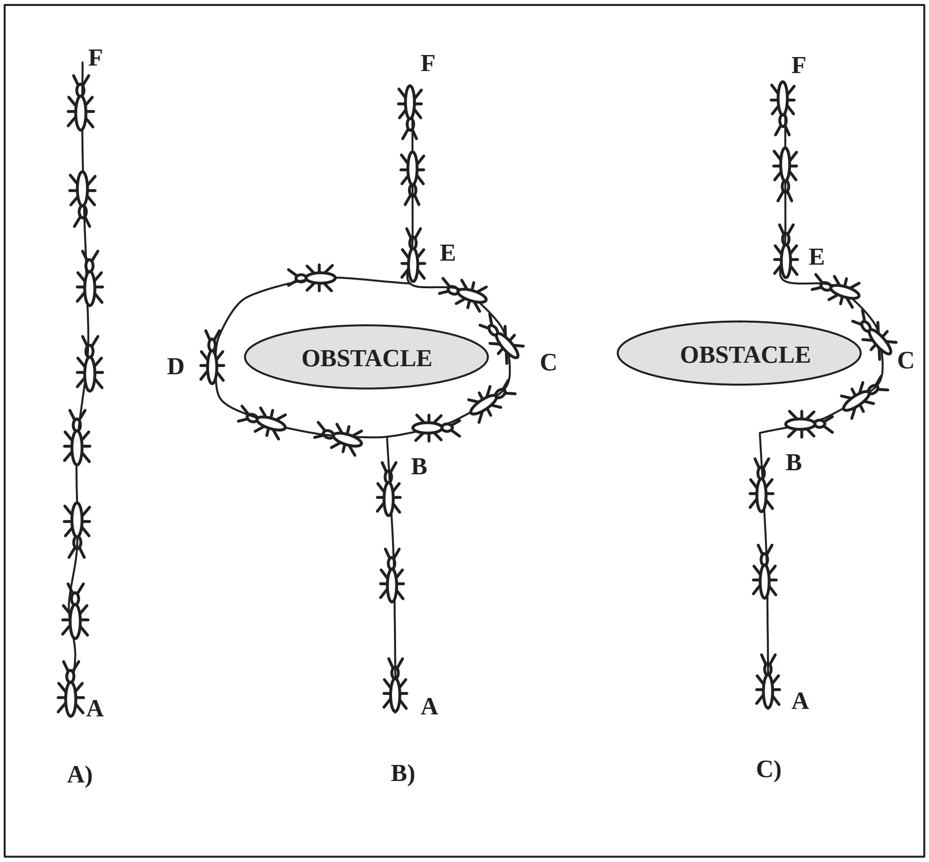

Figure 5. Collective problem solving by ants. A) Some ants are walking on a path between points A and F. B) An obstacle suddenly appears and the ants must get around it. More pheromones are laid by unit of time on path BCE than on path BDE. C) At steady-state the ants follow the shortest path.

This property has been exploited in a program that seeks the optimum solution to the Traveling Salesman Problem. This problem is actually solved collectively by a colony of ants that are turned loose on the network of towns and mark the paths they explore. Similar applications deal with other operation research problems, like job shop scheduling.

3 Associative learning

Although the overall organization of an animat's control architecture can be fixed by a human designer, its inner workings can be still improved over time by means of an individual learning process - like *associative learning* or *reinforcement learning*. The former permits an animat to learn, to memorize, and to reconstruct a pattern by associating the various parts of this pattern with one another. The latter permits an animat to recognize and to favor those behaviors that yield rewards rather than those that yield punishments.

The work of Matarić (1991) resorts to a variety of associative learning to let a robot build a *cognitive map* of its environment. The metaphor of the cognitive map concerns the way in which biologists conceive certain animals' mode of memorizing the information they gain about the spatial organisation of their environment and how they make use of this information to navigate from one point to another. It is thought that these maps contain both topological and metric information about several landmarks that the animal has learned to distinguish in its environment. It is also thought that, at least in some animals like rodents, these spatial *representations* are encoded in a part of the brain called *hippocampus* and involve so-called *place cells* whose activity depends on the animal's current location.

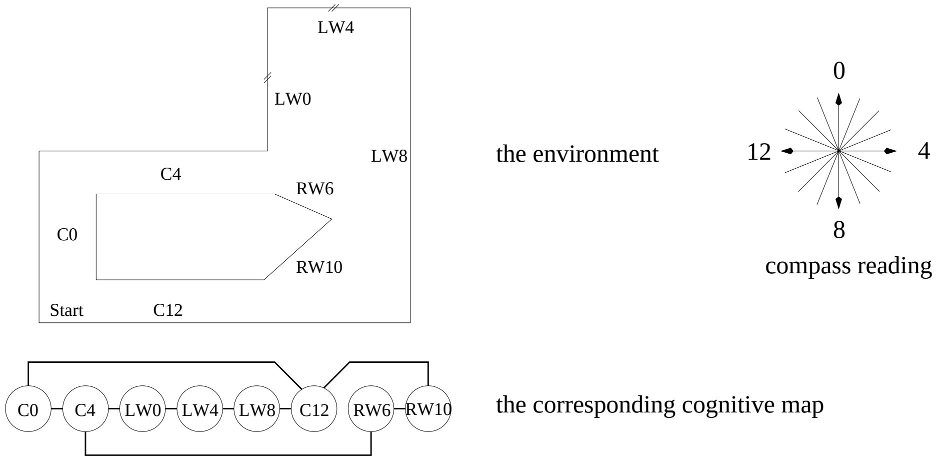

Figure 6. Cognitive map build by Matarić's robot in a given environment. A place is labeled according to the characteristics of the range sensor readings (RW: right wall, LW: left wall, C: corridor) and to the compass reading (from 0 to 15) when the robot, driven by its wall-following procedure, travels through it. In the corresponding cognitive map, each place is represented by a node and any two nodes can be linked dynamically to code the adjacency relationship.

Matarić's animat is a real robot equipped with twelve sonar sensors that report the distance to the nearest obstacle, with a flux-gate compass, and with a series of sensors that can detect stalling - an information which prevents the robot from pushing helplessly against environmental barriers. This robot is also equipped with motors that permit forward and backward motion, right or left turns, and stopping. Such a combination of sensors and actuators allows the robot to wander around office environments, building a cognitive map based on landmarks, and then to use that map to navigate from one location to another.

Control of the robot is ensured by a subsumption architecture that calls upon a hierarchy of three competence layers. In the lowest layer, simple reflex-like rules combine into emergent collision-free motion behavior and allow the robot to wander around following boundaries (such as walls and furniture clutter) in an indoor environment. The middle layer profits from the motion of the robot while tracing boundaries to dynamically extract landmarks in the environment. Such landmarks are selected as large, permanent, robustly-detectable environmental features like walls and corridors, and the method is based on continuous updating of confidence levels associated with features detected as the robot is moving. To allow disambiguation, landmark descriptors are augmented with estimated positions (provided by integrated compass bearings) and size. The top layer constructs a distributed map of the world from those landmarks and uses it to find paths.

The cognitive map is represented internally as a graph with nodes that are computational elements, each representing a landmark unique in the world. Thus, while the robot explores the environment shown on Figure 6, it records in its map that place C4 is passed through while it moves in the direction indicated by its compass and while its sonars detect similar nearby obstacles on its right and on its left - thus suggesting that place C4 is actually a corridor. Likewise, it records that a right turn leads from C4 to another place, RW6, which is passed through while the robot moves in a given direction and while it detects a nearby obstacle on its right - thus suggesting that place RW6 is adjacent to a right wall. In other words, places like C4 and RW6 on the robot's cognitive map play the role of place cells that become activated when the robot passes through the corresponding places in the environment and such a map allows the robot to position itself correctly.

Because the map also provides information about the physical length of each place, the shortest path leading to any given goal from the current place can be generated by initiating a spreading activation process throughout the map, in all directions from the goal. Insofar as the speed of this process depends upon the length of the places through which it travels, the direction from which goal activation first arrives in each place indicates the direction in which to move in order to reach the goal.

4 Reinforcement learning

The control architecture designed by Booker (1991) involves several endogenously-generated time-varying goals and implements what ethologists used to call a *motivational system.* It calls upon a classifier system - i.e., a rule-based structure that aims at reproducing various cognitive processes in animals and men (Holland et al., 1986). Such a structure allows an animat to learn to produce goal-seeking sequences through the use of rules that manipulate objects, goals, and object/goal associations. It therefore contrasts with more classical control architectures that implement rules coding simple stimulus/response associations.

Booker's animat moves about on a grid containing food objects and noxious objects, these objects being located in the middle of a stimulus aura whose intensity diminishes with the distance from the object in question. Contact with food gives rise to a reward, and contact with a noxious object to a punishment. To survive, the animat must learn which actions are appropriate in which situations and, therefore, it must build a model of its environment. To this end, its control architecture mimics the hierarchical structure described by ethologists to model instinctive behavior in animals. Basically, it involves three *instinctive centers* concerned with locomotion, food-seeking, and pain aversion, as well as *innate releasing mechanisms* that prompt the animat to pursue two goals: to eat in the presence of food and to flee in the presence of a noxious object (Figure 7).

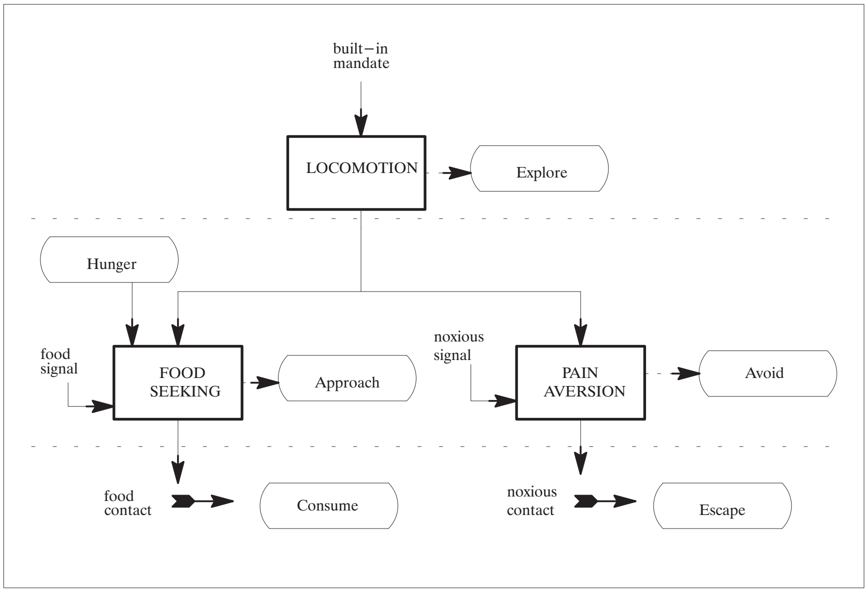

Figure 7. Control architecture of Booker's animat. The rectangles correspond to instinct centers and arrows to innate releasing mechanisms. Each instinctive center is characterized by a releasing stimulus and an associated action, and can be submitted to two sources of control: the current motivational state and other instinctive centers higher in the hierarchy. Here, the food-seeking and pain-aversion centers are located on the same hierarchical level, under the control of the locomotion center. The food-seeking center is subjected to an additional motivational control exerted by the degree of hunger the animat feels.

The animat's default activity consists in exploring its environment. This activity is interrupted when the strength of the sensory or motivational signals associated with a lower instinctive center allow it to be activated and to trigger an approach or avoidance behavior. Eventually, these behaviors are liable to lead the animat feed or flee, according to what kind of object it has come in contact with, and to shut off the activity of the corresponding center, thus allowing the animat to resume its exploration. In this approach, the animat's adaptive capacities derive from its ability to learn representations. These representations are used to classify the objects experienced in the environment into categories that have affective significance and to accordingly release the appropriate behavior. In other words, Booker's animat learns to respond to the regularities hidden behind the equivocal nature of its sensory cues and to take advantage of the useful and structured information afforded by the environment with respect to its goals.

5 Evolution

Because the nervous system of animals has been shaped by *natural selection*, not by a human designer, several researchers advocate the use of automatic designing procedures that would bypass human intervention insofar as possible, and that would adapt the control architecture of an animat to the specific environment it lives in, and to the specific survival problem it has to solve. Thus, several research efforts have addressed the simulation of evolutionary processes that act upon the *genotypes* of individuals in a population in order to improve their *phenotypes*. These efforts involve the implementation of artificial selection processes that eliminate individuals with ill-adapted behaviors and favor the reproduction of individuals displaying well-adapted behaviors. Most often, they call upon a *genetic algorithm* or some variant (Holland, 1975; Goldberg, 1989), which evolves the control architectures, and eventually the overall morphology, of a population of animats.

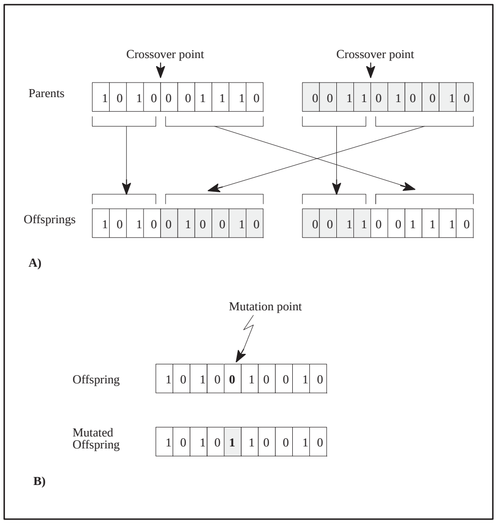

Figure 8. How new solutions are discovered by a genetic algorithm operating on chromosomes organized as chains of binary symbols. A) Role of the crossover operator. A crossover point is chosen at random within the chromosomes of two parents. Before that point an offspring inherits the genetic material of one parent, after that point it inherits the genetic material of the other parent. B) Role of the mutation operator. A mutation point is chosen at random within the chromosome of an offspring and the binary value of that point is swapped. Equivalent genetic operators have to be devised when chromosomes have a more complex structure.

Like classifier systems, genetic algorithms have been invented by John Holland. They manage in parallel a population of so-called *chromosomes*, each of which codes a possible solution to a given optimization problem, like improving the phenotype of an animat. Each of these chromosomes can therefore be assigned a *fitness* that assesses the corresponding solution. The application of the genetic algorithm accordingly causes this population to evolve from generation to generation. It does this by maintaining, for each chromosome, a probability of reproduction proportional to the chromosome's fitness and by using genetic operators such as *mutation* and *crossover* to give rise to new solutions in the population (Figure 8). This evolutionary process generally causes chromosomes of ever-increasing fitness to be generated until the optimum value is reached, or sufficiently nearly so for all practical purposes.

As an example of such an approach, Sims (1994) genetically encodes into chromosomes organized as directed digraphs both the morphology and the control architecture of various animats. In this approach, a first-level directed graph of nodes and connections encodes the phenotype embodiment of a virtual animat as a hierarchy of three-dimensional rigid parts that can exhibit a recursive structure or can duplicate instances of the same appendage (Figure 9).

Each node in the graph contains information describing a rigid part, notably its physical shape and the way its motion relative to its parent part is constrained. Likewise, each connection contains information about the position of a child part relative to its parent. Nodes and connections also encode how many times a given node should generate a phenotype part when in a recursive cycle, and when and how tail- or hand-like components should be incorporated at the end of a chain of repeating units.

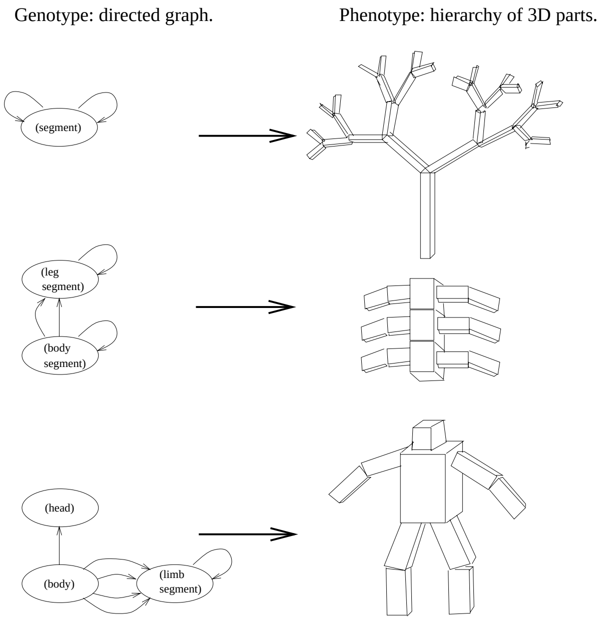

Figure 9. Designed examples of genotype first-level directed graphs and corresponding animat morphologies in Sims's approach.

The control architecture of a given animat is encoded as second-level directed graphs of nodes and connections, included in each first-level node and describing the neural circuitry of the corresponding morphological unit, or belonging to an overall central control system. These

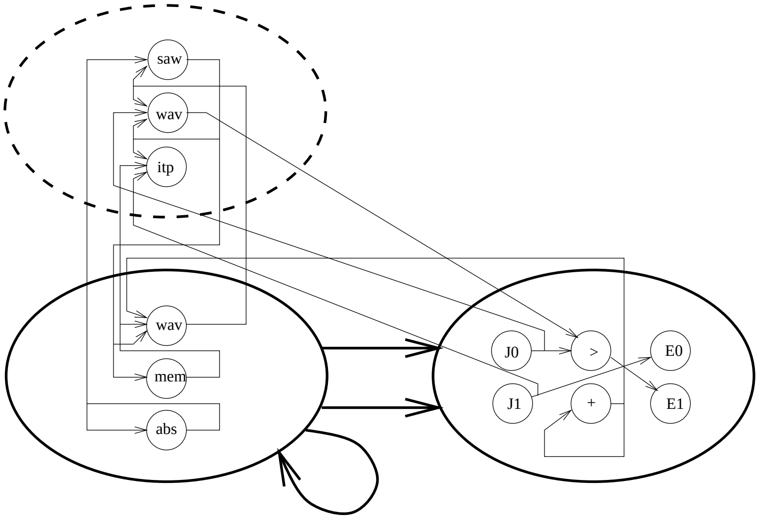

Figure 10. An evolved nested-graph genotype. The first-level graph in bold describes the animat's morphology. The second-level graph describes its neural circuitry. J0 and J1 are joint angle sensors; E0 and E1 are actuator outputs. The dashed node contains centralized neurons that are not associated with any part.

second-level nodes describe either input sensors, internal neurons or output actuators. The second-level connections define the flow of signals between these nodes and allow the neurons and actuators within a morphological unit to receive signals from sensors or neurons in their parent or in their child units (Figure 10).

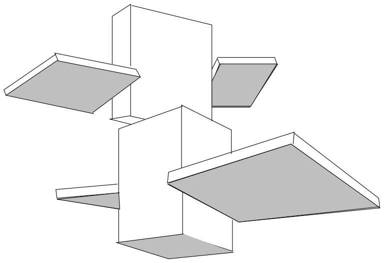

Figure 11. The phenotype morphology generated from the evolved genotype shown in Figure 10.

Each sensor is contained within a specific part of the body and measures either aspects of that part or aspects of the world relative to that part. Some sensors provide information about joint values, others react to physical contacts, and still others react to a global light-source. Internal neurons can perform diverse functions on their inputs — like *sum, product, divide, interpolate, memory, oscillate-wave, oscillate-saw*, etc. — to generate their output signals. As for actuators, each one exerts a muscle force on a specific degree of freedom of a specific joint.

Sims's approach allows virtual animats to evolve by optimizing for a specific behavior, such as swimming, walking or following. Every animat is grown from its genetic description and then placed in a dynamically simulated virtual world, with which it interacts realistically, thus allowing its fitness to be assessed. For instance, Figure 10 shows the evolved genotype of a

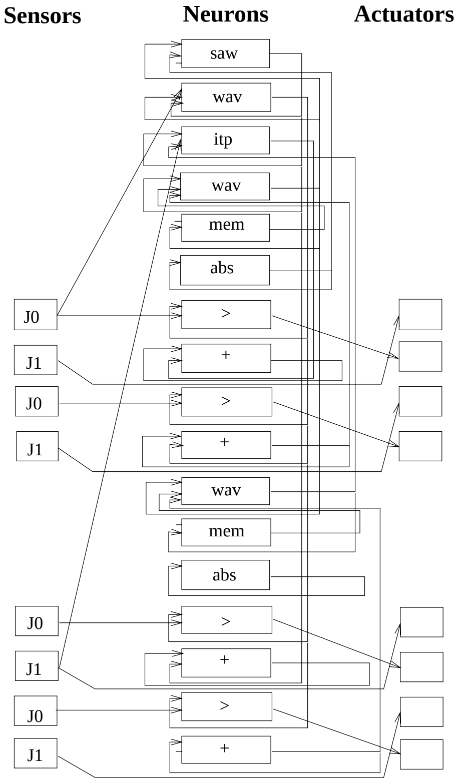

Figure 12. The phenotype control architecture generated from the evolved genotype shown in Figure 10. The actuator outputs of this architecture cause paddling motions in the four flippers of the morphology shown in Figure 11.

swimming animat. Its phenotype morphology includes four flippers shown in Figure 11, which are put into proper paddling motion by the phenotype control system of Figure 12.

In another work, Sims designed a system of co-evolving animats that compete in one-to-one contests for gaining control of a common resource. The results obtained prove that interesting and diverse strategies and counter-strategies are likely to emerge within such a framework.

6 Development and evolution

In the previous realization, the genotype of an individual is instantaneously decoded and transformed into a phenotype. In nature, however, such a transformation entails a more or less long lasting *developmental process*, during which genetic information and environmental influences interact to shape the phenotype of a given organism. To study the consequences of such effects, Kodjabachian and Meyer (1998a, 1998b) encode in a tree-like chromosome the developmental program of a neural network that controls the locomotion of a six-legged animat. Each leg is associated with an angle sensor neuron - whose activity depends upon the leg's angle with the vertical - and with three motor neurons - that are respectively responsible for moving the leg forward or backward, and for lifting the foot. Thus the evolutionary problem to solve is that of designing a nervous system that will coordinate the activities of its sensor, motor and internal neurons, such as to ensure walking.

Within such a framework, each cell in a developing network has a copy of the chromosome that codes the developmental process, and each cell reads the chromosome at a different position. The chromosome is represented as a grammar tree, with ordered branches whose nodes are labeled with character symbols. These symbols represent instructions for cell development that act on the cell or on its connections to other cells. During a given developmental step, a cell executes the instruction referenced by the symbol it reads, and moves its reading head down in the tree. Figure 13 illustrates the corresponding process.

The developing architectures are afforded the possibility to draw connections with sensory and motor neurons whose functionalities and spatial organization are provided by the experimenter, depending upon the body plan of the animat and upon an incremental approach that is sketched in Figure 14. Such an approach takes advantage of the geometrical nature of the developmental substrate to generate and connect successive neural modules implementing different competencies. In particular, it has been used to evolve a neural network that controls mere locomotion in a 6-legged animat and then to evolve other neural modules that control higher-level behaviors. These modules may influence the locomotion module by creating inter-modular connections. For instance, Figure 14 shows how the successive connection of two additional modules with a locomotion controller was used to first generate a goal-seeking behavior and then to generate additional obstacle-avoidance capacities that are shown in Figure 15.

Such results and several other applications (Kodjabachian and Meyer, 1995; Husbands and Meyer, 1998) demonstrate that quite complex control architectures can be generated by much simpler developmental programs, thus tremendously reducing the size of the solution space that the evolutionary process explores, and leaving hope for the automatic generation of more cognitive control architectures than those that it has been so far possible to let evolve.

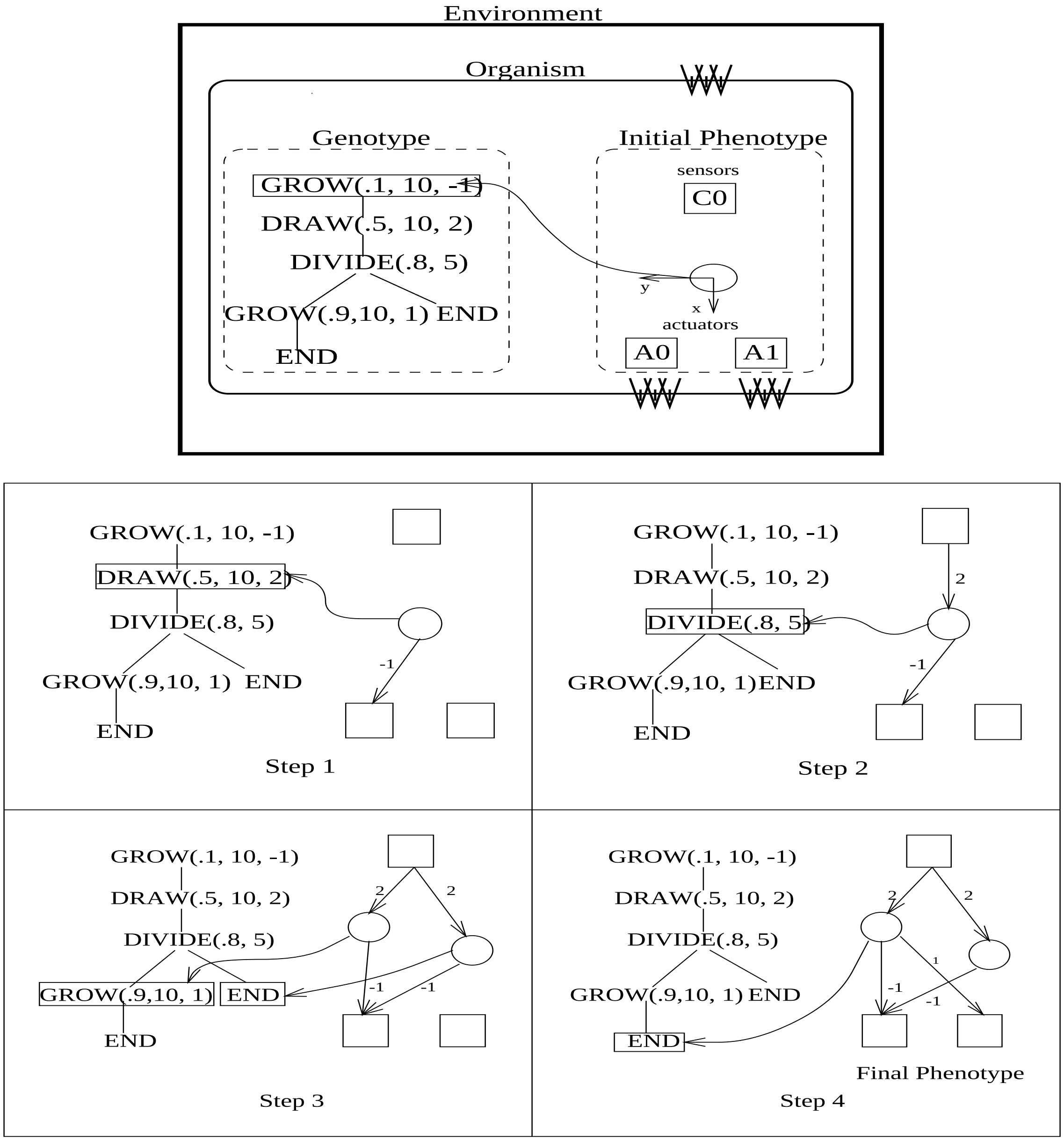

Figure 13. Developmental encoding scheme used by Kodjabachian and Meyer. The genotype that specifies an organism's nervous system is encoded as a grammar tree whose nodes are specific developmental instructions. Within such chromosomes, mutations change an instruction into another, and crossovers swap branches. Each cell in the developping network reads the chromosome at a different position. The DIVIDE instruction causes a cell to divide and generate a daughter cell in a given direction and at a given distance - according to the arguments of the instruction. Likewise, GROW and DRAW instructions cause a cell to draw respectively efferent and afferent connections with other cells, in a given direction, at a given distance, and with a given synaptic weight. END instructions cause a cell stop developing and become a mature neuron. Each neuron is modeled as a leaky-integrator, characterized by a time constant and a bias that can be set by instructions SETTAU and SETBIAS. More or less developmental steps are required to generate a phenotype, depending upon the length of the corresponding genotype.

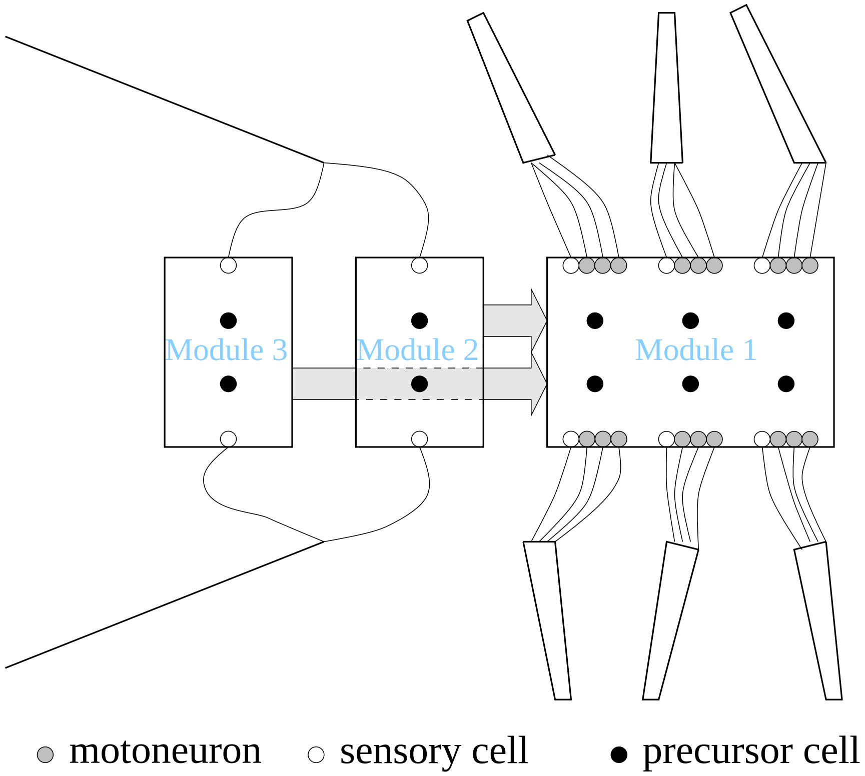

Figure 14. Incremental approach to the evolution of walking and higher-level behaviors in an artificial insect. During a first evolutionary stage, Module 1 is evolved. This module receives proprioceptive information through sensory cells and influences actuators through motoneurons. In a second evolutionary stage, Module 2 is evolved. This module receives specific exteroceptive information through dedicated sensory cells and can influence the behavior of the animat by making connections with the neurons of the first module. Finally, in a third evolutionary stage, Module 3 is evolved. Like Module 2, it receives specific exteroceptive informations and it influences Module 1 through inter-incremental connections. In the application to be described later, no connections between Module 2 and Module 3 are allowed but such a constraint could easily be relaxed in other applications.

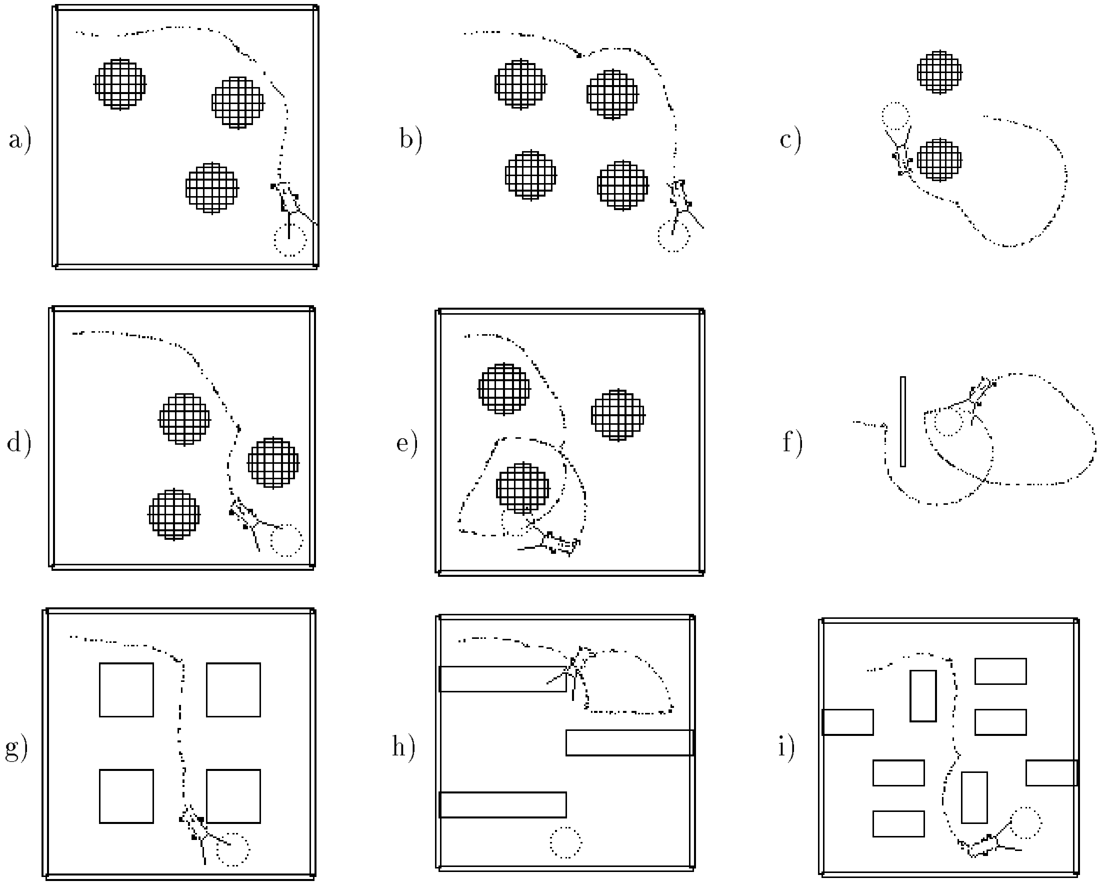

Figure 15. Experimental results obtained when goal-seeking and obstacle-avoidance behaviors are evolved in an artificial insect. Cases (a-c) show the animat's trajectory within 3 test environments. Cases (d-i) show results of generalization experiments, in 6 new environments. The animat can deal with obstacle shapes never met during evolution (f-i). However, it cannot always avoid hitting sharp corners (h) because of the blind spot between its antennas.

7 Prospects

Like many animals (Roitblat and Meyer, 1995), instead of being passive reflex devices, animats are active information processors that seek useful information in their environment, encode it into internal representations of objects and causal effects, and use such representations for their benefit in flexible and intelligent ways. They can move in their environment, avoid obstacles, and reach goals (Trullier and Meyer, 1997a, 1997b; Trullier et al, 1997; Trullier and Meyer, 1998). They can interact, and even communicate, with each other in order to collectively solve difficult tasks. They can evolve, develop, learn, memorize and plan (Donnart and Meyer, 1996a, 1996b).

Although the mechanisms that generate such abilities are presently studied more or less in isolation, future research will certainly aim at combining them. Major improvements in animats' autonomy and adaptiveness, and therefore major steps in the design of truly useful artifacts, are thus to be expected in the near future.

Besides practical applications, future animat research should provide a valuable contribution to theoretical biology, and enable a better understanding to be gained of the interactions between development, learning and evolution - i.e., the three main adaptive processes exhibited by natural systems. Likewise, animat research should contribute to embed human psychology into an evolutionary perspective, and help to understand how the adaptive capabilities of simplest animals ultimately lead to human intelligence and cognition.

8 References

Blumberg, B., Meyer, J.A., Pfeifer, R. and Wilson, S.W. (Eds.). From Animals to Animats 5. Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior. The MIT press/Bradford Books. 1998.

Booker, L. B. Instinct as an Inductive Bias for Learning Behavioral Sequences. In Meyer and Wilson (Eds.) From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. The MIT Press/Bradford Books. 1991.

Brooks, R.A. A Robust Layered Control System for a Mobile Robot. IEEE Journal of Robotics and Automation. 1986, 2, 14-23.

Cliff, D., Husbands, P., Meyer, J.A. et Wilson, S. From Animals to Animats 3. Proceedings of the Third International Conference on Simulation of Adaptive Behavior. The MIT Press/Bradford Books. 1994.

Colorni, A., Dorigo, M., and Maniezo, V. Distributed optimization by ant colonies. In Varela and Bourgine (Eds.). Toward a practice of autonomous systems: Proceedings of the First European Conference on Artificial Life. The MIT press/Bradford Books. 1992.

Connell, J.H. Minimalist mobile robotics. A colony-style architecture for an artificial creature. Academic Press. 1990.

Donnart, J.Y. and Meyer, J.A. Hierarchical map-building and self- positioning with MonaLysa. Adaptive Behavior. 1996a, 5(1), 29-74.

Donnart, J.Y. and Meyer, J.A. Reactive rules learning in a motivationally autonomous robot. IEEE Transactions on Man, Systems and Cybernetics - Part B. 1996b, 3, 381-395.

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

Guillot, A. et Meyer, J.A. Synthetic Animals in Synthetic Worlds. In Kunii et Luciani. Synthetic Worlds. Springer Verlag. In Press.

Holland, J. Adaptation in Natural and artificial Systems. The University of Michigan Press. 1975.

Holland, J., Holyoak, K., Nisbett, R. and Thagard, P. Induction. Processes of Inference, Learning and Discovery. The MIT Press. 1986.

Husbands, P. and Meyer, J.A. (Eds.). Proceedings of The First European Workshop on Evolutionary Robotics - EvoRobot'98. Springer Verlag. 1998.

Kodjabachian, J. and Meyer, J.A. Evolution and development of control architectures in animats. Robotics and Autonomous Systems. 1995, 16, 161-182.

Kodjabachian, J. and Meyer, J.A. Evolution and Development of Modular Control Architectures for 1-D Locomotion in Six-legged Animats. Connection Science. 1998a. In Press.

Kodjabachian, J. and Meyer, J.A. Evolution and Development of Neural Controllers for Locomotion, Gradient-Following, and Obstacle-Avoidance in Artificial Insects. IEEE Transactions on Neural Networks. 1998b. In Press.

Maes, P., Mataric, M., Meyer, J.A., Pollack, J. and Wilson, S.W. (Eds.). From Animals to Animats 4. Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior. The MIT press/Bradford Books. 1996.

Matarić, M. Navigating With a Rat Brain: A Neurobiologically-Inspired Model for Robot Spatial Representation. In Meyer and Wilson (Eds.) From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. The MIT Press/Bradford Books. 1991.

Meyer, J.A. From Natural to Artificial Life: Biomimetic Mechanisms in Animat Designs. Robotics and Autonomous Systems. 1997, 22, 3-21.

Meyer, J.A. Artificial Life and the Animat Approach to Artificial Intelligence. In Boden, M. Artificial Intelligence. Academic Press. 1996.

Meyer, J.A. The Animat Approach to Cognitive Science. In Roitblat, H. et Meyer, J.A. Comparative Approaches to Cognitive Science. MIT Press. 1995.

Meyer, J.A., Roitblat, H. and Wilson, S. From Animals to Animats 2. Proceedings of the Second International Conference on Simulation of Adaptive Behavior. The MIT Press/Bradford Books. 1993.

Meyer, J.A. and Wilson, S. From Animals to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior. The MIT Press/Bradford Books. 1991.

Reynolds, C. W. Flocks, Herds, and Schools: A Distributed Behavioral Model. Computer Graphics. 1987, 21, 25-34.

Roitblat, H., et Meyer, J.A. Comparative Approaches to Cognitive Science. The MIT Press. 1995.

Sims, K. Evolving 3D morphology and behavior by competition. In Brooks and Maes (Eds.) Proceedings of the Fourth International Workshop on Artificial Life. The MIT Press/Bradford Books. 1994.

Trullier, O. and Meyer, J.A. Biomimetic Navigation Models and Strategies in Animats. AI Communications. 1997a, 10, 79-92.

Trullier, O. and Meyer, J.A. Place sequence learning for navigation. In Gerstner et al. (Eds.) Artificial Neural Networks. ICANN'97. Springer Verlag. 1997b.

Trullier, O. and Meyer, J.A. Animat Navigation Using a Cognitive Graph. In Blumberg et al. (Eds.). From Animals to Animats 5. Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior. The MIT press/Bradford Books. 1998.

Trullier, O., Wiener, S., Berthoz, A. and Meyer, J.A. Biologically- based artificial navigation systems: Review and Prospects. Progress in Neurobiology. 1997, 51, 483-544.