Comparing a voting-based policy with winner-takes-all to perform action selection in motivational agents

Orlando Avila-García and Lola Cañamero

Adaptive Systems Research Group
Department of Computer Science, University of Hertfordshire
College Lane, Hatfield, Herts AL10 9AB, UK
O.Avila-Garcia@herts.ac.uk, L.Canamero@herts.ac.uk

Abstract. Embodied autonomous agents are systems that inhabit dynamic, unpredictable environments in which they try to satisfy a set of time-dependent goals or motivations in order to survive. One of the problems that this implies is action selection, the task of resolving conflicts between competing behavioral alternatives. We present an experimental comparison of two action selection mechanisms (ASM), implementing "winner-takes-all" (WTA) and "voting-based" (VB) policies respectively, modeled using a motivational behavior-based approach. This research shows the adequacy of these two ASM with respect to different sources of environmental complexity and the tendency of each of them to show different behavioral phenomena.

1 Introduction

In the mid 80's, a new approach emerged in artificial intelligence (AI) to study intelligence in the context of "complete", embodied autonomous agents [4]. This new paradigm is highly inspired in biology, animal behavior (ethology), neuroscience and evolutionary biology, and has been termed "behavior-based AI" or "animat [1] approach" [13, 11]. The animat approach is appropriate for the class of problems that require a system to autonomously fulfill several time-dependent goals or motivations in a dynamic, unpredictable environment. The main problem in this area is to build architectures for an autonomous agent that will result in the agent demonstrating adaptative, robust and effective behavior. Specifically, two related subproblems have to be solved [8]: The problem of action (or behavior) selection and the problem of learning from experience. This research is focused on the first issue.

Action selection consists in making a decision as to what behavior to execute in order to satisfy internal goals and guarantee survival in a given environment and situation. This implies resolving conflicts between competing behavioral alternatives. We have modeled two action selection mechanism (ASM) using a motivational behavior-based approach that follows the architecture proposed in [5]. This means that the agent's behavior is driven by motivational states—impulses to action based on bodily needs. There are several ways in which the set of motivations can lead to the execution of one behavior, since the same behavior may be able to satisfy more than one motivation

[1] Animat is shorthand for "artificial animal"

and the satiation of one motivation may lead the arousal of another. In this paper, we present an experimental comparison of two action selection architectures implementing approaches that are often regarded as opposite in the literature: “winner-takes-all” (WTA) and “voting-based” (VB). With a WTA policy the animat will execute the behavior that satisfies its highest motivation in the best way. This means that only the highest motivation drives behavior selection. With a VB policy, all the motivations have influence on the final selection since the animat will execute the behavior that best satisfies a subset of motivations at the same time. In previous work [3, 6] we did a systematic study and comparison of four architectures that differed along a few relevant parameters; however, only the differences between the two architectures we present here (WTA and VB) were significant enough, giving rise to more stable results and differences in behavior. Drawing on the notion of viability [1], used within the animat approach for different purposes [11, 12, 2], we have defined several complementary viability indicators to assess behavior selection performance, and we have analyzed our results in terms of some interesting behavioral phenomena that [7] proposes as desirable features of architectures to achieve flexible behavior selection.

2 The Valimar Environment

In order to test our architectures, we have created a typical behavior selection environment, called Valimar, in which our robot must select among and perform different activities in order to survive. The platform we have used is Webots 3.0 (www.cyberbotics.com), a very realistic 3D mobile robot simulator allowing users to create different environments with continuous space and complex physics. For these experiments, we have used Kheperas fitted with a color camera on their top to create two species of robots—Nessas (green Kheperas) and Enemies (red Kheperas). *Nessas* are more complex creatures used to implement the two behavior selection architectures described in Section 3. *Enemies* have the same sensors and actuators as Nessas, but they have a much simpler architecture and behavior (their main goal is to attack Nessas), since their only role is to introduce dynamism in the environment. Neither Nessas nor Enemies can remember the location of objects, and they also lack any planning capabilities. Valimar is surrounded by a wall and contains cylindrically-shaped objects of different colors: food (yellow) and water (blue) sources, nests (purple), obstacles (gray), dull blocks (red), and our two species of robots. Since Enemies and dull blocks have the same color, Nessas can mistake one for another.

3 Architectures for Behavior Selection

The architectures we have studied are neither strictly flat (parallel) nor hierarchical (structured), but a combination of both that lies more on the flat side. They consist of two layers—motivational and behavioral—that lead to a two-step computation of intensity. This computation is parallel within each layer, but motivational intensity must be computed prior to the calculation of behavioral intensity. Both architectures have the same elements but differ in how these are linked together in their arbitration mechanism.

3.1 Elements

Sensing and acting. The robots are equipped with the following *external sensors*: eight infrared proximity sensors and eight binary collision sensors; a radio emitter/receiver used to transmit and detect the attack of another robot (pain_e); and a color camera returning a RGB pattern of 90×90 pixels used to detect direction and discriminate objects. Object learning and recognition is performed by a combination of three ART1 neural networks, each of them specialized to detect patterns in one of the RGB components. In addition to external sensors, we have programmed *internal sensors* to perceive the values of physiological variables. Navigation (including obstacle avoidance) is controlled by a neural network that improves over time through Hebbian learning.

Physiology. The robots have a synthetic physiology of survival-related essential variables that must be kept within a range of values for the robot to stay alive. They thus define a physiological space [10] or viability zone [11] within which survival (continued existence) is guaranteed, whereas transgression of these boundaries leads to death. Nessa's essential variables are: damage, energy, glucose, moisture, internal pain [2] (pain_i), and stress.

External stimuli. In addition to internal variables, behavior selection is also influenced by the presence of external stimuli that affect the motivational state of the robot. There are six types of external stimuli to which Nessas can react: the different (colors of the) objects in the environment plus external pain (pain_e).

Motivations. Motivations constitute urges to action based on bodily needs related to self-sufficiency and survival. They implement a homeostatic process to maintain an essential physiological variable within a certain range. A feedback detector generates an error signal—the drive—when the value of this variable departs from its ideal value (setpoint), and this triggers inhibitory and excitatory controlling elements (in this case, the execution of behaviors) to adjust the variable in the adequate direction. The error is a number normalized in the interval $[0, 1]$, where 0 indicates no error and 1 results when the actual value of the variable overflows/underflows the upper/lower limit, in which case the robot dies. Each motivation receiving an error signal from its feedback detector receives an intensity (activation level) proportional to the magnitude of the error. Several motivations can be active at the same time, with varying degrees of intensity. Nessa's motivations are characterized by: a controlled (essential) physiological variable, a drive to increase or decrease the level of the controlled variable, an (external) incentive stimulus that can increase the motivation's intensity, and a behavioral tendency of approach or avoidance towards the stimulus. Table 1 (left) shows Nessas' motivations with their drives and incentive stimuli.

Behaviors. Our behaviors are coarse-grained subsystems implementing different competencies, similarly to those proposed in [7, 5]. Following the usual distinction in ethology [9], Nessas have consummatory (goal-achieving) and appetitive (goal-directed) behaviors. A consummatory behavior is executed only if it has been selected by the motivational state of the robot and its incentive stimulus is present. If the stimulus is not present, an appetitive behavior is activated to search for it. The execution of a behavior

[2] Pain has a double characterization as internal and external stimulus. Internal pain receives its value from externally felt pain but has more inertia, decreasing more slowly and lasting longer.

has an impact on (increases or decreases) the level of specific physiological variables. Behaviors can be activated and executed with different intensities that depend on the intensities of the motivations related to them. The intensity with which a behavior is executed affects motor strength (speed of the wheels) and the modification of physiological variables (and hence the duration of the behavior). Table 1 (right) shows Nessas' behaviors.

Motivation	Drive	Incentive stimulus
Confusion	$\downarrow stress$	nest
Excitement	$\downarrow energy$	enemy
Fatigue	$\uparrow energy$	nest
Hunger	$\uparrow glucose$	food
Overmoisture	$\downarrow moisture$	none
Overnutrition	$\downarrow glucose$	none
Repair	$\downarrow damage$	nest
Self-protection	$\downarrow pain_i$	enemy, $pain_e$
Thirst	$\uparrow moisture$	water

Behavior	Stimulus	Effects
Avoid	none	$\downarrow energy, \downarrow glucose, \downarrow moisture, \downarrow pain, \uparrow stress$
Attack	enemy	$\downarrow energy, \downarrow glucose, \downarrow moisture, \downarrow pain, \downarrow stress$
Drink	water	$\downarrow energy, \downarrow glucose, \uparrow moisture$
Eat	food	$\downarrow energy, \uparrow glucose, \downarrow moisture$
Rest	none	$\uparrow energy, \downarrow glucose, \downarrow moisture$
RunAway	enemy	$\downarrow energy, \downarrow glucose, \downarrow moisture, \downarrow pain, \uparrow stress$
Search	none	$\downarrow energy, \downarrow glucose, \uparrow stress, \downarrow moisture$
Sleep	nest	$\downarrow damage, \uparrow energy, \downarrow glucose, \downarrow stress, \downarrow moisture$
Wander	none	$\downarrow energy, \downarrow glucose, \downarrow moisture$

Table 1. Nessas' motivations (left) and behaviors (right; names in italics indicate appetitive behaviors, the rest are consummatory).

3.2 Arbitration mechanisms

Motivations and behaviors are connected indirectly through physiological variables, i.e., motivations take into account the effects that the execution of a behavior will have on the physiology to make their selection. Our architectures differ in the way these elements interact to select the behavior that the robot must execute. In the WTA approach, the main selection is made at the level of motivations, and a single motivation is selected to be in charge of selecting the behavior that best satisfies it. In the VB approach the main decision is made at the level of behaviors, and each behavior receives activation from all the motivations that will result affected by its execution, and behavior selection is postponed until behavioral intensity has been computed. In this case, all the behaviors are considered for the final selection, and the robot can satisfy several goals simultaneously.

The behavior selection loops are as follows.
Behavior selection loop in WTA. At every cycle (simulation step):

1. The winner motivation j_{winner} is calculated.
   (a) For each motivation j:
      i. Compute the intensity of the drive as proportional to the error of its controlled variable (e_{vj}).
      ii. Compute the effect of the presence of external stimuli on the intensity of the motivation: $a_j = \sum(s_k \times u_{jk})$, where s_k is the intensity of stimulus k, and u_{jk} is the weight between j and k.
      iii. $m_j = e_{vj} + a_j$ is the final intensity of j.
   (b) The motivation with highest intensity is selected.
2. The intensity of each behavior linked (through the physiology) with the winner motivation is computed as $b_i = m_{j_{winner}} \times f_{iv}$, where b_i, $m_{j_{winner}}$ are the intensities of behavior i and the winner motivation, respectively, and f_{iv} is the effect that the execution of behavior i has on v, which is the physiological variable controlled by j_{winner}.
3. The behavior with highest intensity is selected to be executed.

Behavior selection loop in VB. At every cycle (simulation step):

1. Calculate the intensity of each motivation j:
   (a) Compute the intensity of the motivation's drive as proportional to the error of its controlled variable (e_{vj}).
   (b) Compute the effect of the presence of external stimuli on the intensity of the motivation j: $a_j = \sum(s_k \times u_{jk})$, where s_k is the intensity of stimulus k, and u_{jk} is the weight between j and k.
   (c) $m_j = e_{vj} + a_j$ is the final intensity of j.
2. The intensity of each behavior is computed as $b_i = \sum(m_j \times f_{iv})$, where b_i, m_j are the intensities of behavior i and motivation j, respectively, and f_{iv} is the effect that the execution of behavior i has on v, the physiological variable controlled by j.
3. The behavior with highest intensity is selected to be executed.

4 Experiments

4.1 Viability Indicators

We have used the notion of viability [1] as general criterion to assess behavior selection performance. The values (upper and lower boundaries) of the robot's essential variables define a physiological space [10] or viability zone [11] within which survival (continued existence) is guaranteed, whereas transgression of these boundaries leads to death. The behavior of the robot is thus viable when it keeps the values of the essential variables within the boundaries of the physiological space.

In our opinion, however, this notion of viability is too vague to provide a direct criterion to measure the goodness or performance of a behavior selection architecture, as it leaves several possibilities open. For example, the performance of a behavior selection architecture can be simply assessed in terms of the time it allows the robot to remain viable (survive) in a given environment. Longer life spans usually indicate better behavior selection performance but this correlation is not necessarily straightforward, since the "life quality" or "well-being" of the robot can be very different during its life span depending on how viability is preserved during that period. A robot can live a long life with poor "life quality" if the values of its essential variables are kept close to the critical zone (near the boundaries) of the physiological space for a long period. On the contrary, it can live a shorter life that ended due to "accidental" factors (e.g., the attack of one or several predators or the absence of resources) during which it had good viability in terms of "well-being" if the values of its essential variables remained close to their ideal values. "Life quality" or "well-being" can also have different interpretations. It can for example be measured in terms of global internal stability or "comfort" that takes into account the average level of satisfaction of all the essential variables simultaneously. It can also be seen in terms of how "balanced" the satisfaction of the different physiological needs is, since the same level of comfort and life span can be achieved e.g., by satisfying one or few motivations to a high degree while keeping the rest close to the critical zone, or by satisfying all the motivations in a more homogeneous way.

We have therefore used three indicators of viability to measure and compare the performance of our architectures:

Life span: The time that the robot survived (remained viable) during each run normalized with the total simulation time, $S_{life} = t_{life}/t_{simul}$, where t_{life} is the number of simulation steps that the robot lived and t_{simul} is the total simulation time measured in number of simulation steps.

Overall comfort: The average level of satisfaction of all the essential variables, measured at each step as $c_{step} = 1 - (Err/n)$, where Err is the total sum of errors of the robot's physiological variables normalized between $[0, 1]$ with n (the worst error possible in each step), which corresponds to the sum of the intensities of the motivations' drives ($Err = \sum(e_{vj})$), and n is the number of compatible motivations[3]. Average control for a run is given by $C = \sum_1^{t_{life}}(c_{step})/t_{life}$, where t_{life} is the number of simulation steps that the robot lived.

Physiological balance: The homogeneity with which the different physiological needs are satisfied, measured at each step as $b_{step} = 1 - (Unb/max_{unb})$, where Unb is the variance of the errors of the robot's physiological variables normalized between $[0, 1]$ with max_{unb} (the worst variance possible in each step), which corresponds to the variance of the intensities of the (compatible) motivations' drives ($Unb = \sigma^2(e_{vj})$). Average balance for a run is given by the mean of steps. Average physiological balance for a run is given by $B = \sum_1^{t_{life}}(b_{step})/t_{life}$.

4.2 Method

We have explored the effects of three sources of environmental complexity on the behavior and performance of our architectures: *Amount of objects* (the more populated a world is, the more complex navigation and perception are), *availability of resources* (the fewer resources a world contains, the more difficult and costly it is to obtein them), and *dynamism* (Enemies that can attack and kill Nessas, and also hamper foraging activities). To vary these sources of complexity, we created four Valimar settings (Table 2) with different elements sparsely distributed across the world.

We tested the two architectures in 30 sets of runs for each Valimar world (V1, V2, V3, V4), each set being comprised of a run for each architecture. This made a total of 240 runs (about 45 hours), each run lasting 10,000 simulation steps.

[3] Two motivations are compatible if they do not control the same variable in opposite directions.

Worlds	Water sources	Food sources	Nests	Obstacles	Dull blocks	Enemies	Nessas
V1	4	4	2	2	2	0	1
V2	1	1	1	0	1	0	1
V3	2	2	1	2	0	1	1
V4	2	2	1	2	0	2	1

Table 2. Elements of the four Valimar worlds.

4.3 Results

Figure 1 shows average performance of both architectures in the four Valimar worlds in terms of life span (left), overall comfort (center), and physiological balance (right).

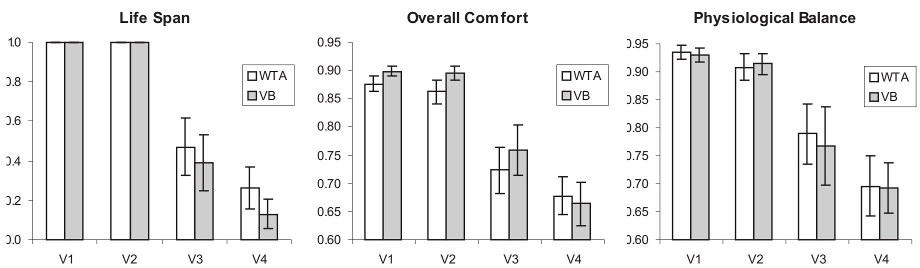

Fig. 1. Average performance of the two architectures in the four Valimar worlds in terms of life span (left), overall comfort (center), and physiological balance (right). Mean confidence intervals shown are calculated with a 95% confidence level.

Analysis of variance (using ANOVA Single Factor) showed that the difference of each architecture across worlds is statistically highly significant (99%) in terms of our three viability indicators. Considering both architectures, these three indicators show significant differences between static (V1, V2) and dynamic (V3, V4) worlds.

In static worlds, the life span obtained by the both architectures is very similar. In terms of overall comfort, there is a clear tendency that shows that VB obtains better results than WTA, although they obtain very similar results in terms of physiological balance. In the dynamic world with one Enemy, big confidence intervals indicate that similarities among architectures in terms of life span might be due to external factors (e.g., the ability of the Enemy to find and attack Nessa) to a big extent. A clear difference appears in the world with two Enemies, where WTA outperforms the VB architecture. Mean confidence intervals of the other indicators show that dynamism affects them too much to be significant when analyzed separately in dynamic worlds, and results have to be nuanced by taking into account life span. It would seem, then, that the trend observed in static worlds appears in dynamic worlds as well.

4.4 Discussion

Let us now discuss some phenomena commonly studied in animal decision making (see e.g., [9]) that we have observed in our simulations, and that allow us to better understand the differences between architectures and how they deal with different properties of the environment. Some of these phenomena have been proposed by [7] as desirable features of architectures to achieve flexible behavior selection.

Reactivity (openness). WTA is more reactive to changes in the environment than VB because only one motivation drives behavior selection. Since with a VB policy the negative effects of executing a behavior that satisfies a motivation are also taken into account, there can be mutual inhibition among motivations that can diminish the extent to which the behavior selected contributes to correct physiological imbalances. This means that it usually takes longer for VB to satisfy the most urgent physiological need and to react to external changes. Therefore, WTA deals better with quick and unpredictable (physiological and external) environmental changes.

Stability of a sequence of behaviors occurs when behavioral intensities are (nearly) similar for all the sequence. A non-stable sequence results in sudden changes [4] in the robot's velocity and modification of its variables. WTA, being more reactive as it uses a single motivation to drive behavior selection, was the less stable architecture.

Displacement behaviors appear in a competition between two motivations highly activated when a third motivation less activated and unrelated to the current context drives behavior selection. This phenomenon was only observed in VB, as its "voting-based" policy, together with the fact that links between motivations and behaviors can be positive (excitatory) or negative (inhibitory), can lead to *mutual inhibition* (cancellation) of two motivations with high intensity.

Opportunism management. WTA is less opportunistic than VB as a consequence of the way in which the influence of external stimuli are taken into account. In WTA each external stimulus is computed only once to calculate the intensity of the winner behavior because it only influences the highest activated motivation. However in VB the same external stimulus can be computed more than once—as many times as motivations contribute towards calculating the intensity of a particular behavior. This may explain the fact that the performance of VB in terms of overall comfort seems to be the same in static worlds in spite of the shortage of resources in V2, while with a WTA policy this indicator is affected.

Situations of self-protection are more difficult to deal with when Nessas are executing a consummatory behavior next to a resource, as they are more exposed to Enemies, which can attack them on the back, where they are not detected, and block them against resources, as shown in Figure 2. This roughly corresponds to what [7] denotes as *varying attention*—the fact that animals pay lower attention to danger when they are in an extreme motivational state, e.g. very hungry. WTA, being more reactive (like a simple emotional system), is the best in this case, while VB is more often trapped in these situations due to its lower reactivity.

[4] Recall that the intensity of the winner behavior has an impact on motor strength and on how physiological variables are modified (and therefore on the duration of the behavior).

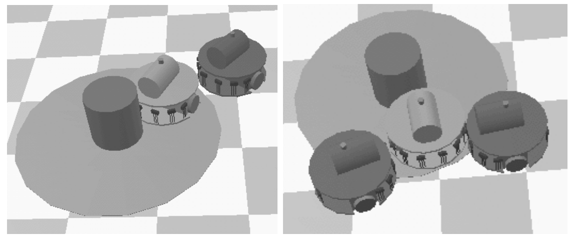

Fig. 2. Nessas (lighter Kheperas) attacked and blocked by Enemies next to a nest.

Maximizing efficiency of behavioral choice, i.e., executing the behavior that best contributes towards approaching the ideal zone of the physiological space, is usually considered an important desideratum for behavior selection mechanisms. The extent to which maximization is achieved is directly related to the amount of "information" available to each architecture about the effect that the execution of the behavior will have on the physiology. WTA uses less information than VB, since it only takes into account the effects of behavior execution on one motivation to make the selection, while VB considers the (positive and negative) effects of behavior execution on all motivations. Executing a behavior that satisfies several motivations simultaneously (VB) is more efficient in static worlds in terms of overall comfort than executing a behavior that only satisfies one goal (WTA), as reflected by our results in Figure 1. However, maximizing efficiency does not necessarily lead to better results in terms of physiological balance and life span, again referred to static worlds, but leads to poorer reactivity and therefore presents drawbacks in dynamic worlds, as seen above.

5 Conclusion and Future Work

We have presented a study comparing the performance of two different motivated behavior selection architectures (WTA and VB) in environments with varying degrees and types of complexity. The indicators used have shown significant differences in the behavior of both architectures between static and dynamic worlds. Interestingly, WTA and VB obtained highly similar results in terms of physiological balance, and complementary ones in terms of life span and overall comfort. We have demonstrated that a "winner-takes-all" policy is more adequate for unpredictable and dynamic environments since it is the most reactive, while a "voting-based" policy is more adequate for static and predictable worlds since in each step it takes the most efficient decision.

To continue this study we envisage several directions for future work. First, we plan to perform quantitative analyses of the different behavioral phenomena observed (opportunism, displacement behaviors, etc.) in order to better understand the differences between architectures. Second, we intend to study the optimization mechanisms underlying "winner-takes-all" versus "voting-based" architectures, to better understand our results in terms of overall comfort and physiological balance. Third, we would like to extend our model of motivational states to consider other aspects that contribute to homeostatic regulation in addition to behavior execution, in particular physiological regulation. Fourth, we want to increase the complexity of the world in terms of dynamism to take into account not only the presence of enemies but also extinction and mobility of resources. Finally, we plan to add basic emotions to our behavior selection architectures, following the design proposed by [5], and compare the performance of both architectures with and without emotions in both more static and highly dynamic worlds.

Acknowledgments

We are grateful to Elena Hafner for her contribution to earlier stages of this research. Orlando Avila-García is supported by the University of Hertfordshire through an EPSRC scholarship.

References

1. Ashby, W.R. (1952). *Design for a Brain: The Origin of Adaptive Behavior*. London: Chapman and Hall.
2. Aubin, J.P. (2000). Elements of Viability Theory for Animat Design. In J.A. Meyer, A. Berthoz, D. Floreano, H. Roitblat, and S.W. Wilson, eds., *Proc. Sixth Intl. Conf. on Simulation of Adaptive Behavior*, 13–22. Cambridge, MA: MIT Press.
3. Avila-García, O., Hafner, E., and Cañamero, L. (2002). Relating Behavior Selection Architectures to Environmental Complexity. In *Proc. Seventh Intl. Conf. on Simulation of Adaptive Behavior*. Cambridge, MA: MIT Press (in press).
4. Brooks, R.A. (1991). Intelligence without reason. In J. Mylopoulos, R. Reiter (eds.), *Proc. 12th Intl. Joint Conf. on Artificial Intelligence (IJCAI)*, 569–595. San Mateo, CA: Morgan Kaufmann.
5. Cañamero, L.D. (1997). Modeling Motivations and Emotions as a Basis for Intelligent Behavior. In W.L. Johnson, ed., *Proc. First Intl. Conf. on Autonomous Agents*, 148–155. New York: ACM Press.
6. Cañamero, L.D. and Avila-Garcia, O. (2002). A comparison of Behavior Selection Architectures Using Viability Indicators. Submitted to the *International Workshop on Biologically-Inspired Robotics: The Legacy of W. Grey Walter*. 14–16 August 2002, Bristol, UK.
7. Maes, P. (1991). A Bottom-Up Mechanism for Behavior Selection in an Artificial Creature. In J.A. Meyer and S.W. Wilson, eds. *Proc. First Intl. Conf. on Simulation of Adaptive Behavior*, 238–246. Cambridge, MA: MIT Press.
8. Maes, P. (1995). Modeling Adaptive Autonomous Agents. In C.G. Langton, ed., *Artificial Life: An Overview*, 135–162. Cambridge, MA: MIT Press.
9. McFarland, D. (1995). *Animal Behaviour*, 3rd. edition. Harlow, England: Longman.
10. McFarland, D. and Houston, A. (1981). *Quantitative Ethology*. London: Pitman.
11. Meyer, J.A. (1995) Artificial Life and the Animat Approach to Artificial Intelligence. In M. Boden, ed., *Artificial Intelligence*. Academic Press.
12. Steels, L. (1997). A Selectionist Mechanism for Autonomous Behavior Acquisition. *Robotics and Autonomous Systems*, 20(2/4): 117–131.
13. Wilson, S.W. (1985). Knowledge Growth in an Artificial Animal. In J.J. Grefenstette (ed.), *Proc. of the First Intl. Conf. on Genetic Algorithms and their applications*, 16–23. Hillsdale, NJ. Lawrence Erlbaum.