Action-Selection in Hamsterdam: Lessons from Ethology

Bruce Blumberg
MIT Media-Laboratory
E15-305F
20 Ames St., Cambridge Ma. 02139
bruce@media.mit.edu

Abstract

A computational model of action-selection is presented, which by drawing on ideas from Ethology, addresses a number of problems which have been noted in models proposed to date including the need for greater control over the temporal aspects of behavior, the need for a loose hierarchical structure with information sharing, and the need for a flexible means of modeling the influence of internal and external factors. The paper draws on arguments from Ethology as well as on computational considerations to show why these are important aspects of any action-selection mechanism for animats which must satisfy multiple goals in a dynamic environment. The computational model is summarized, and its use in Hamsterdam, an object-oriented tool kit for modeling animal behavior is discussed briefly. Results are presented which demonstrate the power and usefulness of the novel features incorporated in the algorithm.

1. Introduction

The problem of action-selection is central to the larger problem of building animats that function robustly in complex and dynamic environments. Specifically, the problem for the animat is to choose the "most appropriate" set of motor actions to perform from its repertoire of potential actions, given some set of internal needs and external stimuli. In the case of animals, "most appropriate" may be measured against some performance criteria such as maximizing their reproductive success. In the case of animats, Maes and others [Maes93, Tyrrell93, McFarland93, Wilson85] have discussed a variety of criteria against which the appropriateness of a set of actions might be measured.

A number of action-selection algorithms have been proposed by ethologists [Lorenz73, Tinbergen50, Baerends76, Davey89, Dawkins76, Ludlow76,80, McFarland75, Toates83] as well as computer scientists [Maes90, Brooks86, Rosenblatt & Payton89, Tyrrell93]. The ethological models are often conceptual rather than computational models and much is left to the discretion of the reader as to how one might implement such a scheme. With the exception of Tyrrell, the computational models proposed by computer scientists bear little resemblance to the models proposed by classical ethologists such as Baerends and Tinbergen. Nonetheless, the computational models have been used successfully in a number of applications, and their very success has called into question traditional AI approaches to planning[Maes93].

At least one researcher[Tyrrell93], however, has noted the difficulty of applying, without modification, the models of Brooks[Brooks86], Maes[Maes90], and Rosenblatt and Payton[Rosenblatt89] to the problem of modeling action-selection in animats whose behavior is to mirror that of real animals. While one may view the difficulties noted by Tyrrell as being specific to the problem of modeling animal behavior, a more serious and general concern may be lurking. Namely, previously proposed computational models of action-selection may be missing elements essential to robust action-selection in animats whose behavioral complexity (i.e. in the number of needs which must be met, and in the range of potential actions which may be taken) approaches that of real animals.

This paper presents an ethologically-inspired computational model of action-selection which improves on existing models in three areas. These include:

The Need for Greater Control Over the Temporal Aspects of Behavior

Computational models of action-selection proposed to date have difficulty providing "just the right amount of persistence". That is, it is difficult to control the temporal aspects of behavior so as to arrive at the right balance between too little persistence, resulting in dithering among activities, and too much persistence so that opportunities are missed or that the animat mindlessly pursues a given goal to the detriment of other goals. The computational model presented in this paper addresses this problem by incorporating an explicit model of inhibition and fatigue first proposed by Ludlow [Ludlow76,80]. The benefits of this approach include:

- It provides control over the level of persistence associated with an active activity, thus reducing the chances of dithering, while still allowing for opportunistic behavior.

- It provides a natural mechanism for modeling the phenomena of time-sharing in which low priority activities are given a chance to execute despite the presence of a higher priority activity. More generally it reduces the chances of pursuing a single goal to the detriment of all others.

- It provides a robust mechanism for implementing winner-take-all arbitration among activities.

The Need for a Loose Hierarchical Structure with Information Sharing

Drawing on the tradition of hierarchical models of behavior from Ethology, Tyrrell[Tyrrell93] has convincingly made the case for computational models of action-selection which incorporate a loose hierarchical structure. However, he departs from the classical Ethological view that at any one time, only one behavior system is being expressed in the movements of the animal. He proposes a model, derived from the work of Payton & Rosenblatt [Rosenblatt89], in which all of the nodes in the hierarchy can influence the subsequent behavior of the creature.

By contrast, the computational model presented in this paper implements a winner-take-all system consistent with traditional ethological thinking, but which nonetheless allows losing activities to express their preferences in the form of recommendations to the winning activity. The winner may use these recommendations as it sees fit. The benefits of this approach include:

- It provides a mechanism for information-sharing which potentially allows the system to arrive at compromise solutions which ultimately may be more efficient for the creature, and thus avoids a potential problem associated with winner-take-all approaches.

- Relative to Tyrrell's approach, it preserves the attractive divide-and-conquer attribute of more traditional hierarchies, and thus may scale better. It is also less dependent on the careful tuning of parameters.

The Need for a Common Currency and a Flexible Means of Modeling

The third contribution of the computational model presented here is that it explicitly includes the concepts of Releasing Mechanisms and Endogenous Variables from Ethology. It treats them as abstractions for more complicated processes and thus allows the designer to model them accordingly. Their values, however, are expressed as continuous quantities in a common currency. This is distinguished from some previous approaches which model the presence of external stimulus as a predicate [Maes90]. The benefits of this approach include:

- A natural way to model the phenomena of motivational isoclines[McFarland75,76] in which differing levels of internal motivation and external stimulus result in the same action.

- It does not require the animat designer to model all systems in a particular way, for example, as strictly homeostatic systems.

By addressing the three issues described above, we have developed an action-selection algorithm which both improves on existing algorithms and which is well suited to animats that must satisfy multiple internal needs in a dynamic and unpredictable environment.

In section 2, we discuss some lessons from Ethology and their computational implications. In particular, we focus on the importance of modeling inhibition and fatigue, the importance of hierarchical organizations of behavior with information sharing and the use of a common currency with which to express internal needs and external opportunities. The computational model is presented in section 3. We then describe Hamsterdam, a tool kit for building animated creatures which incorporates the proposed model of action-selection. In section 5 we present results which show that the algorithm does address the problems described above. We conclude with a discussion of limitations and areas for future work.

2. Lessons from Ethology and Computational Implications

2.1 The Importance of Inhibition and Fatigue

Inhibition plays an important role in ethological models of action-selection and is used to explain some of the temporal aspects of behavior. Ethologists generally agree that animals engage in one behavior at a time [Tinbergen50, Lorenz73, McFarland76, McCleery83]. Yet animals typically do not mindlessly pursue an activity indefinitely to the detriment of other needs. Indeed, animals sometimes appear to engage in a form of time-sharing [McFarland74,93], in which low priority activities are given a chance to execute, despite the presence of a higher priority activity. While animals typically do not dither between multiple activities they will nonetheless interrupt a behavior when another behavior becomes significantly more appropriate. Models of inhibition and fatigue are frequently used to explain these aspects of animal behavior. Thus, it is essential that an action-selection mechanism for animats include an explicit model of inhibition and fatigue.

The model used in the action-selection algorithm presented below was first proposed by Ludlow [Ludlow 76, 80]. However, our system is the first to use it in a complete action-selection algorithm.

In Ludlow's model, an activity such as feeding, or drinking has a value which is based on the sum of its relevant internal and external factors less inhibition it receives from competing activities. Competing activities are mutually inhibiting, where a given activity i inhibits activity j by an amount equal to activity i's value times an inhibitory gain k_{ji}. The higher the gain, the greater the inhibition, and effectively the greater the persistence of the active activity. Ludlow's observation was that if (a) activities are mutually inhibiting, (b) the inhibitory gains are restricted to be greater than 1, and (c) value of activities is restricted to being zero or greater, then this model would result in a winner-take-all system, in which only one activity would have a non-zero value once the system stabilized. In practice we have found this to be true. It is exceedingly rare for it not to converge on a solution, particularly if the inhibitory gains are above 2.0.

An activity with a non-zero value is said to be active. In his model, there is a direct feedback loop between an activity and the endogenous factors upon which it depends so that when an activity is active, it reduces the value of those endogenous factors by some activity-specific gain times the value of the activity. In our implementation of his algorithm we relax this assumption so we can use his model in the context of a hierarchical structure. For example, the value of the feeding activity depends on the level of hunger. The feeding activity, however, includes a number of more specific activities including searching for food, handling it, and finally consuming it. It is only when the animal engages in the later activity that the level of hunger is reduced. Thus, the feeding activity relies on another activity, lower in the hierarchy, to reduce the value of one of the endogenous variables on which it depends.

An active activity will stay active until its value or the value of one of the activities with which it competes changes by an amount proportional to the inhibitory gain. For example, if eating is active, and it inhibits drinking with a gain of 2.0, the value of eating must fall to half the value of drinking before drinking will become active. The value of eating would fall, for example, in response to consuming food. Alternatively, the value of drinking could rise significantly in response to passing a water source on the way to the food.

By modifying inhibitory gains, the level of persistence of a given activity relative to those with which it competes may be adjusted accordingly. An activity associates a specific inhibitory gain with each activity it inhibits. When the gains are low, the system tends to dither between different activities with the result that the system takes longer to reach satiation levels. At higher gains, the active activity shows more persistence with the effect that satiation is reached sooner. However, persistence comes at the expense of opportunism.

More is needed than simple inhibitory gains. For example, use of high gains may result in lower priority activities never becoming active. For example, feeding may always be of higher absolute priority than body maintenance, yet it is important that the creature be able to periodically interrupt searching for food so it may clean itself. As part of his model, Ludlow suggested a mechanism for modeling behavior-specific fatigue and used this, in conjunction with his model of inhibition, to implement time-sharing.

In Ludlow's model, a level of fatigue is associated with every activity. The level of fatigue is influenced by a number of factors, however when an activity is active the level of fatigue increases in proportion to the activity's value (thus implementing another feedback loop), which reduces the value of an active activity over time. When the activity is no longer active, the fatigue decays toward zero, and the value of the activity rises. In his paper [Ludlow80], Ludlow uses his model to replicate some of the results of McFarland's time-sharing experiments [McFarland74].

The computational model presented below draws heavily from Ludlow's model. One important difference is that Ludlow envisioned a flat structure of competing activities all of which were mutually inhibiting. We have chosen to embed his model in a loose hierarchical organization in the spirit of Baerends [Baerends76] or Tinbergen [Tinbergen50] in which an activity competes only with a subset of the other activities, namely those at its same level in the hierarchy. We now turn to a discussion of the implications of using a hierarchical structure.

2.2 The Importance of Hierarchies with Information-Sharing

In Dawkins' view, a hierarchical structure represents one of the essential organizing principles of complex behavior [Dawkins 76] and this view is echoed by many ethologists [Lorenz73, Baerends76, Tinbergen50, Gallistel80, Davey89, Timberlake89 and McFarland75]. Baerends' model represents one example of this approach. The esssential idea is that an animal is considered to have a number of activity systems, or collections of activities each organized to address a specific biological function. The activity systems are organized as loose-overlapping hierarchies. At any one time, only one system is being expressed in the movements of the animal. For example, preening and nesting represent two competing activity systems in herring gulls, each of which has a number of subordinate collections of activities such as settling, building, trimming and bathing. These in turn have motor actions associated with them. Releasing mechanisms (an abstraction for whatever perceptual process signals a biologically important object or event) and endogenous variables (hormone levels, blood sugar levels, etc.) determine in part which activity is expressed. Implicit in this model is the notion that at every level in the hierarchy, a '"decision" is being made among several alternatives of which one is chosen. At the top the decisions are very general (i.e. feed versus drink) and become increasingly more specific as one moves down a hierarchy (i.e. pounce or freeze).

A number of computational arguments have been advanced against this type of hierarchy [Maes90, Tyrrell93,93a]. As an alternative to either a totally flat distributed structure or a strict winner-take-all hierarchy, Tyrrell proposes a "free-flow hierarchy", in which all nodes in the hierarchy can influence the subsequent behavior of the animat. In this latter model, first proposed by Rosenblatt and

Payton[Rosenblatt89], activities express weighted preferences for activities lower in the hierarchy and ultimately motor commands. Arbitration is ultimately done at the motor controller level when it executes the most highly preferred motor action. Tyrrell argues that free flow hierarchies avoid at least two problems associated with winner-take all hierarchies:

- In winner-take-all hierarchies information is lost at each decision point. The system is unable to arrive at compromise solutions which ultimately might be better for the creature since the preferences of losing branches are not taken into account. This problem is reduced in free-flow hierarchies since everyone gets to express their preference.

- Winner-take-all hierarchies can be structured in such a way that upper level nodes have access to all of the sensory information used by nodes beneath them in the hierarchy in order to make the right choice among alternative branches. This results in a "sensory bottleneck".

We agree with the former point, but are less convinced by the second in the case of animats driven in part by internal needs. First, following Tinbergen [Tinbergen50] and others, initial decisions among activity systems tend to be driven by internal needs with just enough sensory input to take advantage of opportunities. The sensory input relevant to a specific sub-activity within a given branch of the hierarchy is often irrelevant to the higher level decision between systems. Second, information is flowing into the hierarchy at all levels via releasing mechanisms and endogenous variables associated with different nodes in the hierarchy. Third, a winner-take-all hierarchy provides a focus of attention allowing it to avoid processing irrelevant sensory data. For example, if an animal's hormone levels are such that it has no interest in sex, there is no need to check the sensory input which is only relevant to that behavior system.

The ability of free-flow hierarchies to arrive at compromise solutions comes at the expense of complexity. In particular, the mechanism for combining preferences need to be chosen very carefully [Tyrrell93], and the performance of the system is highly dependent on careful tuning of weights. This problem is compounded by the fact that since an activity can only express its preference for a given motor command, the weight it uses to represent its preference can not be determined independent of the weights used by all other activities for that particular motor action as well as alternative motor actions. Indeed, with a pure free-flow network, one loses the attractive "divide-and-conquer" aspect of hierarchical systems. This in turn brings into question how easily such a system would scale. In addition, free-flow hierarchies do not provide the "focus of attention" described earlier, and thus irrelevant sensory data may be evaluated.

We have chosen to implement a winner-take-all hierarchical model in keeping with traditional ethological models but provide a simple mechanism for limited information sharing. In this model, there is one activity which ultimately has final say over what set of motor actions should be performed at a given instant. However, losing activities on the path to the winner may "post" recommendations for and against various motor actions. The winner has access to these recommendations, and can use them as a form of taxis, or potential modification of an underlying pattern of behavior. The key point is that the winning activity has control over how it wishes to make use of the recommendations. For example, an anti-predator system which detects the presence of a distant predator (so distant in fact, that the system is not made active) may post recommendations against movement which would bring the animat closer to the predator.

This approach reflects the ethological belief that at any one time there is one activity which is in control, presumably because it is the most important given the internal and external state of the creature. Thus, it is in the best position to decide how and when to modify its default pattern of actions so as to take the preferences of other activities into account. We can use this approach because we model activities as objects with hopefully simple, but potentially complex internal logic which "decides" what motor commands will be executed. This design provides the animat designer with the freedom to decide how a given activity makes use of recommendations from other activities.

2.3 Using a Common Currency for Endogenous Variables and Releasing Mechanisms

Internal needs and external opportunities need to be evaluated using a common currency. This idea, described by Lorenz [Lorenz73], Baerends [Baerends55] and McFarland [McFarland75,76,93] is simply that an animal's response to external stimulus depends both on the strength of the stimulus and on their internal state. This seems to imply 2 key points. First, a stimulus (or more precisely, the output of the releasing mechanisms) needs to be measured as a continuous quantity as opposed to a boolean, otherwise one would not see this phenomenon. Second, some mechanism within the behavior systems of the animal is effectively combining the strengths of the relevant external and internal factors in some way. To model this properly, internal needs and external opportunities need to be expressed using a common currency.

While it is generally agreed that use of a common currency makes sense, it is less clear how the output of multiple releasing mechanisms should be combined and how external and internal factors should be combined. With respect to multiple independent external stimuli, Seitz's law of heterogeneous summation (i.e. a simple additive relationship) [Lorenz 73]may suffice for most cases. However, one simple rule may not be adequate to combine external and internal factors. McFarland argues that " ...we would expect the decision criteria (shape of the isocline) for feeding to be shaped by natural selection in accordance with the animal's ecological circumstances" [McFarland76]. That is, the way that internal and external factors are combined in a given behavior system is determined by natural selection and likely to vary depending on the specific behavior system and the animal's ecological niche. As Tyrrell [Tyrrell93] points out, it is important to be able to "accommodate different rules for combination of stimuli (i.e. internal and external factors), and one should not presuppose strict summation or multiplication".

It is also important to have control over the relative ranges over which the output of specific releasing mechanisms are allowed to vary. In general, the greater the range relative to the likely range for internal factors, the greater the reactiveness of the animat with respect to the stimulus signaled by that releasing mechanism. This results in greater persistence in the presence of a given stimulus. It also means an increased likelihood of opportunistic behavior during appetitive activities associated with one behavior system when stimuli associated with another behavior system are detected.

The computational model presented in the next section reflects all these considerations.

3. The Computational Model

The computational model presented below preserves the loose hierarchical structure which is implicit in the models of Tinbergen [Tinbergen50] or Baerends [Baerends76], but incorporates a modified form of information sharing via the use of recommendations. Ludlow's model of mutual inhibition and fatigue is embedded in the hierarchical model, and is used to implement the winner-take-all arbitration among activities at a given level, as well as to provide added control over temporal aspects of behavior. Relevant external and internal factors are modeled using abstractions called releasing mechanisms and endogenous variables respectively. While they may perform arbitrarily complex calculations to arrive at their value, their value is expressed in a common currency.

The essential points of the model are:

- Activities are organized in loose overlapping hierarchies with more general activities at the top and more specific activities at the leaves. Activities correspond to nodes in a tree, and a node can have 0 or more children. Action-selection is the process of determining which leaf node should be active at a given instant, starting at the root of the tree and descending downward.

- Children (i.e. all of the children activities associated with a given node) are mutually inhibiting, and only one can be active at a time. If the active activity is a leaf node, it may issue motor commands to the animat, otherwise, its children compete for control, and so on until a leaf node is reached.

- Activities compete on the basis of their value. Their value at time t is calculated using a modified form of Ludlow's model. Note while a specific function is specified for combining the output of relevant releasing mechanisms and endogenous variables, this function is intended to be activity-specific, and thus subject to modification as needed for a given activity:

$$V_{it} = Max\left[\left(1 - f_{it}\right) * \left[Comb\Big(\sum_k r_{kt}, \sum_l e_{lt}\Big)\right] - \sum_j (I_{ji} * V_{jt}), 0\right]$$

where:

V_{it} = value of activity i at time t.

f_{it} = level of fatigue of activity i at time t (see below).

$Comb(r,e) \rightarrow$ if $(e < 0)$ return e else return $e + r$.

r_{kt} = value of releasing mechanism k at time t where k ranges over the releasing mechanisms relevant to activity i.

e_{lt} = value of endogenous factor l at time t. l ranges over endogenous factors relevant to activity i.

- Within a collection of mutually-inhibiting activities, the system iterates until a stable solution is found in which one activity has a positive value and the value of remaining activities are within a given tolerance of zero.

- Activity specific fatigue is modeled as follows:

$$f_{it} = clamp((1 + fw_i) * f_{i(t-1)} + (V_{i(t-1)} * kf_i) - fa_i, 0, 1)$$

where:

f_{it} = level of fatigue for activity i at time t

f_{wi} = value - dependent rate of increase in fatigue for activity i.

kf_i = fatigue gain for activity i.

$V_{i(t-1)}$ = value of activity i at time t - 1.

f_{ai} = autonomous decrease in fatigue for activity i.

$clamp(a, \min, \max) \rightarrow$ clamp a to between min & max.

- Significant events and objects in the world relevant to a given activity are identified from sensory input by Releasing Mechanisms. The output of a given Releasing Mechanism may be the result of an arbitrarily complex calculation based on sensory input, but its value is expressed as a continuous variable clamped to a specific range. That is:

$$r_{kt} = clamp(f_k(r_{k(t-1)}, s_{kt}), \min_k, \max_k)$$

where:

r_{kt} = value of releasing mechanism k at time t

s_{kt} = sensory input at time t relevant to releasing mechanism k

$f_k(r_{k(t-1)}, s_{kt})$ = arbitrarily complex function of current sensor input and optionally, previous values of releasing mechanism k.

- Similarly, internal state (hormonal levels, food and water levels etc.) relevant to a given activity are modeled via Endogenous Variables. Since Endogenous Variables are abstractions for the end-results of potentially complicated internal systems, the value of an Endogenous Variable may be the result of an arbitrarily complex calculation. In the default case however, its value may be calculated as follows:

$$e_{it} = e_{i(t-1)} + ea_i - \sum_h (V_{h(t-1)} * ke_h) + fi()$$

where

e_{it} = value of endogenous variable i at time t.

ea_i = autonomous change in e_i

h = ranges over activities which affect e_i.

$V_{h(t-1)}$ = value of activity h at t - 1.

ke_h = endogenous gain associated with activity h.

$f_i()$ = arbitrary function of other factors

• An activity can depend on any number of endogenous variables and releasing mechanisms and these in turn can be shared by any number of behaviors.

• Losing activities on the path to the active activity may nonetheless post 1 or more recommendations. For example, in the diagram below all of the shaded nodes may post recommendations which may be used by the winning activity. A recommendation includes: the name of a motor command and a strength, where positive strength indicates a positive recommendation and a negative strength indicates a recommendation against.

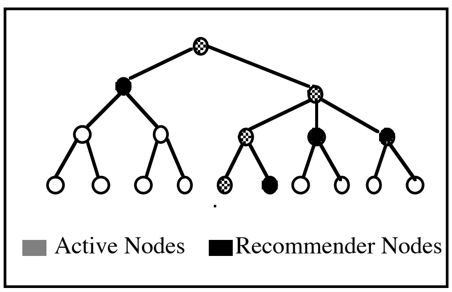

Figure 1. Behavior Hierarchy with Recommender Nodes

An activity at a leaf may issue one or more motor commands to the animat. What motor commands it issues depends on its function, sensory input and state, as well as on its evaluation of the recommendations which have been posted.

The model described above has been implemented as part of Hamsterdam, an object-oriented tool kit for modeling artificial animals The next section provides a quick description of Hamsterdam.

4. Hamsterdam and Alive

Hamsterdam is an object-oriented toolkit for modeling artificial animals in a 3D simulated environment. The classes of Hamsterdam provide the generic core functionality from which a designer builds the creatures, the world in which they live, and the instruments used to gain insight into the inner state of the creatures. The creatures are modeled as autonomous creatures with a set of internal needs, a repertoire of activities which they can perform, a sensory system which allows them to sense their world, a motor system which allows them to move in the world and a behavior system which is an object-oriented implementation of the algorithm described above. Creatures do not have a world model, nor indeed any information about the world except that which is available via their sensors. The creatures "live" in a 3D continuous world (i.e. not a grid world) populated by other creatures as well as food, water and obstacles such as walls. The system is implemented in C++ on an SGI Indigo Elan and operates in real time. Figure 2. shows the system along with a number of its gauges.

Hamsterdam has been used to model a world which includes Hamsters and Predators. The Hamsters are very loosely modeled after real Hamsters, whereas the Predators are "generic". The hamster's repertoire of top-level activities (or behavior systems) includes: Foraging (finding food and carrying it back to their hoard), Feeding (finding food and eating it), Drinking (finding water and drinking it), Cleaning, Avoiding and Fleeing from Predators, Sleeping (finding its nest and sleeping), and Death (which occurs if eaten by the predator or if certain internal state variables exceed a given range of acceptability). Activities are organized in a loose hierarchy. For example, Feeding is a top-level activity with 3 children: Chewing, Preparing-to-Eat, and Searching-for-Food. Searching-for-Food in turn has 3 children: Wander, Avoid-Obstacles, Move-to-Food. Altogether, the activity hierarchy of the Hamster includes: 55 nodes or activities, 57 releasing mechanisms, 8 endogenous variables, and 25 motor commands. The predator's repertoire is similar. The creatures sense the world by means of a sensor which shoots rays of a prescribed range and records what it finds along those rays. The creatures are physically modeled and move via motors controlled by a motor controller which accepts approximately 25 commands. A variety of gauges provide insight into why the Hamster or Predator is behaving as it does. An earlier version of Hamsterdam was used to demonstrate a distributed model of cooperative hunting, in which several predators effectively cooperated using a few simple distributed rules to surround and kill the Hamster. Hamsterdam also formed the basis for the Alive project discussed below.

The goal of the Alive installation at Siggraph 93, was to present a virtual world in which a user could interact, in natural and believable ways with autonomous semi-intelligent creatures The world and the creatures populating the world were built using Hamsterdam. The creatures were slightly modified versions of the Hamster (the foraging behavior was modified to view the user as a potential source of food) and Predator (the user was considered a predator of the Predator) described above. The user was represented in the world via a virtual creature whose actions were based on what the real user was doing as sensed by a vision system. Thus, from the standpoint of the Hamster and Predator, the user was just another creature. Over 500 people interacted with the Alive system over the 5 days of the conference.

Figure 2. Screen dump of Hamsterdam showing the world and some of the gauges

5. Results

This section uses results from Hamsterdam and Alive to demonstrate the importance of some of the ideas incorporated in the computational model.

In the cases presented below, the Hamster is in an enclosure containing food and water. To eat or drink, it must explore the enclosure until it senses the food or water, move to the appropriate resource, position its head accordingly, and then chew or drink. Thus, the various activities have the equivalent of appetitive and consumatory phases.

Figure 3. Low Inhibitory Gains

Figure 4 High Inhibitory Gains

Figures 3 and 4 demonstrate how inhibitory gains may be used to control persistence. They present state-space diagrams [after McFarland] for hunger and thirst levels under 2 different cases of inhibitory gains (low and high). The straight diagonal lines represent switching lines based on the level of inhibitory gains. The starting point for the systems is marked, and the origin represents the point of satiation for both thirst and hunger. As one can see, when the gains are low, the system tends to dither between feeding and drinking with the result that the system takes longer to reach satiation levels. At higher gains, the active activity shows more persistence with the effect that satiation is reached sooner.

Figure 5. Demonstration of Time-Sharing

Figure 5 demonstrates the use of fatigue to provide a form of time-sharing by showing the pattern of activities over time when activity-specific fatigue is included. The various blocks correspond to when a given activity is active. Feeding is represented by the black blocks, drinking by the dark gray blocks and cleaning by the lighter blocks on top. The initial level of hunger is twice that of thirst, and the need to clean is half of the level of thirst, and there is neither food nor water to be had. Without fatigue drinking and cleaning would never become active. When fatigue is introduced the system alternates between feeding and drinking with an occasional interruption for cleaning, even though the internal value (before fatigue) of the various activities stay unchanged. This is an important demonstration because it shows how the system can avoid the mindless pursuit of an unattainable goal to the detriment of other goals.

Figures 6 and 7 demonstrate how adjustments to the range associated with a given Releasing Mechanism can result in opportunistic behavior. The figures show the levels of hunger and thirst over time. In both cases, the Hamster starts at the same location and in the process of searching for food passes near water. In figure 6, with a lower allowed maximum for the water releasing mechanism, the Hamster ignores the water until after it has eaten. When a higher value is used (figure 7), the Hamster interrupts its search for food to take advantage of the water. This can be seen by comparing the respective traces of endogenous needs and noting that the level of thirst drops sooner in figure 7 than in 6.

Figure 6. Low Range for Releasing Mechanism

Figure 7. High Range for Releasing Mechanism

The value of recommenders can be seen in the following experiment in which we turned off recommenders. When the move-to activity is modified so that it ignores recommendations made by the avoid activity, the Hamster quickly becomes stuck oscillating between the two activities. In another experiment we turned off the anti-predator activity's ability to make recommendations when it was not active. The Hamster was attacked and killed within 1600 time steps. By contrast, in the normal case the Hamster manages to survive for 6400 time steps.

Performance of the algorithm has not been an issue to date since it takes less than 10ms to perform action-selection in the full behavior hierarchy of the Hamster. Note, this does not include the time taken for sensors to update their state, although it does include the time taken by the releasing mechanisms to evaluate the sensor data.

The experience with the Alive project demonstrated that the approach described in this paper was capable of generating generally believable and robust behavior in a highly dynamic and uncertain environment and with many of the problems of real-world "sensing". People generally found the creatures' patterns of behavior and responses believable. It should be noted that the actual modifications to the Hamster's behavior code to make it work in the context of the Alive project were extremely minimal.

6. Issues and areas for future work

The activity hierarchies are currently built by hand, and parameters such as inhibitory gains and rates of fatigue tuned in response to the observed behavior. As Maes points out [Maes93], this raises questions of how well the approach will scale. It should be noted that in practice parts of the hierarchy can be shared or used in multiple instances with only minor modifications. Nonetheless, more and better tools are needed, particularly to aid in finding the right values for parameters so as to achieve a given temporal pattern. Perhaps a genetic algorithm approach could prove useful, and in fact, Hamsterdam incorporates the idea of a genome which includes many of the parameters associated with the activity networks. However, no experiments have been performed, and Tyrrell had limited success using a similar approach[Tyrrell93].

Learning needs to be added and it remains to be seen how easily that can be done. However, it is believed that an ethological approach also provides a useful perspective from which to approach certain types of learning. For example, it may be possible to model "Pavlovian Conditioning" as the modification of existing releasing mechanisms or the creation of new ones, and operant conditioning as the copying (or associating) the motor patterns from one activity system into (or with) the repertoire of appetitive motor patterns of another activity system. [Davey89, Timberlake89, Lorenz73].

Even though the behavioral complexity is more complex than many of the animats described in the literature, the behavioral complexity modeled to date is still relatively simple, and it is an open question as to how well the approach will scale. We intend to investigate this issue by attempting to model a more complex creature, for example, on the order of a dog.

7. Conclusion

A computational model of action-selection has been presented, which by drawing on ideas from Ethology, addresses a number of problems which have been noted in computational models of action-selection proposed to date including the need for greater control over the temporal aspects of behavior, the need for a loose hierarchical structure with information sharing, and the need for a flexible means of modeling the influence of internal and external factors. The larger message of this paper is that in order for an animat's behavior to be natural and animal-like, we believe that incorporating some of the mechanisms proposed by Ethologists will be necessary.

Acknowledgements

The author wishes to acknowledge and thank Professor Pattie Maes for her many contributions to my research and this paper. Her intuition, insight, and wealth of knowledge coupled with generous amounts of patience and encouragement has been invaluable.

References

Baerends, G.,Brouwer, R. & Waterbolk H., 1955, Ethological studies on Lebistes Reticulatus,I: An analysis of the male courtship pattern. Behavior 8.

Baerends, G., 1976, On drive, conflict and instinct, and the functional organization of behavior. In: Perspectives in Brain Research 45, Corner M. & Swaab, D. eds.

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

Davey G., 1989, Ecological Learning Theory, Routledge Inc., London.

Dawkins, R. 1976, Hierarchical Organization: A Candidate principle for ethology. In: Growing Points in Ethology, Bateson P. & Hinde R. ed. Cambridge University Press.

Gallistel C., 1980, The Organization of Action: A New Synthesis, Lawrence Erlbaum , Hillsdale NJ.

Lorenz, K., 1973, Foundations of Ethology, Springer-Verlag. New York.

Ludlow, A.,1976, The Behavior of a Model Animal, Behavior Vol. 58.

Ludlow, A.,1980, The Evolution and Simulation of a Decision Maker, in Analysis of Motivational Processes, ed. Toates, F. & Halliday, T. , Academic Press, London.

Maes P., 1990, Situated Agents Can Have Goals, In: In: Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back, Edited by P. Maes, MIT Press, Cambridge.

Maes, P., 1993, Modeling Adaptive Autonomous Agents, to appear in The Artificial Life Journal, Vol. 1, Numbers 1&2, Summer 1994.

McCleery, R., 1983, Interactions between Activities. In: Animal Behavior: Causes and Effects, Vol. 1, Halliday, T. & Slater P. ed., W.H. Freeman & Co., New York.

McFarland, D., 1974, Time-Sharing as a Behavioral Phenomenon, in Advances in Animal Behavior, Vol. 5., Academic Press, New York.

McFarland, D., 1976, Form and Function in the temporal organization of behavior. In: Gowing Points in Ethology, Bateson P. & Hinde R. ed. Cambridge University Press.

McFarland, D., 1993, Animal Behavior, Longman Scientific and Technical, Harlow.

McFarland, D. & Sibley, R. , 1975, The behavioral final common path, Philosophical Transactions of the Royal Society, B. 270.

Payton D., Keirsey D., Krozel J., and Rosenblatt K., 1991, Do Whatever Works: A Robust Approach to Fault-Tolerant Autonomous Control, In: Journal of Applied Intelligence, Vol. 3. 1992.

Rosenblatt K., & Payton D., 1989, A Fine-Grained Alternative to the Subsumption Architecture for Mobile Robot Control, In: Proceedings of the International Joint Conference on Neural Networks, June 1989, Washington DC.

Timberlake, W. and Lucas, G. , 1989, Behavior Systems and Learning: From Misbehavior to General Principles, in Contemporary Learning Theories: Instrumental Conditioning Theory and the Impact of Biological Constraints, ed. Klein S. & Mowrer, R. Lawrence Erlbaum Associates, Inc. Hillsdale NJ.

Tinbergen, N., 1950, The Study of Instinct. Clarendon Press, Oxford.

Toates, F., 1983, Models of Motivation. In: Animal Behavior: Causes and Effects, Vol. 1, Halliday, T. & Slater P. ed., W.H. Freeman & Co., New York.

Toates F. & Jensen P., 1991, Ethological and Psychological Models of Motivation: Towards a Synthesis, In: From Animals To Animats, Proceedings of the First International Conference on the Simulation of Adaptive Behavior, Edited by Meyer, J. and Wilson, S.W., MIT Press, 1991.

Tyrrell T., 1993, Computational Mechanisms for Action Selection, PhD. Thesis, Centre for Cognitive Science, University of Edinburgh.

Tyrrell T., 1993a, The Use of Hierarchies for Action Selection, In: From Animals To Animats, Proceedings of the Second International Conference on the Simulation of Adaptive Behavior, Edited by Meyer, J. and Wilson, S.W., MIT Press, 1993.