Ecological Modelling, 44 (1988) 125–141
Elsevier Science Publishers B.V., Amsterdam – Printed in The Netherlands

AN ARTIFICIAL INTELLIGENCE MODELLING APPROACH TO SIMULATING ANIMAL/HABITAT INTERACTIONS

H. SAARENMAA [1], N.D. STONE [2,4], L.J. FOLSE [3], J.M. PACKARD [3], W.E. GRANT [3], M.E. MAKELA [2] and R.N. COULSON [2]

[1] *The Finnish Forest Research Institute, Unioninkatu 40 A, 00170 Helsinki (Finland)*
Departments of [2] *Entomology, and* [3] *Wildlife and Fisheries Science, Texas A&M University, College Station, TX 77843 (U.S.A)*

(Accepted 7 June 1988)

ABSTRACT

Saarenmaa, H., Stone, N.D., Folse, L.J., Packard, J.M., Grant, W.E., Makela, M.E. and Coulson, R.N., 1988. An artificial intelligence modelling approach to simulating animal/habitat interactions. *Ecol. Modelling*, 44: 125–141.

Ecological modellers have begun to recognize the potential of object-oriented programming techniques in structuring models. However, little has been done to take advantage of artificial intelligence's (AI) symbolic representations to model the decision-making processes of animals. Here, a generic model of animal–habitat interaction and a specific model of moose–, *Alces alces* L., forest interactions in Finland are described that are event-driven and behavior-based. Individual level simulation is accomplished through an object-oriented knowledge representation scheme and AI techniques to implement a hierarchical decision-making model of behavior. The habitat is likewise represented in an object-oriented scheme, allowing the simulation of a heterogeneous environment. Other AI techniques for modelling behavior, memory, and actions are discussed including LISP methods, rule-based reasoning, and several search algorithms. Simulations of the moose–forest system show the power of this approach but are not intended to advance the theory of large-herbivore behavior and foraging. AI techniques are found to be most beneficial in (a) studying population processes based on individual level models of behavior, (b) modelling spatial heterogeneity, (c) building event-driven models, (d) providing a conceptual clarity to model construction, and (e) providing a structure equally well suited to simulating resource management.

INTRODUCTION

Behavioral ecology is a field that almost certainly will benefit from AI research in the near future (Bobrow and Hayes, 1985). Aaron Sloman

---

[4] Present address: Department of Entomology, Virginia Polytechnic Institute and State University, Blacksburg, VA 24060, U.S.A.

Technical Article 22923 of the Texas Agricultural Experiment Station.

0304-3800/88/$03.50

(quoted in Bobrow and Hayes, 1985) suggests that AI methodologies will contribute most directly to ecological disciplines by providing precise techniques to model and represent the decision-making mechanisms found in lower animals. While ecological modellers have recognized the potential of AI programming techniques (Coulson et al., 1987), little has been done to take advantage of cognitive modelling capabilities. Much work is still preliminary. Loehle (1987) suggests that expert systems and AI techniques may benefit modellers by providing new intelligent simulation tools to analyze and develop traditional models of continuous systems, and Coulson et al. (1987) described the process of applying AI to understand, manage and model natural resource systems.

Very few models of natural systems have been developed using AI techniques. One technique described below, object-oriented programming, has previously been advocated for biological simulation. Stone (1987) addressed the application of AI techniques to the process of systems analysis and modelling, and included an an object-oriented simulation model of insect host–parasitoid population dynamics, and Graham (1986) used an object-oriented programming approach in her model of feral horses on the island of Shackleford Banks, North Carolina. One of the main advantages to the object-oriented approach cited by both authors was the ability to simulate a heterogeneous environment. In those models, however, animal behavior was modelled in traditional ways despite the structure of the models. Here we demonstrate the utility of AI's symbolic representations and programming approaches to model a general class of ecological problem – animal–habitat interactions – based on explicit behavioral models of individuals in spatially heterogeneous environments.

Modelling approaches

Population processes are the sum of individuals' actions. When populations are large enough and the environment is relatively homogeneous, population level equations are adequate to describe the population's dynamics with acceptable variance. This is the case in many insect pest models in agriculture, for example. However, when populations are small or patchy, or individual behavior is sensitive to local events in a spatially heterogeneous environment, population level models are unlikely to provide adequate resolution and sensitivity. The need for incorporating behavioral processes is also obvious when modelling intraspecific mechanisms of population regulation in socially structured populations (De Jong and Saarenmaa, 1985).

Building models of individual behavior, however, is complicated by the fact that behaviors can be highly variable and difficult to describe quantitatively. Still, simulation of population processes at the individual level has been used successfully to explain, for example, the dispersion of pest insects in crop fields (Jones, 1977) and the foraging behavior of mites (Sabelis, 1981) and parasitic wasps (Waage, 1979). However, such models have been restricted to component processes of natural systems, and have not been incorporated into population models.

In pursuing the goal of making computers behave intelligently, AI researchers have developed several symbolic and nonalgorithmic techniques that mimic human learning, goal-oriented problem solving, environmental perception, decision-making, etc. These techniques aimed at reproducing human intelligence may also be adapted to model similar processes in animals to address applications involving spatial heterogeneity of envrionments, goal-directed behavior, interactions among motivational systems, memory of spatial relationships (cognitive maps), and integration of individual decision rules with population or ecosystem processes. Each of these aspects has been addressed using traditional modelling approaches such as optimality theory or cybernetic systems (Owen-Smith and Novellie, 1982; Staddon, 1983; Toates, 1986). However, the use of analytical models or linear programming techniques to represent parallel processes and hierarchically organized information is cumbersome.

The problem of European moose, *Alces alces* L., as pests of Finnish forests, described below, provides a test case for the application of AI techniques to ecological problems. The management of a moose-forest system is one of a class of problems involving large herbivores (Owen-Smith and Novellie, 1982) that have eluded analysis by traditional approaches like optimal foraging theory (Stephens and Krebs, 1986). The model presented is intended to illustrate novel simulation techniques rather than to advance the understanding of systems dominated by large herbivores. More work is needed before a realistic and generic model of such systems can be developed.

Moose–Forest problem

Moose is the key pest of pine and hardwood plantations in Scandinavia today. Winter feeding causes losses of $10 million annually in Finland alone. Repo and Löyttyniemi (1985) have shown that moose damage is a function of landscape characteristics such as proximity to human structures and the characteristics of neighboring stands in a forest. This suggests that moose foraging behavior depends on the availability of visual cover and resting sites. Moose feed at night on a variety of tree species including aspen, sorb, willow, pine, birch, and spruce, but show no consistent preference (Sainio, 1956; Ahlén, 1971; Andersson, 1971; Danell and Ericson, 1986). Instead, their diet varies according to their nutritional needs. Pine species are high in protein and energy, while hardwood species contain necessary minerals. Moose may choose plants on the basis of mineral content even though species higher in energy and protein are available (Belovsky, 1978).

Ultimately, individual moose needs, preferences, and experience relative to the plantation structure of its forest environment determine the extent of damage caused. We assume moose in Finnish plantations have two basic behavioral modes: searching for food during the night, and searching for shelter before daybreak. An individual moose's searching behavior is dependent on its internal state and experience. Each moose requires different types and amounts of food depending on its age, nutritional status, and thermoregulatory requirements. Depending on its needs and memory, each moose selects the forest plantation that best meets its demands.

Moose habitat consists of a mosaic of internally uniform stands (here called compartments), each with a characteristic tree density, age or height, area, and species composition. Some plantations are inaccessible due to physical barriers or undesirable due to predators and/or human habitation. Thus, moose develop patterns of movement among compartments, effectively creating a set of paths through the forest.

The forest manager must choose strategies for cutting and reforestation of compartments that reduce moose damage to acceptable levels. Developing an appropriate strategy depends, therefore, on a thorough undertanding of moose behavior in relation to forest structure, and the ability to make quantitative or qualitative comparisons between alternative management scenarios.

Two approaches previously used in modelling moose foraging are inappropriate for use as decision aids on individual plantations. Optimal foraging models have been used to examine whether moose maximize energy intake or minimize the time spent satisfying mineral needs (Belovsky, 1978), but such models do not account for the spatial aspects of the problem. Models based on statistical summaries of moose behavior within experimental environments have likewise had low predictive power for individual plantations (Repo and Löyttyniemi, 1985; Näslund, 1986).

Here, individual moose were modelled using AI techniques to allow each moose to act as an autonomous entity in a spatially heterogeneous environment of forest compartments. Each moose in the simulation was based on the same generic description of moose behavior. Only differences in accumulated experience of individuals differentiated the moose. Because individual behavior drove the model, population level phenomena were expressed as the sum of individual actions, not as the solution to population equations. This model structure is also well suited to the simulation of a forest manager's control strategies as the behavior of an autonomous individual.

Knowledge representation and search

The two areas of AI research most relevant to modelling animal behavior and animal–habitat interaction are *knowledge representation*, how information is stored in the computer; and *search*, how to select and evaluate paths from a starting point to a goal. Knowledge representation schemes and search strategies capture and make explicit the relationships among and behaviors of the key elements in the system.

The fundamental lesson of AI is that choosing a good description of the problem (i.e., the representation of the information known about a problem) can make the problem's solution trivial (Winston, 1984). For the current task, representations of moose and the forest are needed. As discussed below, a representation scheme based on hierarchical *object classes* (dynamic data structures) was used for both moose and the forest; this style of programming is called *object-oriented* (Bobrow and Stefik, 1986). Similarly, there are numerous search strategies within AI available for evaluating possible paths to achieve specific goals. The most fundamental are *depth-first*, in which only one path is explored at a time; and *breadth-first*, in which all paths are considered in parallel. Selection of a search strategy is dependent on the knowledge that is available. In the case of the moose, knowledge arises from investigative behavior, is stored in associative memory, and is used to manifest a feasible (although not necessarily optimal) foraging strategy. AI search methods that approximate this are *best-first, hill-climbing*, and *A** (Winston, 1984).

Object-oriented simulation

One of the AI techniques that has been recognized as being suitable for building simulation models is object-oriented programming (Pascoe, 1986; Stefik and Bobrow, 1986). Object-oriented programming uses modules called objects to describe elements in the knowledge domain, and message passing between objects to describe interaction between elements. The objects themselves contain data describing the state of the object and procedures that allow the objects to take action. These procedures are called *methods* and are invoked when an object receives a message. Thus, messages are like procedure calls to objects, except that the interpretation of the message is carried out by each object and is not specified in the message itself. This allows two objects to respond differently to the same message. In addition, many object-oriented languages allow for a hierarchy of object classes in which subclasses inherit the attributes and procedures of their parent class

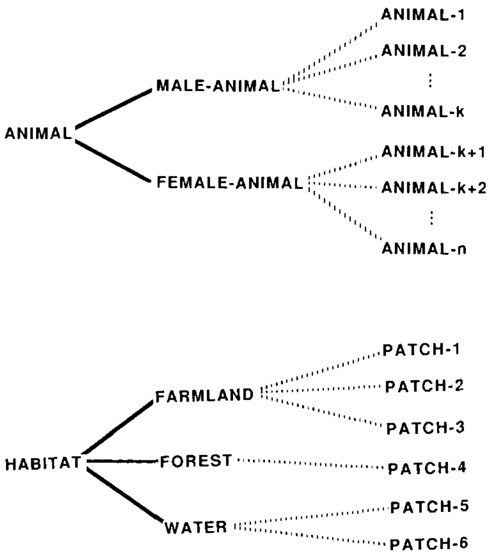

Fig. 1. Object-class structure for a generic model of animal–habitat interaction. Animal objects are instances either of male or female animal classes, both of which are subclasses of the animal class. Patch objects are instances of one of three classes of habitat. Solid lines represent class membership, dotted lines connect objects to their parent class.

as a default. This provides another level of description since the structure of the hierarchy itself can be used to depict class/subclass relationships in the natural system.

These features of object-oriented programming provide a conceptual structure to the knowledge domain that can enhance the clarity of animal/habitat models designed using that structure. Because flow of control logic is contained within the objects, objects themselves appear to act and react in the modelled system. This format provides a clear link to observations of the natural world in which individuals and objects interact and affect the current states (attributes) of one another. Furthermore, object-oriented programming is well suited to simulating animal behavior because there is a clear analogy between signal perception in natural systems and message passing between objects in the model.

A generic model of hierarchical decision making using object-oriented techniques serves to illustrate the power of the approach. Figure 1 shows the simple class-object hierarchy for an unspecified animal species and its habitat. The environment is made up of objects that represent discrete patches in a landscape, linked together in a mosaic or network. Each patch is an instance of a class of land type, including water, agricultural fields, and forest. Males and females are considered distinct enough in their attributes

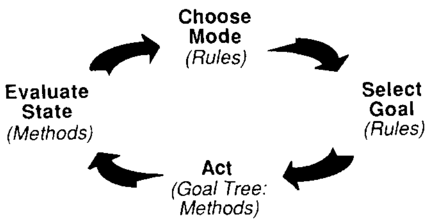

Fig. 2. Hierarchical decision process carried out by each animal object in each iteration of the generic behavior model. Each cycle begins with an evaluation of current state and ends when actions are carried out, both implemented through methods. Choice of behavioral mode and action goal are made through evaluating a rule base.

and behaviors to be represented by different classes. Attributes and procedures that are shared by all animals are defined in the ANIMAL class. These might include attributes like age or position in the environment, and methods like moving, sleeping, and eating. Sex-specific attributes and behaviors would be defined in the subclasses, MALE-ANIMAL and FEMALE-ANIMAL. The methods for social behaviors would be different for the two sexes, but the names of the methods could be the same. Thus a male animal object (an instance of the MALE-ANIMAL class) could respond to the message 'dominant-male-approaching' differently from a female animal object.

The behavior of the animal is modelled as a series of decisions (Fig. 2). Initially, a behavioral mode is chosen based on an evaluation of the animal's internal state and its current environmental situation. In this general model, the choice of mode is carried out by evaluating a rule base, represented as a set of rules which are instances of a hierarchy of rule classes (Fig. 3). This object-oriented treatment of rule-based reasoning is based on the KEE™ (IntelliCorp, Inc.) system's implementation in which the inference engine is invoked by a message passed to a rule class (e.g., CHOOSE-MODE-RULES). The rule base includes conflict-resolution rules as well. Next, an action goal is selected again through rule-based reasoning using the CHOOSE-GOAL-RULES (Fig. 3).

Action goals are carried out by the animal object sending itself a message to perform the action. The methods associated with each action goal in turn call specialist methods (Winston, 1984) to carry out the steps involved to achieve the goal. Figure 4 shows a goal-tree for the action goal Rest. Rest sends two messages to itself; first Seek-shelter, then Sleep. Seek-shelter either chooses the current environment, by sending itself the message Stay-put, or moves using the method Move which in turn calls Seek-shelter. Note that the implementation of the Move method involves the choice of an appropriate search strategy (Choose-best option) as discussed above. Some actions affect the local environment and involve messages to the animal

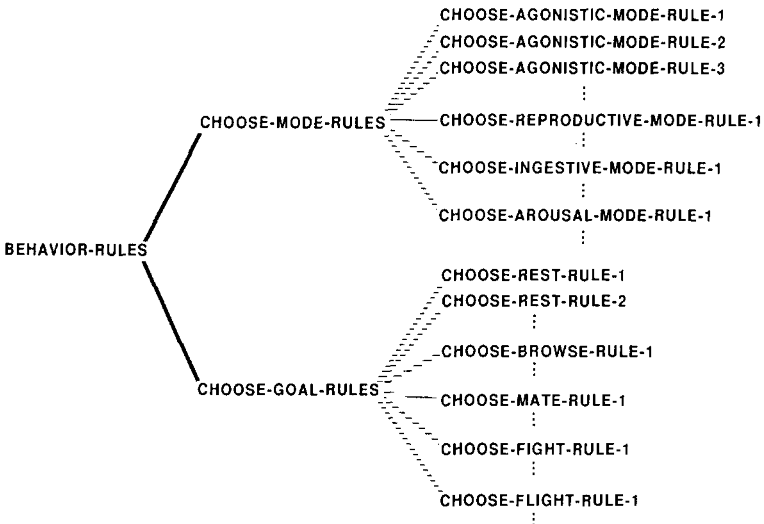

Fig. 3. Object-class rule structure for the rules used in making behavioral choices. Each rule is an object and an instance of class of rules concerning a particular decision. Inference is initiated when an object sends a message to the appropriate rule class.

object's current area object. Other actions update the internal state of the animal object so that the cycle can begin again with an evaluation of the animal's new status and position.

Conflicting or multiple goals are resolved through rule evaluation. At each evaluation point (e.g., choosing an action goal, choosing the best path) rules can be invoked that test the internal status of the animal. In this way, for example, an animal seeking shelter while thirsty could prefer a path that goes past water. In general, however, reevaluation of the animal's goal occurs following each action so that needs can be reconsidered in the context of the animal's new condition and environment.

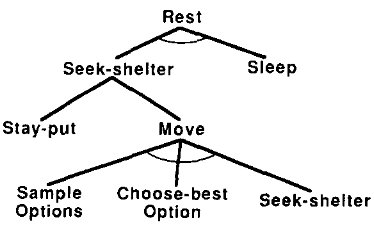

Fig. 4. A goal tree or 'and/or tree' for the method, Rest. Each name represents a method that calls the one or more methods below it and connected by lines. An arc connecting lines indicates that all intersected methods are called, otherwise only one of the connected methods need be called.

By incorporating both rules and methods for modelling animal behavior, one gains the flexibility to employ knowledge in a variety of forms to simulate behavioral decisions and actions. The general model of behavior described above simulates a hierarchial decision process (Dawkins, 1976) in which the final action depends on a chain of previous decisions (Fig. 2), but other types of reasoning can be simulated as well. For example, animals often behave as if they are pursuing a specific goal by generating a plan of action. That plan is followed until progress is impeded, whereupon another plan is adopted or the goal is abandoned. This scenario involving *action* and *stop-action* rules is identical to the inference procedure in backward-chaining expert systems, and rule-based reasoning is well suited for modelling it. Other models of behavioral choice like random selection, as in Graham's (1986) HAREMS model, or selection by evaluation functions could be effectively modelled using algorithms contained in methods.

In the following description of the moose–forest model, object-oriented representations for both animals and the environment are described. While animal species can easily be conceived as single object classes, appropriate habitat structure is less clear and will depend on the specific application. Still, object-oriented representations of landscapes seem to be flexible enough for a variety of problem types in ecological modelling (Graham, 1986; Loehle, 1987).

MOOSE–FOREST SIMULATION

The model of moose–forest interaction was developed on a Symbolics 3640 computer workstation by first constructing a prototype in LISP which represented attributes of objects as property lists (Winston, 1984), and second using a higher-level programming environment, KEE™, to implement the object-oriented paradigm. Individual moose were simulated as instances of the class, MOOSE, and the forest was represented by a set of adjacent compartment objects, instances of the COMPARTMENT class. Both models treated each object as an independent unit with its own internal dynamics and with specific ways of relating to other objects in the model. As each moose moved among forest stands, it could gain knowledge about those specific stands, updating its memory and allowing the model to simulate individual behaviors as dependent on individual experience.

Forest and moose objects

The search space of the moose is the forest, represented in the model as a set of compartments analogous to patches. The simulated environment comprised 47 forest compartments, each consisting of stands of trees of a

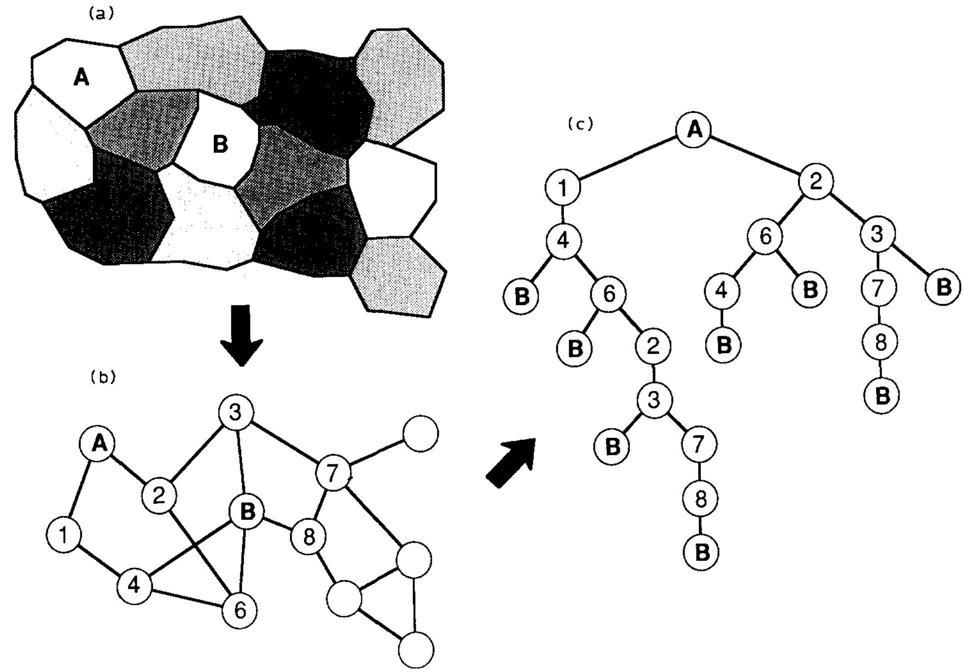

Fig. 5. Graphical, network, and tree representations of a hypothetical forest as a collection of compartment class objects. Differences in shading in the graphical representation (a) indicate different object attributes, such as stand composition. The network representation (b) is derived from the neighbor properties of the compartments. The network shows nodes (circles) for each compartment and lines representing paths between compartments. Note that not all adjacent compartments are connected by paths in this example. The tree representation (c) is derived from the network; its branches represent all possible nonrepeating paths from A to B.

TABLE 1

Ranges of random variables used to construct simulated forest

Compartment attribute	Units	Uniform distribution range	
		Upper bound	Lower bound
'height	m	33.3	0
'volume	m^3 (timber) ha^{-1}	300	0
'pine-number	–	10000	0
'hardwood-number	–	10000	0

Total tree number varied between 0 and 10000 so that within any single compartment 'pine-number + 'hardwood-number ≤ 10000. For any compartment, values were generated from a single random index so that numbers, volume, and height were correlated.

given average age (height), including a characteristic mix of hardwood and pine species. Spatial relationships among forest compartments were represented by a network (Fig. 5). Nodes represent compartments, and lines between nodes represent available paths between compartments. Search algorithms treated this network as a tree in evaluating possible paths from one compartment to another. Each compartment object was an instance of the class, COMPARTMENT, with descriptive attributes chosen to correspond to data available from typical forest maps and databases. The COMPARTMENT object and its attributes (preceded by a single quote) were:

COMPARTMENT	
'pine-number	(number of pine trees)
'hardwood-number	(number of hardwood trees)
'height	(average height (m) of all trees in stand)
'volume	(cubic meters of timber per hectare)
'area	(surface area of compartment)
'neighbor-compartments	(list of adjacent compartments)

Compartment centers were assigned randomly within a rectangular grid. Compartment boundaries, area, and neighboring compartments were then calculated based on the randomly generated compartment centers. Attributes 'pine-numer, 'hardwood-number, 'volume, and 'height for each compartment were assigned randomly based on a single random index for each compartment (Table 1). Compartments with a characteristic height of 5 m or less were considered acceptible food resources for the moose. Volume, a measure of the density of tree growth within the compartment, was used as an indicator of available shelter. The neighbor-compartments attribute is a list and may be any length, however no compartment in the simulation had more than eight neighbors. Methods associated with forest compartment objects allowed each compartment to update its attribute values after moose damage, respond to queries from moose objects about the compartment's attributes, and to display its attributes on a graphical representation of the environment. No model of forest growth was included, but one could have been added as a method to simulate growth of the compartments' trees.

The moose class included the following attributes as well as search methods for selecting among and carrying out behaviors:

MOOSE	
'location-compartment	(name of currently occupied compartment)
'memory	(list of visited compartments and attributes)
'nutritional-balance	(list of hardwood and pine requirements)
'shelter-balance	(value representing need for shelter)

Balances indicated the difference between the demands of the individual moose for shelter or food and the supply. When moose needs were not met by available nutrients or shelter, balances became negative and the moose sought to correct the imbalance. Hunger, in this scheme is measured as the absolute value of a negative nutritional balance for pine or hardwood. Moose chose between two basic behavioral modes, foraging and resting. Social interactions among moose were ignored. Switches between these two modes depended on nutritional balances and the light/dark cycle. Hunger directed the moose to seek a suitable young forest compartment in which to feed. Shelter imbalance, during daylight hours, caused moose to seek shelter in dense or mature forest stands. Conflicts occurred when there was a negative balance for both nutrition and shelter. If the nutritional imbalance was severe enough (below a negative threshold), the moose attempted to feed during the day, in a sheltered compartment if possible. These conflict resolution strategies were implemented through the rules for choosing goals.

The model implemented individual learning through the memory property. As a moose moved through compartments, the attributes of those and neighboring compartments were stored in memory as an association list. At a later time, if the moose was unable to meet its demands by moving to an adjacent compartment, it chose from memory the closest accessible (within a maximum travelling distance) compartment that contained the required attributes. Note, however, that the compartment's attributes could have changed between the moose's visit and return. In the absence of an acceptable goal in memory, the moose chose the nearest unexplored compartment. This search methodology approximated 'best-first' search (Winston, 1984).

Simulations

Three alternative models of moose search were simulated. First was a RANDOM SEARCH strategy, in which individual moose moved randomly among compartments when searching for food. Second was a strategy called *global search* based on perfect knowledge of the entire habitat. Each moose had full knowledge of the resources available in each forest stand and all possible paths between stands. The third search strategy was *local search* augmented by experiential associative memory, as described above. For each strategy, one simulated winter (100 days) was run with ten moose per simulation.

The simulation was driven by a simple day/night clock. In each phase of each day, the moose were allowed to choose a behavior and act. This involved selecting an appropriate compartment and feeding or resting. The model was therefore event-driven, rather than time-driven. This fact that actions were the result of other events and states of the system provides a very direct and conceptually clear model of the real world, since organisms tend to behave in a reactive way to events in their environment (Coulson et al., 1987). The object-oriented approach does not exclude time-driven processes from occurring. However, time becomes an internal factor as each object provides its own clock for use in methods that require rate equations.

RESULTS AND DISCUSSION

Figure 6 shows the mean number of visits in 100 days of simulation to each of the ten forest compartments (of 47) containing young trees for each of the three search models. Moose using both global and local search tended to favor the same specific compartments, though in different proportions, while moose searching randomly visited less desirable compartments regularly.

Nutritional results represent the efficiency of moose search and are summarized in Fig. 7. The vertical axis shows the natural log-transform of pine and hardwood deficits accumulated over 100 days. The random strategy

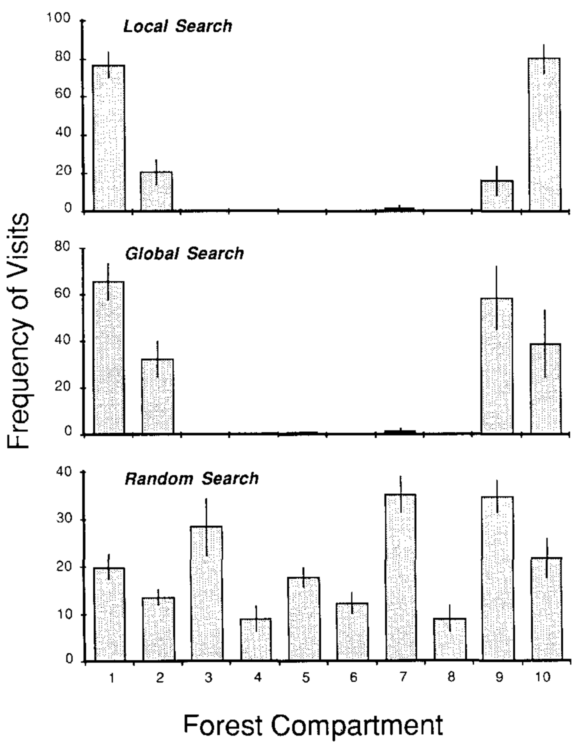

Fig. 6. Means (±1 SD) of the number of visits to ten of the 47 compartments in the simulated moose habitat over the 100-day simulation. Means are for ten moose in each of the three trials using different search algorithms.

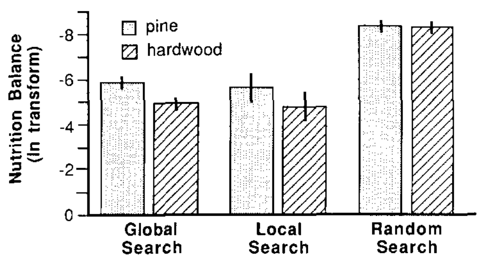

Fig. 7. Mean (± 1 SE) nutrition balance from simulations of ten individual moose feeding in a forest habitat for 100 days using each of three search strategies.

produced the largest deficit, indicating that moose searching randomly were the least successful at finding needed resources. Surprisingly, however, the global search strategy did not significantly improve the searching efficiency of the moose over the local search strategy. Moose using local search exhibited greater variation in efficiency, however. The small variance associated with the globally searching moose reflects the different starting locations of the moose in the habitat. Since each moose's behavior was based on the same information, it is not surprising that there was little variation in the net results. By contrast, each locally searching moose's experience was different, so each moose moved according to its individual needs and experience, and variation was thus increased.

Generalizing the moose–forest model

The moose–forest model illustrates how hierarchical processes within an animal can be represented as a choice between modes (feeding, resting, or travelling) followed by evaluation and search procedures specific to each mode. Internal processes controlling nutritional balance activated the animal to behave in a manner that was not simply controlled by external stimuli. Other models and descriptions of motivational systems (Staddon, 1983; Toates, 1986) and behavior could easily be included in models of this type. The level of mechanistic detail represented by the methods associated with each behavior must be determined by the purpose of the model, and the complexity of the behavioral model should be adjusted to fit the level of resolution required.

Furthermore, the model of moose and its forest environment included a simple structure for simulating the interactions between animals and their habitat as the messages passed between animal-objects and environment objects. The state of an animal population and environment was represented as the set of values for attributes of objects. The behavior of the system was governed by procedures internal to each object. This simple approach can be extended or collapsed as needed to meet specific modelling objectives and has been successfully employed in other models of animal systems (Graham, 1986; Stone, 1987).

Other programming approaches from AI not discussed here (action programming, logic programming) might also be employed. PROLOG, for example, is very well suited to problems involving nondeterministic processes as are commonly found in complex ecosystems and in evolutionary systems. Parallel processing, currently on the verge of becoming widely available, will also offer novel tools to the ecological AI modeller (Feigenbaum and McCorduck, 1984).

CONCLUSIONS

We introduced a generic and a specific model of animal behavior based on a hierarchical decision making process and rule based reasoning; however, the actual mechanism used is of secondary importance. One has to choose the mechanism that best represents a chosen conceptual model of the real world. AI techniques extend the capabilities of the modeller through the use of rules, logical evaluation functions, hierarchical class structure, and search algorithms. The emphasis in the modelling should be on the question of: Which are the testable heuristics in the search strategy; Do these heuristics produce realistic goal-oriented behaviors; and, Can the simulated strategies be proven to be optimal or satisfactory?

Modelling animal–habitat interactions using the knowledge representation structures, search strategies, and inference procedures from artificial intelligence makes building models with a clear conceptual link to the natural world easier. Constructing models of natural systems as sets of differential equations or as algorithms in a procedural computer code like FORTRAN does not lend one to think about the real objects in the system. AI programming methods, on the other hand, have many built-in ways to represent objects, their attributes, and object interactions. Modellers knowledgeable only of procedural languages would not think of building the type of behavior-based model presented here.

In their present state, the models presented above are conceptual constructions and do not attempt to broaden our understanding of the ecology of the natural system. However, the search and representation concepts of AI have been shown to work in ecological modelling and the results strongly encourage further work in AI-based simulation of animal-habitat interactions. The main advantages gained from using AI techniques are that (a) they facilitate development of truly mechanistic behavioral models with which one can study population processes based on hypotheses about individual search and decision strategies; (b) they provide a simple way to incorporate environmental heterogeneity into landscape simulation; (c) they allow the development of event-driven, rather than time-driven models; (d) they allow the construction of models with clear and direct correspondence to the natural system; and (e) they provide a good structure for assessing the effects of management strategies. AI-based models thus represent a new set of tools for testing and formulating hypotheses about animal behavior, ecology, and evolution; as such they represent a breakthrough in ecological modelling.

ACKNOWLEDGEMENTS

The authors wish to thank Douglas Loh and Lee Graham of the Knowledge Engineering Laboratory at Texas A&M for their insights and suggestions.

REFERENCES

Ahlén, I., 1971. Research on moose-vegetation interplay in Scandinavia. Proc. North Am. Moose Workshop, 5: 23–34.

Andersson, E., 1971. Observations on the winter food and diurnal rhythm of moose (*Alces alces*). Soumen Riista, 23: 105–118.

Belovsky, G.E., 1978. Diet optimization in a generalist herbivore: the moose. Theor. Popul. Biol., 14: 105–134.

Bobrow, D.G. and Hayes, P.J., 1985. Artificial intelligence – where are we? Artif. Intell., 25: 375–415.

Bobrow, D.G. and Stefik, M.J., 1986. Perspectives on artificial intelligence programming. Science, 231: 951–957.

Coulson, R.N., Folse, L.J. and Loh, D.K., 1987. Artificial intelligence and natural resource management. Science, 237: 262–267.

Danell, K. and Ericson, L., 1986. Foraging by moose on two species of birch when these occur in different proportions. Holarct. Ecol., 9: 79–84.

Dawkins, R., 1976. Hierarchical organization: a candidate principal for ethology. In: P.P.G. Bateson and R.A. Hinde (Editors), Growing Points in Ethology. Cambridge University Press, Cambridge, pp. 7–54.

De Jong, M.C.M. and Saarnemaa, H., 1985. A mechanistic simulation model for the movement and competition of bark beetle larvae (Coleoptera, Scolytidae). Ecol. Modelling, 27: 109–138.

Feigenbaum, E.A. and McCorduck, P., 1984. The Fifth Generation (2nd Edition). Pan Books, London, 378 pp.

Graham, L.A., 1986. HAREMS: A generalized data base manager and simulator for barrier island feral horse populations. Coop. Park Studies Unit Tech. Rep. 32, 49 pp.

Jones, R.E., 1977. Movement patterns and egg distributions in cabbage butterflies. J. Anim. Ecol., 46: 195–212.

Loehle, C., 1987. Applying artificial intelligence techniques to ecological modelling. Ecol. Modelling, 38: 191–212.

Näslund, B.-Å., 1986. Simulation of damage and mortality in young stands and associated stand development effects. Swed. Univ. Agric. Sci., Dep. Silvic. Rep. 18, 147 pp.

Owen-Smith, N. and Novellie, P., 1982. What should a clever ungulate eat. Am. Nat., 119: 151–178.

Pascoe, G.A., 1986. Elements of object-oriented programming. Byte, 11(8): 139–144.

Repo, S. and Löyttyniemi, K., 1985. The effect of immediate environment on moose (*Alces alces*) damage in young Scots pine plantations. Folia For. 626, 14 pp.

Sabelis, M.W., 1981. Biological control of two-spotted spider mites using phytoseiid predators. Part I. Modeling the predator–prey interaction at the individual level. Pudoc, Wageningen, 242 pp.

Sainio, P., 1956. On the feeding of the elk in winter. Silva Fenn. 88, 24 pp.

Staddon, J.E.R., 1983. Adaptive Behavior and Learning. Cambridge University Press, Cambridgei, 555 pp.

Stefik, M. and Bobrow, D.G., 1986. Object-oriented programming: themes and variations. AI Mag., 6: 40–62.

Stephens, D.W. and Krebs, J.R., 1986. Foraging Theory. Princeton University Press, Princeton, NJ, 247 pp.

Stone, N.D., 1987. Artificial intelligence approaches to systems modeling. ASAE Pap. 87-1083, American Society of Agricultural Engineering, St. Joseph, MI, 21 pp.

Toates, F.M., 1986. Motivational Behavior. Cambridge University Press, Cambridge, 188 pp.

Waage, J.K., 1979. Foraging for patchily distributed hosts by the parasitoid, *Nemeritis canescens*. J. Anim. Ecol., 48: 353–371.

Winston, P.H., 1984. Artificial Intelligence (2nd Edition). Addison-Wesley, Menlo Park, CA, 524 pp.