Behavioral Models of the Praying Mantis as a Basis for Robotic Behavior [1]

Ronald C. Arkin [a] Khaled Ali [a] Alfredo Weitzenfeld [b]
Francisco Cervantes-Perez [b]

[a] *College of Computing, Georgia Tech, Atlanta, GA, 30332-0280 U.S.A.*

[b] *Depto. Academico de Computacion, Instituto Technologico Autonomo de Mexico Mexico City, MX*

Abstract

Formal models of animal sensorimotor behavior can provide effective methods for generating robotic intelligence. In this article we describe how schema-theoretic models of the praying mantis derived from behavioral and neuroscientific data can be implemented on a hexapod robot equipped with a real-time color vision system. This implementation incorporates a wide range of behaviors, including obstacle avoidance, prey acquisition, predator avoidance, mating, and chantlitaxia behaviors that can provide guidance to neuroscientists, ethologists, and roboticists alike. The goals of this study are threefold: to provide an understanding and means by which fielded robotic systems are not competing with other agents that are more effective at their designated task; to permit them to be successful competitors within the ecological system and capable of displacing less efficient agents; and that they are ecologically sensitive so that agent-environment dynamics are well-modeled and as predictable as possible whenever new robotic technology is introduced.

Key words: Schema-based control; behavior-based robotics; praying mantis behavior

1 Ecological Robotics

The study of sensory guided behaviors in living animals has become significant not only for scientists working in neuroscience and computational neuro-

[1] This research was conducted under funding provided by the NSF under Grant #IRI-9505864 and CONACyT under Grants #546500-5-C006A and #546500-5-C018A)

science, but also for those studying robotics and distributed artificial intelligence who are using functional principles generated from the study of living animals as models to build computer-based automata that display complex sensorimotor behaviors. The research reported in this article, which follows these lines, is tied together by a common theoretical framework: schema theory.

We strive to gain a better understanding of the relationship an agent must maintain with its surroundings. Ecological robotics refers to the incorporation of aspects of the relationship a robot maintains with its environment into its control system (i.e., its ecology). One means for developing such a control system is by exploiting models of behavior developed by ethologists or neuroscientists. Although considerable research has been conducted in the modeling of neural controllers based on animal models (e.g., [8,15,47]), incorporation of environmental interactions has been far less studied within the robotics community (e.g., [50]). Although some work has been undertaken within the artificial life arena [39,36], almost all of this work has been conducted in simulation or at best on primitive robotic implementations. All too often, these approaches lack both a strong biological basis for their working assumptions and any formal underpinnings (neural, behavioral, and computational) for the results they obtain. It is our contention, that the use of schema theory [2] and neurophysiological and ethological modeling methods can provide credible, generalizable, and useful results in this domain.

Most of our previous research has considered the behavioral process dynamics within an agent, and in some limited ways, collective behavior among similar agents [12]. In so doing we have neglected significant aspects of the environment that can and should be incorporated into a systemic view of a robotic agent's place within the world. We now focus on this broader view of robotics, to gain a fuller understanding of how an agent participates with its environmental processes.

McFarland, for some time, has advocated the concept of an agent's ecological niche [48,49]. This view mandates that in order to have a successful robotic implementation, a robot must find its place within the world, i.e., its niche. This niche will enable it to survive and successfully compete with other agents. This perspective holds not only for robotic systems but organizations as well - the novelty lies in its application to robotic systems. McFarland's work has to date heavily emphasized economic pressures, but of course there are also many others.

An in-depth understanding and dynamic modeling of the relationship a robot has with its environment (i.e., the overall ecology) is important to ensure that fielded robotic systems are not competing with other agents that can do the task more effectively and hence prove themselves useless; be successful com-

petitors within the ecological system and can potentially displace less efficient agents; and be ecologically sensitive so that agent-environmental system dynamics are well-modeled and as predictable as possible whenever new robotic technology is introduced. This article examines how such an understanding can be developed through the use of biological models of behavior that are ported onto robotic control systems. It is not the intention that these robots directly displace their biological counterparts, but rather that they become capable of ultimately finding a unique niche in the world within which they can prosper.

In this article, we present both simulation studies and physical results obtained on the implementation of a model of praying mantis behavior on a robotic hexapod equipped with a real-time color vision system. As we are working with models generated by animal scientists, we hope that not only will these results have value within the robotics community in terms of providing a path for generating intelligent behavior in machines, but that they may also serve as a basis for feedback for stimulation, regeneration, and refinement of the animal models themselves.

2 Background and Motivation

The relationships between the interdisciplinary areas in this research are depicted in Figure 1. Biological data are used to generate abstract schema models that can either be directly imported into the our robotic software control system generator (*MissionLab*) [43,44], or abstracted further into the context of neural networks (NSL) and then translated to abstract behavioral schemas (ASL) prior to importation into a specific robot control program. These software tools (*MissionLab*, ASL, NSL) that embody our notion of schema theory are described further in Section 2.2. First, however, we present the biological motivation for our system.

2.1 Neuroscience and Ethology

On the biological side, we have been studying visuomotor coordination phenomena in amphibia (toad) and insects (praying mantis). These animals live within a three dimensional environment, rich in multiple modes of sensory signals, but their behavior is mainly guided by visual information. From an ecological point of view, these animals react to visual environmental domains of interaction which can be classified into two groups: moving and non-moving objects. Diverse stationary objects may influence the animal's next action which, in general, is directed to improve the animal's survival chances. For

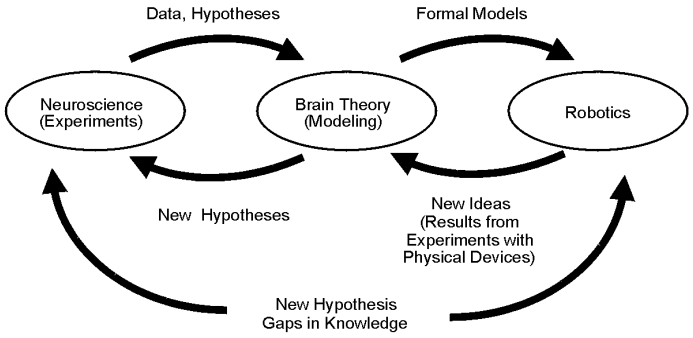

Fig. 1. Framework for the study of models of biological organisms as a basis for robotic control.

example, frogs move towards zones in the visual field where blue is preponderant, a situation that might be associated with the presence of prey to eat, and of water to maintain its body humidity [37]. In the case of the praying mantis (Fig. 2), when it is placed in an open field with no mobile objects around, it executes several motor actions that conform to what we have called the *chantlitaxia* (i.e., in search of a proper habitat) behavior.

Fig. 2. Overhead outdoor view of a praying mantis.

Different moving objects may elicit a specific behavior from these animals. For example:

- During the mating season, the presence of a female frog in the male's visual field yields an orienting response towards the female, followed by an approaching action if the female is far away, or a clasping behavior if the female is within reaching distance in the frontal part of the visual field.
- A predator-like stimulus may yield one of several avoidance behaviors depending upon its parametric composition. In amphibia, a large flying stim-

ulus close to the animal releases a ducking response [30,31,38,41], whereas, in the mantis, a similar stimulus elicits a deimatic behavior (i.e., the mantis stands up, and opens the wings and forearms displaying a posture that demonstrates a much bigger size than it actually has) [46].

- The presence of potential prey may elicit one of several actions, depending on the spatio-temporal relationship between the prey and the animal (i.e., amphibia or insect). These include an orienting response towards the part of the visual field where the prey is located, followed by an approaching behavior when the prey is located far afield in the frontal part of the visual field. In the case of amphibia [29] a snapping response follows, or a grasping response in the praying mantis [14], if the stimulus is within reaching distance.

Our group has developed theoretical (i.e., neural and schema-theoretic) models of visuomotor coordination phenomena in amphibia [22,18,40,23]. These results have allowed us to postulate the underlying mechanisms of visuomotor integration and have developed into a parallel distributed neural processing system, in which those neural structures receiving direct input from the retina (retinula in the insects) represent more than a visual (sensory) map. Rather they are the site of integration of external stimuli coming through the retina, and signals generated in brain structures that might be involved in the processing of information related to motivational factors and learning processes. The animal's response towards, or away from, visual stimuli could be described as the integration of neural signals generated by dynamic systems working at different time scales:

- *Activation Dynamics:* signals activated in the animal's nervous system by the presence of a visual stimulus in its visual field (e.g., prey-catching may take 100s of milliseconds).
- *Motivational Dynamics:* some changes in factors that modulate the animal's motivation to display certain behaviors (e.g., prey catching intensity varies depending on the time of the day).
- *Time-varying Dynamics:* learning processes require at least the occurrence of one activation dynamic process to modify future interactions with the same kind of stimulus (e.g., the bee sting at the toad's tongue is enough for preventing that toad from catching bees in the future [31]), whereas other learning processes require a longer training (e.g., stimulus-specific habituation is accomplished after stimulating repetitively the toad with the same stimulus for few minutes to an hour and a half) [32,20].

Regarding motivational changes of visually guided behavior, an experiment was conducted [21] where the combined effect of changes in motivation and in the stimulus configuration (i.e., form and velocity) was evaluated. Two groups of animals were used that were fed at 19:00 hrs, and stimulated with visual dummies (i.e., worm-like and square stimulus) the following day at different

times: a) toads stimulated at 9:00 am showed a low motivation to display prey-catching behavior; and b) those stimulated at noon displayed a high number of predatory responses (high motivation). These results offered the conclusion that a toad's motivational state modifies the efficacy of a prey-like visual stimulus to elicit the animal's prey-catching behavior.

2.2 Frameworks for Expression

There have been a number of attempts to define a methodology for the analysis of large complex dynamic systems such as these. One approach is schema theory [2] which lays down the conceptual framework for knowledge representation inspired from biological and cognitive studies. Figure 3 shows the software tools used within this project for expressing computational models and robotic behavior (ASL, NSL, and *MissionLab*) and their relationship with the underlying biological models.

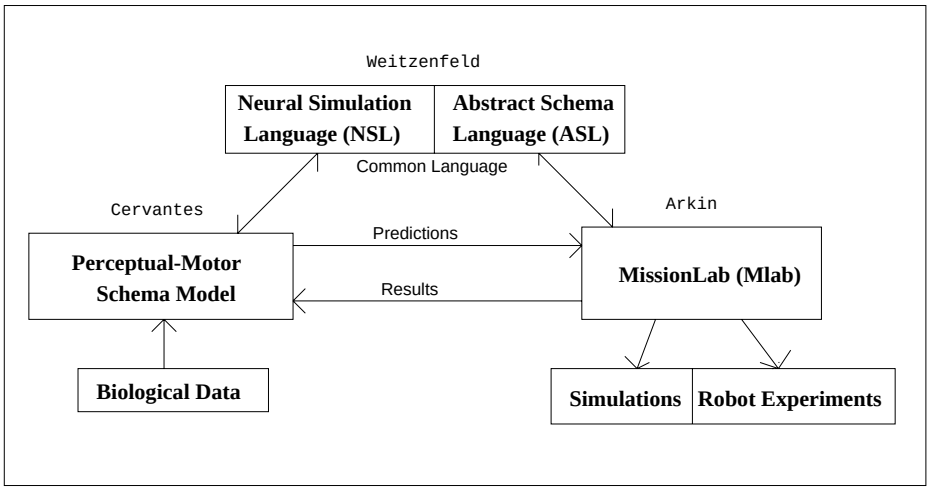

Fig. 3. Schema-based modelling tools connecting neuroscientific studies with robotic control systems.

2.2.1 ASL

The (Abstract Schema Language (ASL) [53] follows a hierarchical model, enabling top-down and bottom-up designs, supported by a concurrent language permitting a distributed implementation, while integrating neural network processing. ASL's main characteristics are its dynamic and asynchronous nature, and the inclusion of dynamic schema assemblages as the basis for composition. Essentially a schema is a template from which many instances can be created, in a fashion similar to that of object-oriented systems. The behavioral description of a schema describes how an instance of that schema will behave in response to external communications. As action and perception progress,

certain schema instances need no longer be active, while new ones are added as new objects are perceived and new plans of action are elaborated. A schema assemblage, the basis for aggregation, is a network of schema instances, and it may be considered a schema for further processing. Since a schema may be decomposed into any number of component schemas, there may be virtually any level of abstraction. The major properties introduced in ASL:

- **Delegation**: Schema implementation may be chosen in a dynamic way, via the ASL high-level language or by delegating processing to neural networks.
- **Wrapping**: Previously developed code may be statically linked within a schema.
- **Heterogeneity**: Incorporation of two different programming concepts, neural processing and procedural processing, into a single model.
- **Encapsulation**: A schema instance includes a public interface while all data and its particular implementation are internal, thus providing flexibility and extensibility since local changes to its internal data structure and implementation do not affect its interaction with other schema instances. Furthermore, the communication abstraction of input and output ports permits greater flexibility in communication and in the design of schema architectures.
- **Reusability**: Following object-oriented abstractions, such as inheritance, the definition of schemas as shared templates from which schema instantiation takes place, permits their reusability in new schema definitions.

In order to integrate neural processing with schemas into a single computational model, it was first necessary to design a distributed environment where neural entities can be mapped as multi-granular concurrent processes [55]. In terms of neural networks per se, we have developed the Neural Simulation Language, for simulation of large-scale neural networks [33,56,54].

2.2.2 NSL

The Neural Simulation Language (NSL) is a model development and simulation system for the creation and execution of scalable neural networks. In terms of neural modeling, NSL provides support for varying levels of neural model detail, particularly important to those users doing biological neural modeling. In artificial neural modeling the neural model is usually very simple, where models vary primarily in terms of their network architectures and learning paradigms. While NSL is not particularly intended to support single neuron modeling, NSL does provide sufficient expressiveness for such a task. In NSL, a neural model requires: (1) a set of neural modules defining the entire model; (2) neurons comprised in each neural module; (3) neural interconnections; (4) neural dynamics; and (5) numerical methods to solve the differential equations. In terms of simulation NSL offers both interactivity

and execution efficiency incorporating a compiled language for model development and a scripting language for model interaction and simulation control. During simulation the user interacts with the model through rich graphics and a full menu oriented window interface supporting creation of new models as well as their control and visualization. The simulation process consists primarily in specifying: (1) model parameter assignment; (2) neural network input specification; (3) simulation control specification, such as integration steps and simulation duration; and (4) visualization specifications. As a software system, NSL is built exclusively on object-oriented technology providing extensibility through Java and C++ for users that may want to develop applications under specific programming environments or to integrate with other software or hardware. Furthermore, NSL runs on a large number of platforms and supports users with different ability levels. [57]

2.2.3 MissionLab

To support the ease of generation of behavior-based robot control systems the *MissionLab* mission specification system has been developed [44]. An agent-oriented philosophy is used as the underlying methodology, permitting the recursive formulation of societies of robots. A society is viewed as an agent consisting of a collection of either homogeneous or heterogeneous robots. Each individual robotic agent consists of assemblages of behaviors, coordinated in various ways. Temporal sequencing [10] affords transitions between various behavioral states which are naturally represented as a finite state acceptor. Coordination of parallel behaviors can be accomplished via fusion, action-selection, priority, or other means as necessary. These individual behavioral assemblages consist of groups of primitive perceptual and motor behaviors which ultimately are grounded in the physical sensors and actuators of a robot.

An important feature of *MissionLab* is the ability to delay binding to a particular behavioral architecture (e.g., schema-based or MRPL) until after the desired mission behavior has been specified. Binding to a particular physical robot also occurs after specification, permitting the design to be both architecture- and robot-independent.

MissionLab's architecture appears in Figure 4. Separate software libraries exist for abstract behaviors, and the specific architectures and robots. The user interacts through a design interface tool (the configuration editor) which permits the visualization of a specification as it is created. Individual icons correspond to behavior specifications which can be created as needed or preferably reused from an existing repertoire available in the behavioral library. Multiple levels of abstraction are available, which can be targeted to the abilities of the designer, ranging from whole robot teams, down to the configuration description language for a particular behavior, with the higher levels being those easiest

to use by the average user.

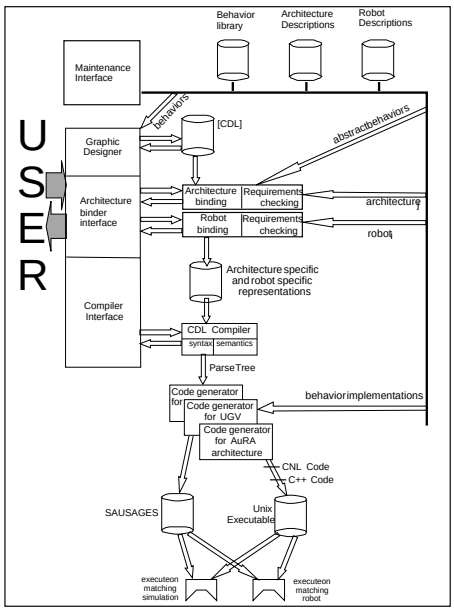

Fig. 4. *MissionLab* system architecture.

MissionLab affords the additional capacity for multi-robot testing. The results obtained using the animal models have been tested first in simulation studies and were then ported to real robotic platforms through the use of tools found within this software testbed. *MissionLab* is available via the world-wide web at http://www.cc.gatech.edu/aimosaic/robot-lab/research/MissionLab.html.

An important goal in pursuing a powerful software environment to enable both simulation and robot testing of ecological agents is the coupling of ASL/NSL with *MissionLab* in a tightly integrated environment. An early prototype was developed a few years ago integrating neural networks modeling in NSL within a sensorimotor testbed in robotic applications [33]. ASL/NSL and *MissionLab* differ in some of their basic modeling capabilities (neural networks in NSL and robot control in *MissionLab*), yet they have a common unifying theme in schema-based modeling providing a basis for integrating the two systems.

3 Schema-based Praying Mantis Models

Schema theory is a powerful and expressive means for describing behavior, both neuroscientific [1] and robotic [4,42]. Schemas themselves are distributed concurrent processes, charged with enacting the internal behavioral goals and

intentions of the agent in response to external stimuli. The output of these processes can be combined in a host of ways including, for example, priority-based arbitration (subsumption) [16], behavioral fusion [4], and action-selection [45] to name a few.

A series of models, using schema theory, represents an agent's participation with its world. This involves the extension of our common schema-theoretic approach to incorporate external, as well as internal, processes. Fortunately, schema-theory is quite amenable to this strategy, which we demonstrate initially from a modeling perspective, then using robotic simulations, and ultimately implementations. Steels and McFarland [52] have also begun to explore these issues experimentally, but have not brought powerful modeling tools to the table, nor have looked closely at the biological underpinnings of the problem. Our research does both. There is also a relationship to the school of ecological psychology [28] as formulated by Gibson [35] and developed further by Neisser using schema theory [51]. We have previously shown a connection between affordance-based perception and perceptual schemas [6]. We continue to explore this avenue in light of new biological data.

One study assesses how praying mantises react to different aspects of their environment, by placing them in various experimental situations. We are interested in what strategies they might use, and whether or not these strategies are fixed or may be modulated by the outcome of the animal's interactions with diverse environmental situations. We have developed schema-theoretic models of the chantlitaxia behavior, including results of an ontogenetic study that suggest that the linkage between the integration of sensory signals and the activation and control of global motor patterns is not parametric [34], as Lara and coworkers postulated [40], but rather it is a modulated process (i.e., function) that varies depending on the state of a diversity of factors (e.g., animal's age). Some of our preliminary observations suggest that, as the praying mantis grows, it might use this kind of behavior to move to a proper habitat: young mantises live in the bushes, while adult ones stay in the higher part of trees.

As both robotic systems and neural network models increase in their sophistication, it becomes crucial to simulate and analyze the interaction of an increasing number of functional and structural subsystems. While progress has been made in modeling the less sophisticated robotic and single neural network systems, we have reached the stage where it is necessary to define a framework for developing multi-level neural architectures in particular, in their application to robotics. Thus, the design and implementation of a schema-based model has been undertaken, utilizing large neural networks developed in a hierarchical fashion for the robotics domain. This involves the extension of our preliminary ASL model to address issues arising from the integration of ASL with neural network simulation in NSL, while implemented in a distributed environment.

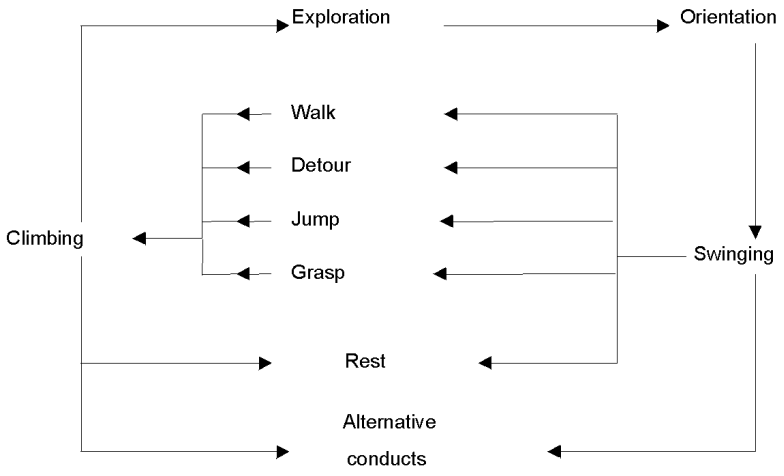

Fig. 5. Ethogram of praying mantis behavior. (From [4])

Our research has focused on ethological models of visuomotor behavior for the praying mantis. An ethogram for this creature appears in Figure 5. It encompasses a wide range of behaviors ranging from exploration, locomotion, rest, eating, mating, and defensive activities. From this model, we have implemented a subset of these behaviors including simple locomotion incorporating obstacle avoidance, prey acquisition (for eating), predator avoidance (for survival), mating, and chantlitaxia behavior [19], where the agent searches for a hospitable environment, which in the case of the mantis is low brush or bushes when young and trees when older.

Revisiting the high-level goals of this research: (a) to provide strong task/environment fit with agent, enhancing its survivability (i.e., finding a suitable ecological niche), (b) to make robots successful competitors that can potentially displace less efficient agents; and (c) to make them sufficiently sensitive to adaptation by including suitable models of agent-environment system dynamics. In our attempt to provide solutions to these problems, we develop a series of models of specific biological organisms that facilitate this study. In many animals (e.g., toads and the praying mantis) visuomotor integration implies a complex transformation of sensory signals, since the same locus of retinal activation might release different types of motor actions: some directed towards the stimulus (e.g., prey-catching); and others directed towards an opposite part of the visual field (e.g., predator avoidance). Furthermore, the efficacy of visual stimuli to release a response (i.e., type of behavior, intensity, and

frequency) is determined by a series of factors:

(1) The stimulus situation (e.g., form, size, velocity of motion, the spatio-temporal relationship between the stimulus and the animal)
(2) The current state of internal variables of the organism, especially those related to motivational changes (e.g., season of the year, food deprivation, time interval between feeding the animal and the time of experimentation)
(3) Previous experience with the stimulus (e.g., learning, conditioning, habituation).

We have chosen to study the praying mantis and its environmental dynamics. Models have been developed using schema theory as the basis for expression that have led to the fielding of these results on a Hermes II robot for testing (Fig. 6).

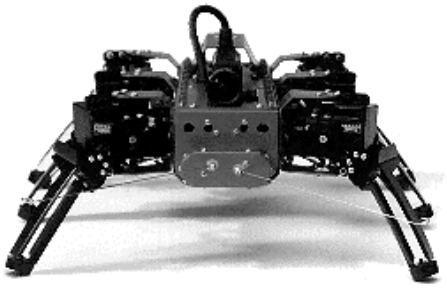

Fig. 6. Hermes Robot (photograph courtesy of IS Robotics, Somerville, MA)

.

The repertoire of mantis behavior appears in the ethogram depicted in Figure 5. In our modeling efforts for mantis behavior we have abstracted away much of this complexity. Our model encompasses 4 different visuomotor behaviors (that are also surprisingly similar to frog ethology):

- *Prey Acquisition:* This behavior first produces orienting, followed by approach (if sufficiently far), then grasping by the mantis when the target is within reach.
- *Predator Avoidance:* At the most abstract level, this produces flight of the insect. But when considered in more detail there are several forms of avoidance behavior. A large flying stimulus can yield either a ducking behavior or a fight-type response referred to as deimatic behavior where the insect stands up and opens its wings and forearms to appear larger than it is.
- *Mating:* This is an attractive behavior generated by a female stimulus during the mating season producing an orienting response in the male followed by approach, then actual mating.
- *Chantlitaxia:* This involves an agent's search for a proper habitat (i.e., finding its niche). The praying mantis climbs to higher regions (e.g., vegetation) when older, actively searching for a suitable place to hunt.

This ethologically-derived schema model initially starts as a purely abstract depiction of behavioral relationships (Fig. 7 top). This is then translated into an operational schema model that more effectively depicts the data flow within the system (Fig. 7 middle). Finally it is implemented within the *MissionLab* robot mission specification system for testing in simulation (Fig. 7 bottom).

A graphical depiction of the simulation output of the model shown in Figure 7 appears in Figure 8. The upper figure shows the mantis moving away from the large gray disk (its hiding place) in search of food and mates as the internal motivational variables representing its hunger and mating needs change over time. A three dimensional window at the bottom of the figure also shows the simulation world that the mantis is housed in. Predators can also be introduced into the simulation and the agent's responses observed and recorded. The intent of the simulation is to serve as the way station for testing the control algorithms that have been ported to the Hermes II robot (Fig. 6), which is standard practice for the *MissionLab* system.

Our model also incorporates motivational variables (shown below in parentheses) which affect the selection of motivated behaviors such as predator avoidance (fear), prey acquisition (hunger) and mating (sex-drive). These variables are currently modeled quite simply (described in Section 5) but may be extended to incorporate factors such as diurnal, seasonal, and climatic cycles and age-related factors. This simplified model, depicted in Figure 9, serves as the basis for the implementation described in this article. It is derived from the model of praying mantis behavior developed by Cervantes-Pérez [17]. (shown at the top of Figure 7). This model makes a weak commitment to an action-selection mechanism (modeled as lateral inhibition in Fig. 7 top) for arbitration between the motivated behaviors: prey acquisition, predator avoidance, and mating; while using a colony-style architectural strategy [24] for arbitration between the winner of the motivated behaviors, obstacle avoidance, and chantlitaxia.

The outputs of these behaviors are encoded in the form of two percentages, one representing the forward motion as a percentage of the maximum forward speed, and the other representing the turning motion as a percentage of the maximum turning speed (with negative percentages representing turns in the other direction). In the implemented version described below, vision provides the stimuli for chantlitaxia and the set of motivated behaviors, while obstacle avoidance is triggered by the hexapod's whiskers.

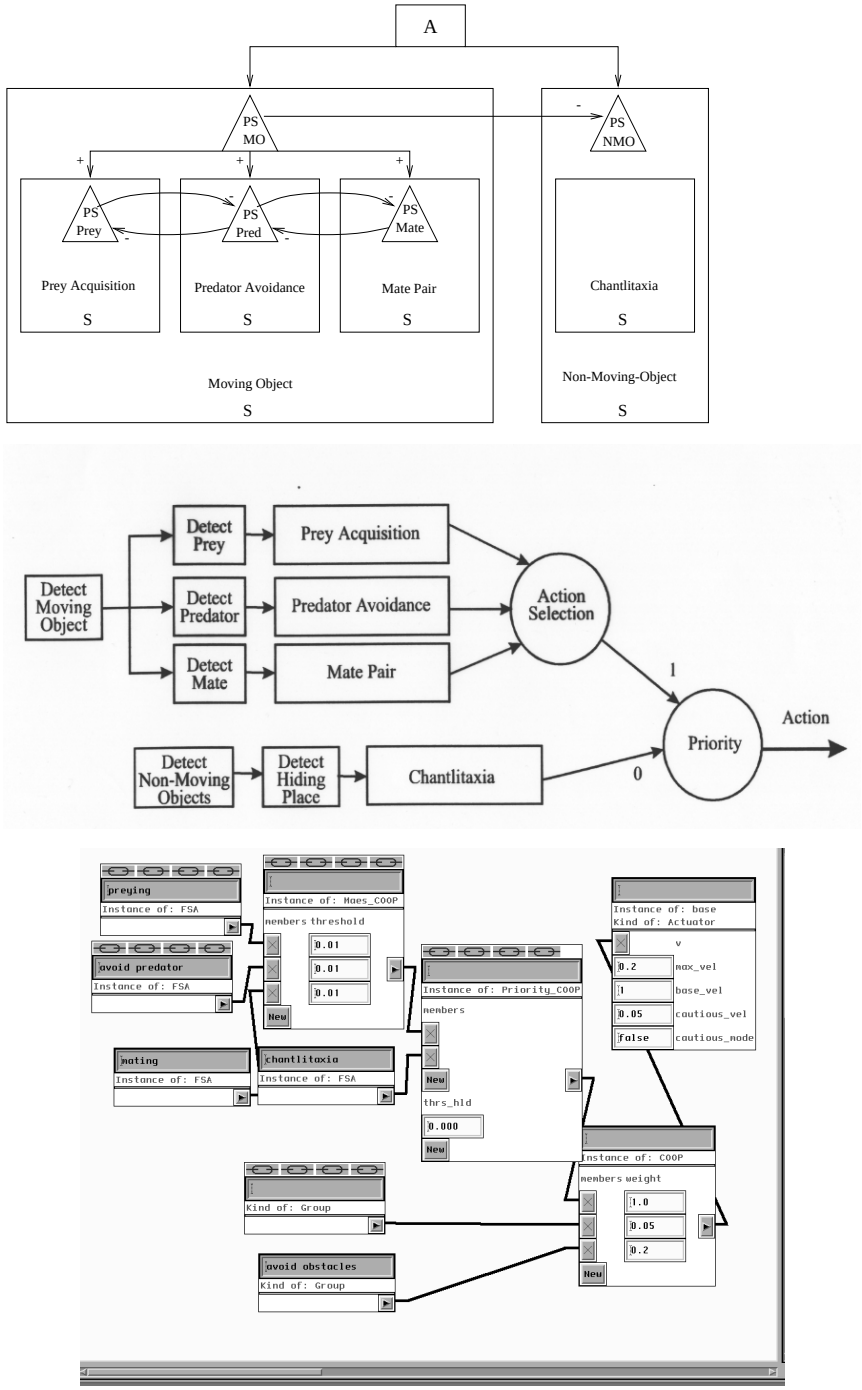

Fig. 7. Visuomotor Behaviors of Mantis.
(Top) Model-developed by Cervantes.
(Middle) Abstract adapted model.
(Bottom) Model as implemented in *MissionLab*.

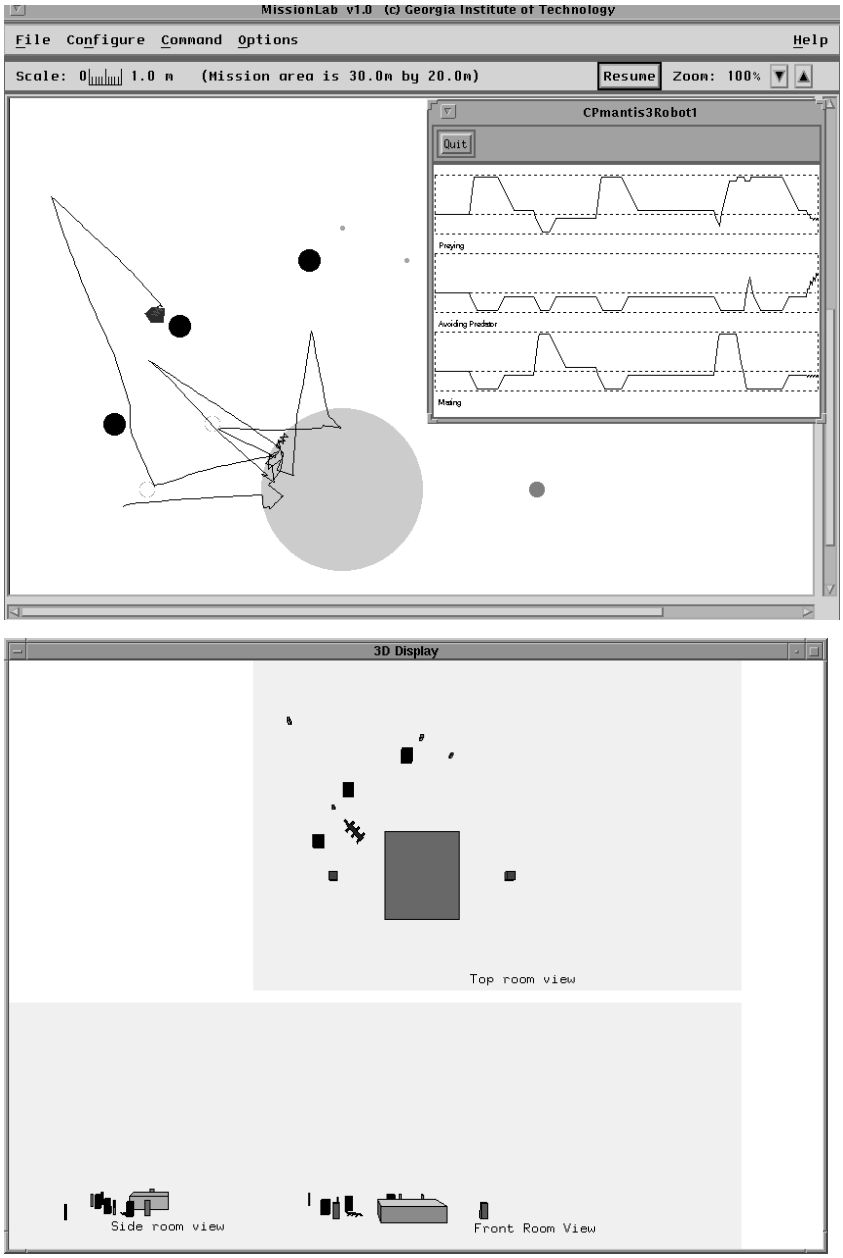

Fig. 8. *MissionLab* Mantis Simulation.
(Top) Two-dimensional view.
(Bottom) Three dimensional view
.

3.1 Schema models on Robot

To realize these models within robotic systems, we have adopted the framework of the Autonomous Robot Architecture (AuRA) [5], using motor schemas to provide a behavioral implementation. Motor schemas are selected and instantiated in a manner that enables the robot to interact successfully with

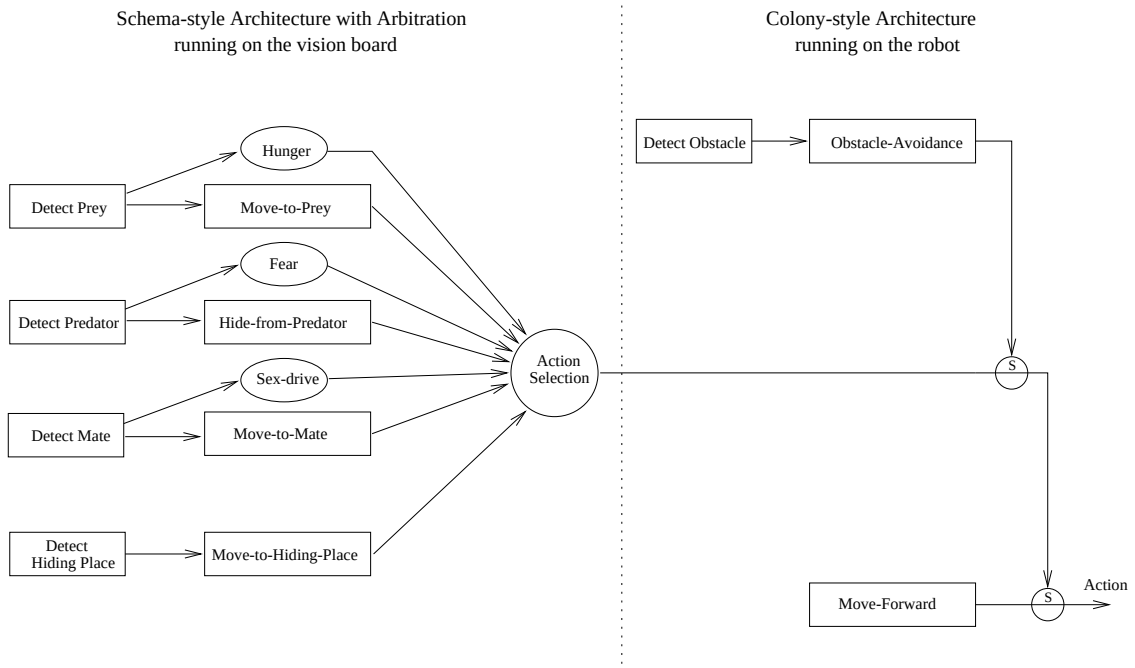

Fig. 9. The model implemented on Miguel, our Hermes robot. It incorporates aspects of action-selection arbitration and colony-style architecture suppression. The interactions of the modules is based on a model of the praying mantis's behavior developed by Cervantes-Pérez [17].

unexpected events while still striving to satisfy its higher level goals. Multiple active schemas are usually present, each producing a velocity vector driving the robot in response to its perceptual stimulus. The resultant vectors are combined as necessary according to the underlying animal model and normalized to fit within the limits of the robot vehicle, yielding a single combined velocity for the robot. These vectors are continually updated as new perceptual information arrives, with the result being immediate response to any new sensory data. Advantages of this approach include rapid computation and the ability to be mapped onto parallel architectures making real-time response easily attainable. Modular construction affords ease of integration of new motor behaviors simplifying both system maintenance and the ease of transfer to new problem domains. Motor schemas readily reflect uncertainty in perception, when such a measure is available, and also react immediately to environmental sensor data. These factors all contribute to the needs of a motor control system that will successfully assist a robot's intentional goals.

Our earlier work [9,12], describes our position on integrating biological bases for multiagent teams. Even earlier work from our laboratory [3,6,8] shows the influence that biological models have had on our control methods for robotic systems. Exemplified by our multiagent research, we have demonstrated a system which uses homogeneous units to carry out tasks of foraging, grazing, and consuming objects in a cluttered world [12,13]. We have extended our

research in schema-based navigation [4,10] to include patterned sequences of behaviors [1] and their use in multiagent navigation. This approach to reactive navigation has been previously demonstrated in a variety of application domains. Emergent behavior is evidenced as the phenomena of recruitment, the shared effort of many robots to perform a task, which occurs even in the absence of communication between the agents [7]. Foraging consists of searching the environment for objects (referred to as attractors) and carrying them back to a central location. Consuming requires the robot to perform work on the attractors in place, rather than carrying them back. Grazing is similar to lawn mowing; the robot team must adequately cover the environment. More recently we have focused on complex mission scenarios that utilize bounding and traveling overwatch and formation maintenance, among other objectives [43]. These complex missions can be constructed from the more basic tasks described above.

4 Visualization of Behavior

Modeling and simulation of biologically inspired agents can become very complex as more realistic behaviors are pursued. Providing powerful tools for modeling and simulation can manage this complexity, in particular through visualization. While existing visualization techniques are appropriate for simple models at single modeling levels, more complex models incorporating a number of behavior, schema agents, and neural levels, provides a greater challenge.

In the kind of behavior modeling and simulation described in this article, visualization plays an important role in both generating the models and analyzing the resulting robotic agent behavior. Visualization takes place at different levels, depending on whether it relates to the modeling stage or during simulation (or actual robot execution).

During model development, behaviors are defined in terms of interconnected schema agents, drawn from *MissionLab* (e.g., Figure 7 Bottom). Each schema is defined either recursively in terms of further schema networks (or finite state automata) or alternatively refined in terms of neural blocks for ASL/NSL as shown in Figure 10. These neural blocks are refined until each block represents an actual neuron or neural array. In general, at the modeling level, behavior is not actually visualized but its underlying representation is.

The Schematic Capture System (SCS) in ASL/NSL facilitates the creation of modular and hierarchical neural networks by providing a graphical interface to design modules consisting of input and output ports, internal module instantiations and a local interconnectivity specification (Fig. 10). Ports are

specified by triangles while modules are specified by boxes. Once all modules are specified for a particular model in ASL/NSL, the editor generates the model code to be compiled by either the C++ or Java ASL/NSL simulator [11]. The example illustrated in Figure 10 is taken from a "cortical model for generation of spatially accurate sequential saccades" [27] and shows four input ports, a single output port and two locally instantiated submodules, all interco

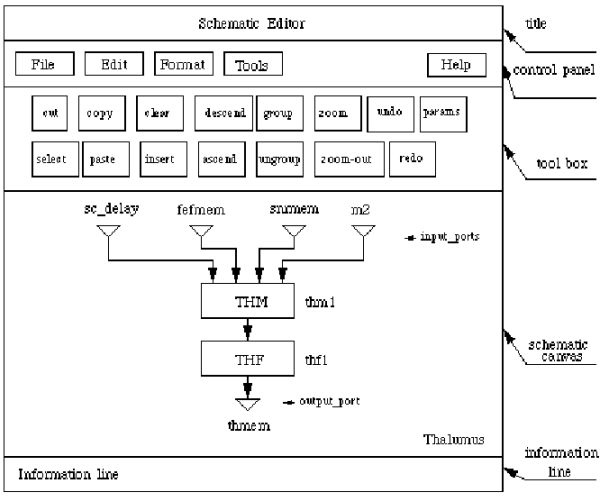

Fig. 10. ASL's view of saccade's thalamus schema with linkage between schema levels.

During simulation, however, we can visualize model behavior at different levels of granularity, starting from whole agent behavior as part of its environment, all the way down to the behavior of every individual neuron taking part in the overall behavior. At the top level we can see the agent in its environment, virtual or real, as shown in Figure 8 (2d and 3d virtual environments) and Figures 14-17 (real robot). At this level interaction is primarily in the form of user control: starting, executing and stopping the simulation, with the modification of the environment and the location of the agent in it, but not the modification of the actual agent itself.

Descending to the next level, we visualize the network of interconnected schema agents, corresponding also to Figure 7 (bottom), as seen from *MissionLab*. At this level, interaction takes place by changing the network connection parameters or the control parameters for individual schema agents. Note that if network connectivity is to be modified or additional schema agents are to be added to the network then we would typically revert to the modeling stage.

In trying to understand why an agent behaved in a particular manner, corresponding agent behavior has to be analyzed. Since schema agents are recursive,

input and output data analysis is required for each schema in a recursive fashion, together with data passed between interconnected schema agents at the same level or between delegated schemas at different levels. Further down, ASL provides the linkage to neural visualization in NSL by letting the user "navigate" in the schema hierarchy until reaching a bottom level neural component. An important concern with data generated at this level, as well as in the higher levels, are the time intervals and the synchronization between data passed between different schema agents or neural components.

At the lowest granularity level, that of neurons and neural networks, visualization takes the form of temporal and spatial graphs of various dimensions and forms, corresponding to neural input, output (firings), and membrane potentials. Again, time intervals play a major role across multiple neurons and neural networks. For example, figure 11 shows two dimensional spatial and temporal graphs from NSL for a maximum selector model [26] (this model is particularly important in its relation to prey or predator selection when more than one of them is present). Neural modules in ASL/NSL describe neural networks composed of a number of interconnected neural layers. Each layer consists of an array of neurons functionally visualized through spatial or temporal displays. In Figure 11, the top portion of the display shows a spatial graphic output for each neuron in a layer, where the neural activity level is seen as a function of the size of the black box in the graph. On the other hand, the bottom part of the display shows a temporal display that includes the activity history for each neural layer element. This example is taken from the "maximum selector" or "winner-take-all" model [26] where two values are input to the network, and only a single value stays above zero (dashed line) after processing for all elements in the layer.

Figure 12 shows a three dimensional spatial graph, also from NSL, illustrating the graphical output associated with a barrier input to a 2d array of neurons active during detour behavior [25]. The left figure shows a two dimensional frog retinal view of a barrier consisting of a number of fenceposts with gaps in between. The right hand figure shows a three dimensional view of the barrier as processed further by a neural network layer in the "prey acquisition with detour" model [25]. This output could then be associated with an avoid-obstacle behavior in a robot.

While in practice we do not model beyond simple neurons when simulating large systems, in theory one could visualize the anatomical and physiological temporal behavior of neural synapses and electrical or chemical concentrations in detailed neurons.

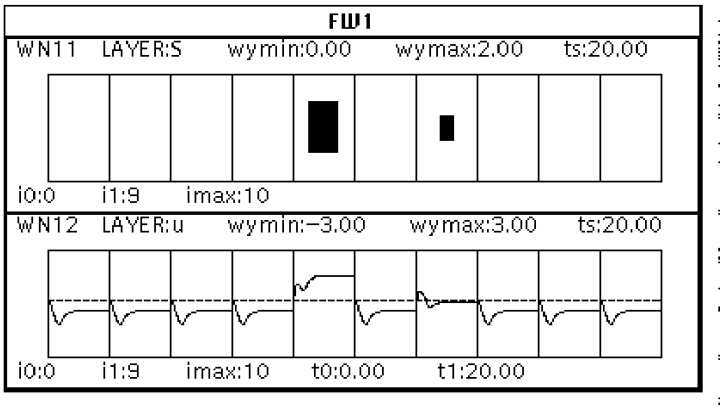

Fig. 11. NSL spatial and temporal displays from maximum selector model.

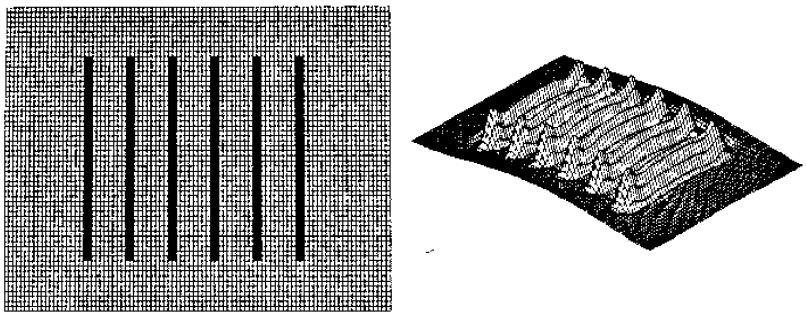

Fig. 12. NSL 2- and 3-dimensional spatial displays from prey acquisition with detour model.

5 Robotic Implementation

The model of praying mantis behavior (Fig. 9) has been implemented on our robot Miguel. Miguel is a Hermes II hexapod robot manufactured by IS Robotics. It has a color camera mounted at its front, feeding live video into a Newton Research Labs Cognachrome vision System. Additionally, Miguel has two whiskers mounted at its front that serve as contact sensors, infrared proximity sensors on each leg, and contact sensors on its underside.

The portion of Cervantes-Pérez's model that corresponds to moving-objects as shown on the left side of Figure 9 is implemented on the Cognachrome vision processor. In our implementation, however, rather than responding to

movement, the system responds instead to colors. Green objects represent predators, purple objects represent mates, orange objects that are at least twice as tall as they are wide represent hiding-places, and all other orange objects represent prey. Figure 13 shows the algorithm that runs on the vision processor. The robot maintains three internal variables that represent the robot's hunger, fear, and sex-drive. Initially, the values of each of these variables is zero. The hunger and sex-drive levels increase linearly with time, with the hunger arbitrarily increasing at twice the rate as the sex-drive. When the robot has contacted a prey or mate, the robot is considered to have eaten or mated with the object, and the relevant motivational variable resets to zero. Contact is determined by the position of the prey or mate blob in the image captured by the camera mounted on the front of the robot. In this case, the object is considered to be contacted when the bottom of the object blob is in the lower five percent of the image. The fear level remains zero until a predator becomes visible. At that time, the fear variable is set to a predetermined high value. When the predator is no longer visible, the fear level resets to zero. It is possible to incorporate more complex modeling, including habituation to stimuli, but that remains for future work.

```
(1) Increment sex-drive and hunger, and set fear.
      sex-drive := sex-drive + 1;
      hunger := hunger + 2;        /* increment hunger twice as fast as sex-drive */
      if predator is detected,
         then fear := 10,000;      /* set fear at a high level */
         else fear := 0;           /* reset fear when no predator is visible */

(2) Check if mate or prey are close enough to eat.
      if mate is contacted,
         then sex-drive := 0;      /* reset sex-drive after mating */
      if prey is contacted,
         then hunger := 0;         /* reset hunger after eating */

(3) Each behavior produces a direction or Stop command, based on the input from its corresponding perceptual schema.
    (a) move-to-prey, move-to-mate, and move-to-hiding-place
          if prey/mate/hiding-place blob is in upper-right of image,
             then output Forward Right;
          if prey/mate/hiding-place blob is in middle-right or lower-right of image,
             then output Right;
          if prey/mate/hiding-place blob is in upper-left of image,
             then output Forward Left;
          if prey/mate/hiding-place blob is in middle-left or lower-left of image,
             then output Left;
          if prey/mate/hiding-place blob is in middle, upper-middle, or lower-middle of image
             then output Forward;

    (b) hide-from-predator
           if predator is detected,
              then output Stop,
              else output DONT-CARE;

(4) Choose an output from a behavior, to pass along to the robot.
      if there is an associated stimulus for the motivational variable with greatest value,
         then output direction from behavior corresponding to this variable,
         else if there is an associated stimulus for the motivational variable with second greatest value,
            then output direction from behavior corresponding to this variable,
            else if there is an associated stimulus for the motivational variable with third greatest value,
               then output direction from behavior corresponding to this variable,
               else if there is a hiding-place visible,
                  then output direction from {\bf move-to-hiding-place behavior},
                  else do nothing;
```

Fig. 13. Algorithm associated with the schema-style architecture running on the vision processor.

The **move-to-prey** behavior produces a direction that will move the robot toward the largest visible prey, based on the input from the Detect-Prey perceptual schema. Similarly, the **move-to-mate** and **move-to-hiding-place**

behaviors output directions that will move the robot towards the largest mate and the largest hiding-place, respectively, based on the input from their corresponding perceptual schemas. The **hide-from-predator** behavior outputs a Stop command if the Detect-Predator schema indicates that there is a predator visible and outputs DONT-CARE otherwise. The output of these behaviors are discrete directions or commands of the following nature: Right, Left, Forward, Forward Right, Forward Left, Backward, and Stop.

The values of the three internal variables (hunger, sex-drive, and fear) and the currently visible stimuli (prey, mates, predators, and hiding-places) are used by the action-selection module to select the appropriate action to send to the robot's processor. The motivational variable with the greatest current value is chosen. If there is an associated stimulus present, such as a prey for the hunger variable, then the output of the corresponding behavior is sent to the robot. If there is no associated stimulus visible, then this process is repeated with the motivational variable with the next greatest value. If there is no associated stimulus visible for any of the three motivational variables, but there is a hiding-place visible, then the output of the **move-to-hiding-place** behavior is sent to the robot. Otherwise, if there are no predators, prey, mates, or hiding-places visible, then the action selection mechanism does not send any command to the robot. There is no predetermined hierarchy or layering; the action chosen depends directly upon the value of the motivational variables and visible stimuli at that moment in time.

For example, if the current values of the motivational variables hunger, sex-drive, and fear are 568, 343, and 0, respectively, and there are a prey, mate, and hiding-place visible in the environment, then the action-selection module will send the output of the **move-to-prey** behavior to the robot. This is because the hunger variable has the greatest value and a prey is visible. If the motivational variables are the same as above, but only a mate and hiding-place are visible in the environment, then the output of the **move-to-mate** behavior will be sent, since the sex-drive variable has the greatest value among the motivations with currently visible stimuli associated with them.

When prey is visible and the hunger variable is greatest, the appropriate behavioral response is to move towards the detected prey. The response is similar when a mate is visible and the sex-drive level is greatest. When a predator is visible, however, and the fear variable is greatest, the appropriate response is for the robot to freeze in the hope that the predator will not notice its presence.

The remaining part of the model, as shown on the right side of Figure 9, is a colony-style architecture [24] with three levels. This part of the model runs entirely on the processor in the Hermes II. Each level contributes in determining the overall behavior of the robotic agent. The output of higher-

level behaviors can override the output from a lower-level behavior. Effectively, this causes the output of the higher-level behavior to replace the output of the lower-level behavior for a predefined amount of time.

At the lowest level is the move-forward behavior. This behavior directs the robot to move forward in search of some stimuli. If there is a stimulus such as a predator, prey, mate, or hiding-place visible in the environment, then the action-selection module on the left side of Figure 9 will produce an output. This output is translated from a direction or Stop command to spin and speed commands that the robot's built-in Walk behavior understands. These commands will subsume the output of the move-forward behavior for one second. However, if the stimulus is still visible after the one second, the action-selection module will continue to produce an output, and this output will repeatedly subsume the lower-level outputs. Finally, the obstacle-avoidance behavior causes the robot to back up a few steps and then turn to the side when an obstacle is detected by a contact with one of the whiskers. When the obstacle-avoidance behavior generates commands, it suppresses the output from the lower behaviors for one second.

The model shown in Figure 9 was implemented incrementally. First, the move-forward behavior was created. Then the obstacle-avoidance behavior was added. Next, prey tracking behavior was developed. When this was working properly, the predator response, and later the mate tracking and hiding-place tracking behaviors were added.

6 Robotic Results

To test the performance of the implemented model, Miguel wanders around our lab, responding to colored boxes used to represent a predator, prey, mate, and hiding-place. When the execution begins, the hunger, fear, and sex-drive levels are all zero. Therefore, if a predator is detected, Miguel freezes, regardless of whether there is prey or mate visible. Furthermore, if Miguel sees both prey and mate, but no predator, it will move toward the prey, since the hunger increases faster than the sex-drive. If Miguel has not "eaten" for a long time, then its hunger level will increase beyond the static level that fear is set at when a predator is visible. In this case, if both predator and prey are visible, the robot will move toward the prey even though there is a predator in sight. This also is true when the robot has not mated for a long time.

The following is a description of one particular execution sequence, shown in Figure 14. When the robot was started, the predator, prey, and mate were all in view. In the first picture, Miguel is remaining motionless, because there is a green box representing a predator in his view just out of the picture to the right. Then, the predator was removed from sight, and Miguel began

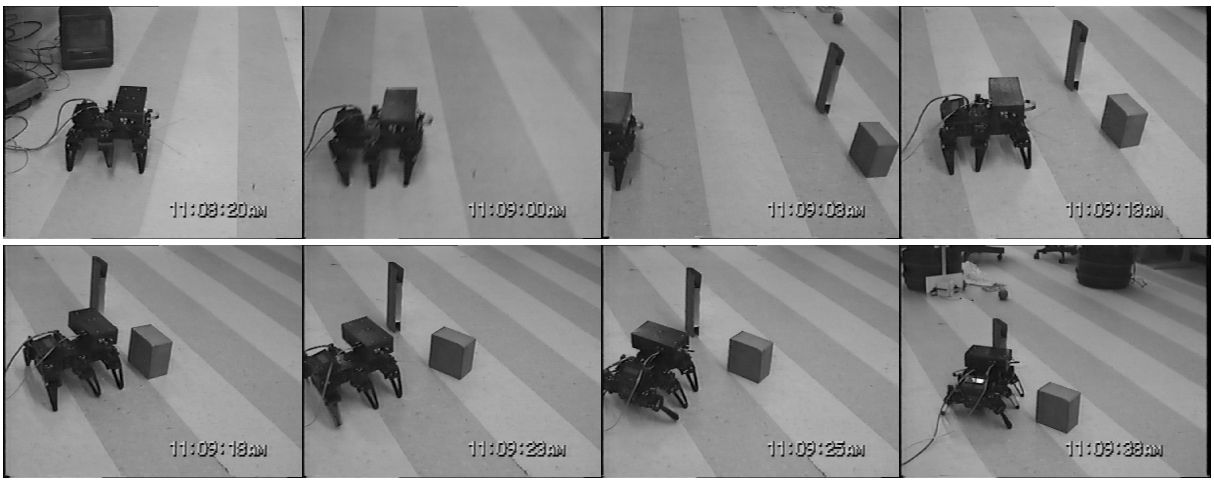

Fig. 14. This sequence of pictures depict Miguel's actions during one execution. The sequence proceeds from left to right. At first, Miguel is stationary because there is a box representing a predator just out of the picture to the right. After the predator is removed, Miguel then moves towards the orange box in the foreground, which represents prey. Once it has contacted the prey, it moves toward the purple box in the background, which represents a mate. More details of this execution sequence are given in the text.

to walk towards the prey and mate, as seen in the second picture. In the third and fourth pictures, as Miguel gets closer to the two stimuli, we can see that it is heading for the prey, which is represented by the orange box in the foreground. After contacting the prey with one of its whiskers in the fifth picture, the obstacle-avoidance behavior took over, and the robot backed up, as shown in the sixth picture. This put both the prey and mate in view once again. Since the hunger level had been reset after contacting the prey, Miguel turned toward the mate in the seventh picture, and moved to contact the mate, as shown in the last picture. After the mate was contacted, the predator was placed in view again. Miguel froze and the run was over. (The tethers seen in the pictures are for power, one for the robot and one for the vision board; all computation is performed in real-time on-board Miguel).

Miguel's behavior has advanced incrementally. The execution sequences depicted in Figures 15 and 16 show Miguel's behavior during earlier stages in his development. At the time the execution in Figure 15 was conducted, Miguel was using obstacle-avoidance and prey tracking behaviors. Figure 15 shows Miguel following an orange box, which represents prey, as it is moved around.

Figure 16 shows Miguel running obstacle-avoidance and the move-forward behavior, as well as responding to both prey and predator in the environment. In the first part of the sequence shown in Figure 16, Miguel follows prey, represented by the orange box, as it is moved. Then a predator, represented by a green box, is introduced into the robot's field of view. Miguel stops and remains motionless until the predator is removed. At this point, Miguel

resumes his movement toward the prey.

The execution sequence depicted in Figure 17 shows Miguel's actions when confronted by a prey, predator, mate, and hiding-place. Initially, all of these stimuli were visible, as can be seen in the first picture. The green box to the right side represents a predator. To the left of that, the tall purple box in the background and the orange box in the foreground represent a mate and prey, respectively. And at the left side of the picture, the tall orange card represents a hiding-place for the robot. When the robot is started, the fear motivational variable is greater than the hunger or sex-drive, because of the presence of a visible predator. Therefore, the robot remains motionless. In the second picture, the predator has been removed from the robot's field of view, and it is walking towards the prey. This is because hunger is now the motivational variable with the greatest value since hunger increases faster than sex-drive. In the next two pictures, the prey is moved around, and Miguel tracks this movement and continues to move toward the prey. In the fifth picture, the prey has been removed, leaving only the hiding-place and the mate. Hunger is still the motivational variable with the greatest value, but since there is no visible prey, the sex-drive, which is the next greatest variable, controls the robots actions. The robot is moving toward the mate in this picture. The mate was moved around, and Miguel continued to move towards it as it moved. The mate has been removed in the seventh picture. Since there is no stimulus related to one of the motivations, Miguel now looks for hiding-places and moves toward them. In the last picture, all stimuli have been removed, and so Miguel is simply moving forward, searching for some stimuli to react to.

Several other trials have been conducted with Miguel. In some of these, both predator and prey have been placed in view, causing the robot to stop moving, and then left in view long enough that the hunger level becomes greater than the fear level. In this case, the robot starts moving again toward the prey. The same test has been conducted using a predator and mate, with similar results. If there is no stimulus in the environment, then the robot moves forward.

If the robot contacts an obstacle it will back up, turn a little, walk forward a few steps, and then respond to the present view in the environment. Since the robot does not retain any previous locations of stimuli in memory, this can cause the robot to abandon any prey or mate that it had previously been trying to acquire. For instance, if the robot is moving toward prey and contacts an obstacle, then after it backs up, turns, and moves forward, the prey may not be visible anymore. In this case, the robot would abandon its attempt to acquire that prey.

Fig. 15. This sequence of pictures shows Miguel following a prey as it is moved around. The sequence proceeds from left to right. The prey is represented by an orange box. The video monitor shows the broadcast results of Miguel's visual processing. Note how in the bottom two photographs the region size increases as Miguel approaches the prey object.

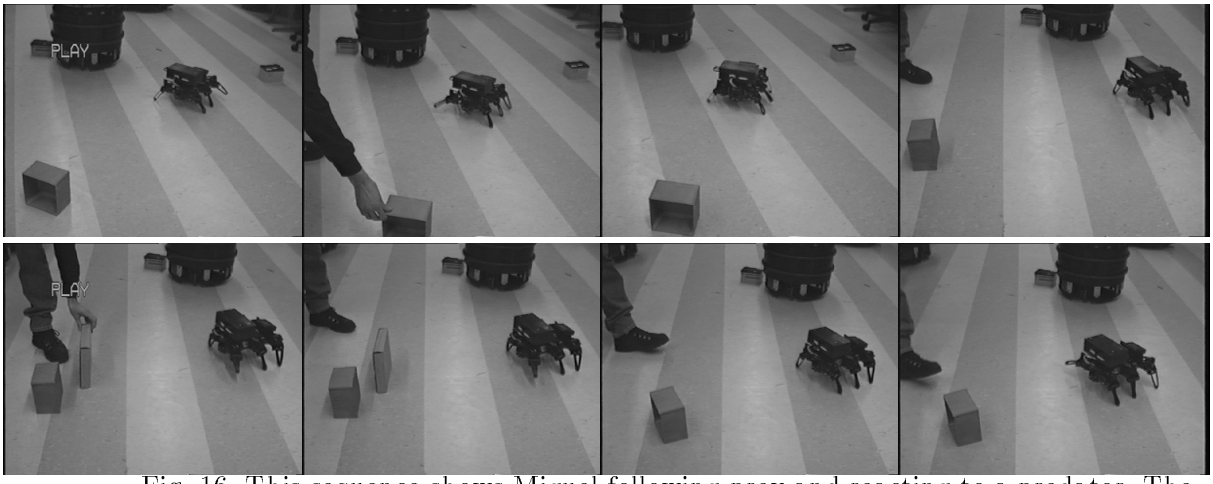

Fig. 16. This sequence shows Miguel following prey and reacting to a predator. The sequence proceeds from left to right. In the first four pictures Miguel is moving towards the prey as it is moved around. The prey is represented by an orange box. In the fifth picture, a predator is placed alongside the prey in Miguel's view. A tall green box represents the predator. When Miguel sees the predator, it stops moving, and in the sixth picture, we can see that Miguel has remained motionless. In the last two pictures, the predator has been removed from Miguel's view, and the robot resumes moving toward the prey.

7 Summary and Conclusions

This article shows how scientists from a range of disciplines can work together using schema-theoretic methods as an interlingua. In particular, agent-

Fig. 17. This sequence of pictures depicts Miguel's reactions to many different stimuli, including when they are all simultaneously visible. The sequence proceeds from left to right. At first, all the stimuli are visible, and Miguel remains motionless because of the presence of the predator. Once the predator is removed, Miguel is moving towards the prey, both when it is stationary, and when it is moving. The prey is then removed, and Miguel tracks the mate. After the mate is removed, Miguel is moving towards a hiding-place. And, in the last picture, Miguel is just moving-forward since all stimuli have been removed from view.

environment interactions of the praying mantis have been used as a model to show how ethological studies can lead to robotic implementations. The studies themselves can assist in providing a basis for determining the means by which robot's can discover their own ecological niche within the world.

We have presented an implementation of an ethological model of a praying mantis on a robotic hexapod which incorporates visually guided motivated behaviors such as prey acquisition, mating, and predator avoidance. These were further integrated with obstacle-avoidance and chantlitaxia behaviors. These efforts demonstrate the feasibility of importing models from the biological community into robotics and show that species-specific activities can lead to interesting robotic performance. It is hoped that these results will engender future research within the biological community that will lead to iterative refinement of models such as the one presented here.

References

[1] Arbib, M., "Perceptual Structures and Distributed Motor Control", *Handbook of Physiology – The Nervous System II*, ed. Brooks, pp. 1449-1465, 1981.

[2] Arbib, M.A., "Schema Theory", In the *Encyclopedia of Artificial Intelligence*, 2nd Edition, edited by Stuart Shapiro, 2:1427-1443, Wiley, 1992.

[3] Arkin, R.C., "Neuroscience in motion: The Application of Schema Theory

to Mobile Robotics", in *Visuomotor Coordination: Amphibians, Comparisons, Models, and Robots*, eds. J.-P. Ewert and M. Arbib. New York: Plenum Press, 1989, pp. 649-672.

[4] Arkin, R.C., "Motor Schema-Based Mobile Robot Navigation", *International Journal of Robotics Research*, Vol. 8, No. 4, August 1989, pp. 92-112.

[5] Arkin, R.C., "Integrating Behavioral, Perceptual, and World Knowledge in Reactive Navigation", *Robotics and Autonomous Systems*, 6 (1990), pp. 105-122.

[6] Arkin, R.C., "The Impact of Cybernetics on the Design of a Mobile Robot System: A Case Study", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 20, No. 6, Nov/Dec 1990, pp. 1245-1257.

[7] Arkin, R.C., "Cooperation without Communication: Multi-agent Schema Based Robot Navigation", *Journal of Robotic Systems*, Vol. 9(3), April 1992, pp. 351-364.

[8] Arkin, R.C., "Modeling Neural Function at the Schema Level: Implications and Results for Robotic Control", chapter in *Biological Neural Networks in Invertebrate Neuroethology and Robotics*, ed. R. Beer, R. Ritzmann, and T. McKenna, Academic Press, pp. 383-410, 1993.

[9] Arkin, R.C. and Hobbs, J.D., "Dimensions of Communication and Social Organization in Multi-Agent Robotic Systems", *From animals to animats 2: Proc. 2nd International Conference on Simulation of Adaptive Behavior*, Honolulu, HI, Dec. 1992, MIT Press, pp. 486-493.

[10] Arkin, R.C. and MacKenzie, D., "Temporal Coordination of Perceptual Algorithms for Mobile Robot Navigation", *IEEE Transactions on Robotics and Automation*, Vol. 10, No. 3, June 1994.

[11] Alexander, A., Arbib, MA., and Weitzenfeld, A., 1999, "Web Simulation of Brain Model" to appear in *1999 Int'l Conf On Web-Based Modelling and Simulation*, San Francisco, CA.

[12] Balch, T. and Arkin, R.C., "Communication in Reactive Multiagent Robotic Systems", *Autonomous Robots*, Vol. 1, No. 1, Nov. 1994.

[13] Balch, T., Boone, G., Collins, T., Forbes, H., MacKenzie, D., and Santamaría, J., "Io, Ganymede, and Callisto - A Multiagent Robot Trash-collecting Team", *AI Magazine*, Vol. 16, No. 2, Summer 1995, pp. 39-51.

[14] Balderrama, N. and Maldonado, E., *Insect Physiol.*, 19:93-101, 1973.

[15] Beer, R., *Intelligence as Adaptive Behavior: An Experiment in Computational Neuroethology*, Academic Press, 1990.

[16] Brooks, R., "A Robust Layered Control System for a Mobile Robot", *IEEE Journal of Robotics and Automation*, Vol. RA-2, No. 1, 1986 pp. 14-23.

[17] Cervantes-Pérez, F., "Schema Theory as a Common Language to Study Sensori-Motor Coordination", in in *Visuomotor Coordination: Amphibians, Comparisons, Models, and Robots*, eds. J.-P. Ewert and M. Arbib. New York: Plenum Press, 1989, pp. 421-450.

[18] Cervantes-Pérez, F., and Arbib, M. A., "Stability and parameter dependency analyses of a Facilitation Tectal Column (FTC) model", *J. Math. Biol.*, 29:1-32, 1990.

[19] Cervantes-Pérez, F., Franco, A., Velazquez, S., Lara, N., 1993, A Schema Theoretic Approach to Study the "Chantitlaxia" Behavior in the Praying Mantis, *Proc. of the First Workshop on Neural Architectures and Distributed AI: From Schema Assemblages to Neural Networks*, USC, October 19-20.

[20] Cervantes-Pérez, F., Guevara, D., and Herrera, A., "Modulation of prey-catching behavior in toads: data and modeling", in: *Visual structures and integrated functions*, Eds. Arbib, M.A., and Ewert, J.P., Springer Verlag Research Notes in Neural Computing vol. 3, 1991, pp. 397-415.

[21] Cervantes-Perez, F., Herrera, A., and Garcia, M., "Modulatory Effects on Prey-Recognition in Amphibia: A Theoretical- Experimental Study", in *Neuroscience: From Neural Networks to Artificial Intelligence*, eds. Rudomin et. al., Springer Verlag, Research Notes in Neural Computing Volume 4, pp. 426-449 1993.

[22] Cervantes-Pérez, F., Lara, R., and Arbib, M.A., "A neural model of interactions subserving prey-predator discrimination and size preference in anuran amphibia", *J. Theor. Biol.*, 113:117-152, 1985.

[23] Cobas, A., and Arbib, M.A., "Prey-catching and predator avoidance 2: modeling the medullary hemifield deficit", in: *Visual structures and integrated functions*, Eds. Arbib, M.A., and Ewert, J.P., Springer Verlag Research Notes in Neural Computing vol 3, 1991, pp. 153-166.

[24] Connell, J., "A Colony Architecture for an Artificial Creature", *Technical Report 1151*, MIT A.I. Laboratory, August 1989.

[25] Corbacho, F. and Arbib, M.A., "Learning to Detour", *Adaptive behavior*, 3(4):419-468, 1995.

[26] Didday, R.L., "A Model of Visuomotor Mechanisms in the Frog Optic Tectum", *Math. Biosci.*, 30:169-180, 1976.

[27] Dominey, P. and Arbib, MA, "A Cortico-Subcortico Model for Generation of Spatially Accurate Sequential Saccades", *Cerebral Cortex*, 2:152-175, 1047-3211/92 Mar/Apr 1992.

[28] Duchon, Warren, and Kaelbling, L., "Ecological Robotics: Controlling Behavior with Optic Flow", *Proc. 17th Annual Conference of the Cognitive Science Society*, 1995.

[29] Ewert, J.P., *Neuroethology: an introduction to the neurophysiological fundamentals of behavior*, Ewert, J.P., Springer, Verlag, 1980.

[30] Ewert, J.P., "Tectal mechanisms that underlie prey-catching and avoidance behaviors in toads", in: *Comparative Neurology of the optic tectum.* Ed. Vanegas, H., Plenum Press, 1984, pp. 247-416.

[31] Ewert, J.P., "Neuroethology of releasing mechanisms: prey-catching in toads", *Behav. Brain. Sci.*, 10:337-405, 1987.

[32] Ewert, J.P., "The release of visual behavior in toads: stages of parallel/hierarchical information processing", in: *Visuomotor coordination: amphibians, comparisons, and robots.* Eds. Ewert, J.P., and Arbib, M.A., Plenum Press, 1989, pp. 39-120.

[33] Fagg A.H., King I.K., Lewis M.A., Liaw J.-S., Weitzenfeld A., "A Neural Network Based Testbed for Modelling Sensorimotor Integration in Robotic Applications", *Proc. of IJCNN92*, Baltimore, MD, 1992.

[34] Franco, A., "Estudio teórico-experimental de la conducta de Busqueda de Refugio o Chantlitaxia de la mantis religiosa Stagmomantis limbata hahn", Masters dissertation, UACPyP del CCH National University of Mexico, April, 1990.

[35] Gibson, J.J., "The Theory of Affordances", *Perceiving, Acting, and Knowing*, ed. Shaw, R., and Bransford, J., Erlbaum, 1977.

[36] Grand, S., Cliff, D., Malhotra, A., "Creatures: Artificial Life Autonomous Software Agents for Home Entertainment", *Proc. Autonomous Agents '97*, Marina Del Rey, CA, Feb., 1997.

[37] Grosser, O. and Grosser-Cornhels, U., "Neurophysiology of the anuran visual system", in: *Frog Neurobiology*, Eds. Llinas, R., and Precht, W., Springer Verlag, 1976, pp. 298-385.

[38] Ingle, D., "Spatial vision in anurans", in *The amphibians visual system: a multidisciplinary approach*, Ed. Fite, K.V., Academic Press, 1976, pp. 119-140.

[39] Langton, C., (ed.), *Artificial Life: An Overview*, MIT Press, Cambridge, MA, 1995.

[40] Lara, R., Carmona, M., Daza, f., and Cruz, A., "A global model of the neural mechanisms responsible for visuomotor coordination in toads", *J. Theor. Biol.*, 110:587-618, 1984.

[41] Liaw, J., and Arbib, M.A., "Neural mechanisms underlying direction-sensitive avoidance behavior", *Adaptive Behavior*, 1:227-261, 1993.

[42] Lyons, D. and Arbib, M., "A Formal Model of Computation for Sensory-based Robotics", *IEEE Trans. on Robotics and Auto.*, Vol. 6, No. 3, June 1989, pp. 280-293.

[43] MacKenzie, D., "A Design Methodology for the Configuration of Behavior-based Mobile Robots", *Ph.D. Dissertation*, College of Computing, Georgia Tech, Atlanta, GA, Fall 1996.

[44] MacKenzie, D., Arkin, R.C., and Cameron, R., "Multiagent Mission Specification and Execution", *Autonomous Robots*, Vol. 4, No. 1, Jan. 1997, pp. 29-52.

[45] Maes, P., "The Dynamics of Action Selection" *Proc. Eleventh Intern. Joint Conf. on Artificial Intelligence (IJCAI-89)*, Detroit, MI, pp. 991-997, 1989.

[46] Maldonado, E., *Z. Vergl. Physiol.*, 9:435-445, 1970.

[47] Mataric, M., "Navigating with a Rat Brain: A Neurobiologically-Inspired Model for Robot Spatial Representation", *Proc. 1st Int. Conf. on Simulation of Adaptive Behavior*, 1990, pp. 169-75.

[48] McFarland, D. and Bosser, T., *Intelligent Behavior in Animals and Robots*, MIT Press, 1993.

[49] McFarland, D., "Towards Robot Cooperation", *From Animals to Animats 3*, ed, D. Cliff et al, 1994, pp. 440-451.

[50] Miller, D., "Experiences looking into Niches", *Working Notes, 1995 AAAI Spring Symposium: Lessons Learned from Implemented Software Architectures for Physical Agents*, Palo Alto, CA, March 1995, pp. 141-145.

[51] Neisser, U., *Cognition and Reality: Principles and Implications of Cognitive Psychology*, Freeman, 1976.

[52] Steels, L., "A Case Study in the behavior-oriented design of Autonomous Agents", *From Animals to Animats 3*, ed, D. Cliff et al, 1994, pp. 445-452.

[53] Weitzenfeld, A., "ASL: Hierarchy, Composition, Heterogeneity, and Multi-Granularity in Concurrent Object-Oriented Programming", *Proceeding of the Workshop on Neural Architectures and Distributed AI: From Schema Assemblages to Neural Networks*, Center for Neural Engineering, USC, Los Angeles, CA, Oct. 1993.

[54] Weitzenfeld A., "NSL - Neural Simulation Language", in *The Handbook of Brain Theory and Neural Networks*, Ed¿ M. Arbib, MIT Press, pp. 654-658, 1995.

[55] Weitzenfeld, A., Arbib, M., 1991, "A Concurrent Object-Oriented Framework for the Simulation of Neural Networks", *Proceedings of ECOOP/OOPSLA '90 Workshop on Object-Based Concurrent Programming*, OOPS Messenger, 2(2):120-124, April.

[56] Weitzenfeld, A., and Arbib, M.A., "NSL - Neural Simulation Language, in Neural Networks Simulation Environments", Editor J. Skrzypek, Kluwer, pp. 73-93, 1994.

[57] Weitzenfeld A, Arbib MA and Alexander A, *The Neural Simulation Language NSL: System and Application*, MIT Press (in preparation).