Weitzenfeld, A., Journal Robotics and Autonomous Systems, Elsevier Vol 56, No. 2 pp 177-197, Feb 200, ISSN 0921-8890

From Schemas to Neural Networks: A Multi-level Modeling Approach to Biologically-Inspired Autonomous Robotic Systems

Alfredo Weitzenfeld

Professor
Computer Engineering Department
Mexico Autonomous Institute of Technology (ITAM)
Rio Hondo #1, San Angel Tizapan, CP 01000, Mexico DF, MEXICO
tel: +52-55-56284060
fax: +52-55-56284065
email: alfredo@itam.mx

Weitzenfeld, A., Journal Robotics and Autonomous Systems, Elsevier Vol 56, No. 2 pp 177-197, Feb 200, ISSN 0921-8890

From Schemas to Neural Networks: A Multi-level Modeling Approach to Biologically-Inspired Autonomous Robotic Systems[1]

Alfredo Weitzenfeld

Abstract

Biology has been an important source of inspiration in building adaptive autonomous robotic systems. Due to the inherent complexity of these models, most biologically-inspired robotic systems tend to be ethological without linkage to underlying neural circuitry. Yet, neural mechanisms are crucial in modeling adaptation and learning. The work presented in this paper describes a schema and neural network multi-level modeling approach to biologically inspired autonomous robotic systems. A prey acquisition model with detour behavior in frogs is presented to exemplify the modeling approach. The model is tested with simulated and physical robots using the ASL/NSL and MIRO robotic system.

Key words : biological-inspired robotics, multi-level schemas, biological neural networks, prey acquisition with detour.

1. INTRODUCTION

Biology has provided throughout the years an important source of inspiration in the construction of robotic systems. Living organisms are studied through cycles of experimentation and theoretical modeling to better understand their underlying mechanisms, both structural and functional. Experimentation, in the form of data gathering (ethological, physiological and anatomical), feed theoretical models that generate predictions to be validated by simulation and robotic experimentation. This cycle of biological experimentation, theoretical modeling, robotic simulation and experimentation is depicted in Figure 1.

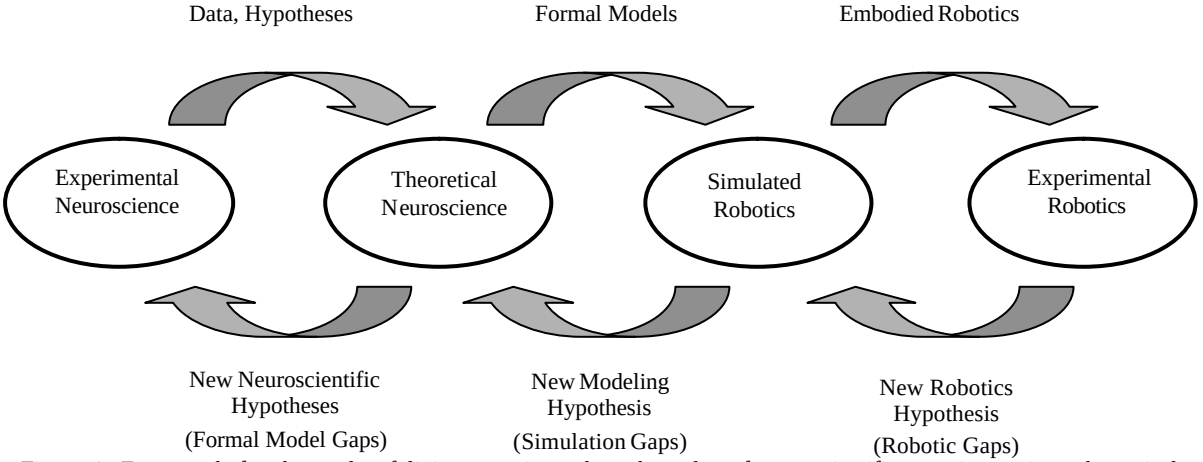


Figure 1. Framework for the study of living organisms through cycles of neuroscientific experimentation, theoretical modeling, simulation and physical robotics experimentation. Data and hypothesis obtained from neuroscientific experimentation are the basis for theoretical modeling. Resulting formal models are simulated in virtual environments. Once simulated these models can then be tested under real robotic environments. The cycle provides a general methodology where new hypothesis can be proposed in response to knowledge gaps.

[1] This work has been funded in part by CONACYT grant 42440, UC MEXUS-CONACYT and LAFMI-CONACYT collaborative projects, and the "Asociación Mexicana de Cultura, A.C.".

Many living organisms have been studied as inspiration to robotic systems, such as frogs [1], praying mantis [7, 15], cockroaches [12], hoverflies [17], and crickets [52], among others. In most cases, biological-inspired robotic systems tend to be *ethological* intended to imitate animal behavior or mechanics without very limited or no linkage to corresponding underlying neural structure (see Arkin [6] for a review of behavior-based robotics and Webb [51] for a discussion on robotics as modeling platform to animal behavior). In many of those architectures the goal is to reproduce animal behavior without restricting their underlying implementation. Yet, some aspects such as adaptation and learning can be further understood by *neuroethological* studies with behavior and mechanics linked to underlying neural circuitry. To achieve these goals it is not only necessary to develop theoretical neuroscientific models based on animal studies but also to have appropriate methodologies and supporting tools for simulation and robotic experimentation. In this paper we present a multi-level computational model and a real-time robotic architecture where neurothological systems can be developed, simulated and experimented with real robots in the way towards a better understanding of living organisms and the inspiration of adaptive autonomous robotics systems.

The paper is structured as follows: Section 2 presents a Schema and Neural Network Multi-Level Computational Model; Section 3 presents an exemplary Prey Acquisition with Detour Model; Section 4 describes the ASL/NSL and MIRO Robotic System; Section 5 presents Simulation and Experimentation Results; Section 6 presents a discussion on Closing the Research Loop and Section 7 presents final Conclusions and Discussion.

2. A SCHEMA AND NEURAL NETWORK MULTI-LEVEL COMPUTATIONAL MODEL

To address the underlying complexity in building neuroethological models and corresponding robotics systems we distinguish between two different levels of modeling: behavior and neural structure [2, 57]:

1. At the behavior level, ethological and neuroethological data from living animals is gathered to study the relationship between living entities and their environment, giving emphasis to aspects such as cooperation and competition among animals. Examples of behavioral models include praying mantis *Chantlitaxia* ("search for a proper habitat") [8], and frog and toad prey acquisition and predator avoidance models [18]. The schema computational model describes behavior in terms of *perceptual*, *sensorimotor* and *motor schemas* [3] decomposed and refined in a recursive fashion in such a way that complex behaviors can be described in terms of simpler ones.
2. At the neural level, neuroanatomical and neuronphysiological data are used to generate perceptual and motor neural network models corresponding to schemas developed at the behavioral level [30]. These models are intended to explain the underlying mechanisms for sensorimotor integration in, for example, visually guided animals [32, 58]. Examples of neural network models include tectum and pretectum/thalamus responsible for discrimination among preys and predators in toads and frogs [14]. These neural models are involved in prey acquisition and predator avoidance behaviors [16].

2.1 Behavior

In general, animal behavior may be specified in a number of ways including stimulus-response diagrams, state machines, etc. For example, in Figure 2 we show a stimulus-response diagram corresponding (a) to a frog prey acquisition behavior and (b) to a frog prey acquisition with

detour behavior where frog and prey (worm) are interposed by a fencepost barrier [20]. It should be noted that although having two eyes, the frog relies primarily on its monocular lateral visual field to approach a prey. Only when close to the prey will the frog use its binocular visual field for orientation, fixation and snapping [38]. In the detour experiment (to be further described in Section 3), the frog will also use its lateral visual field to detour around the barrier in reaching the prey. Note that in both experiments the prey must be moving in order for the frog to detect it [31].

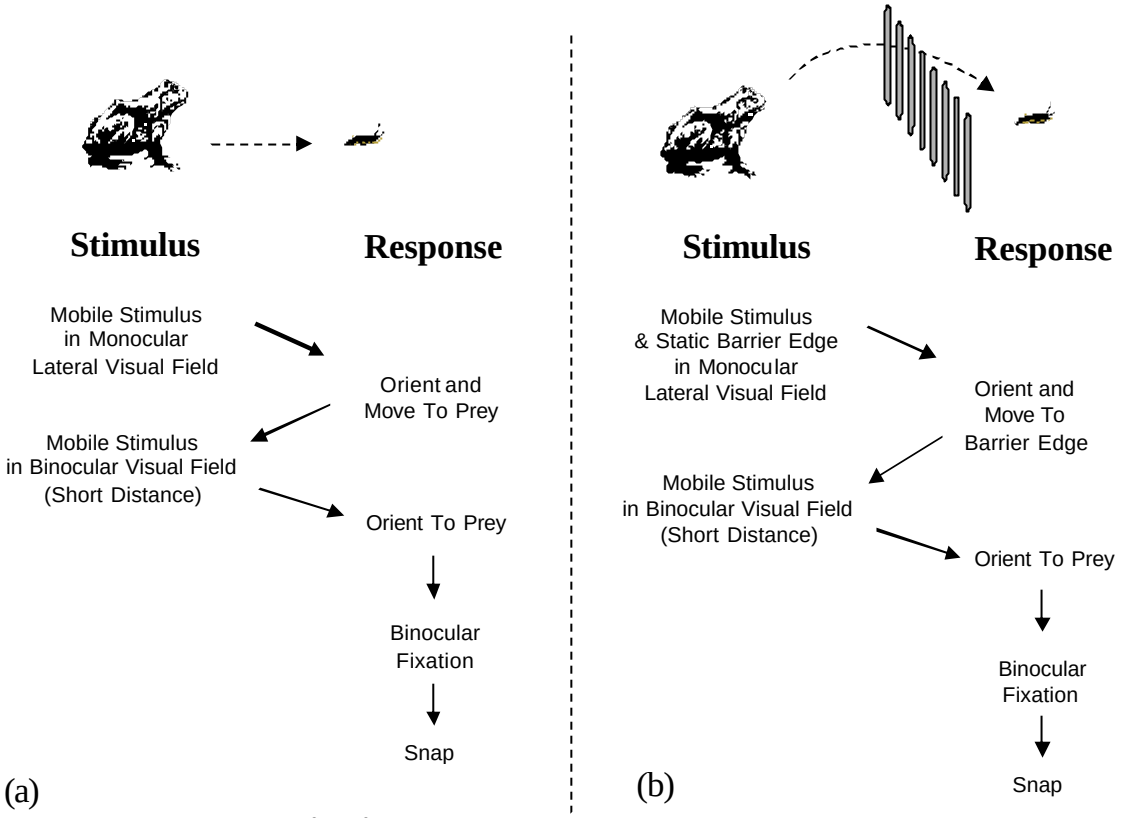


Figure 2. Stimulus-response diagrams for a frog and worm (a) prey acquisition behavior and (b) detouring around barrier in reaching stimulus.

In Figure 3 we show a state machine diagram corresponding to the stimulus-response diagram previously shown in Figure 2.

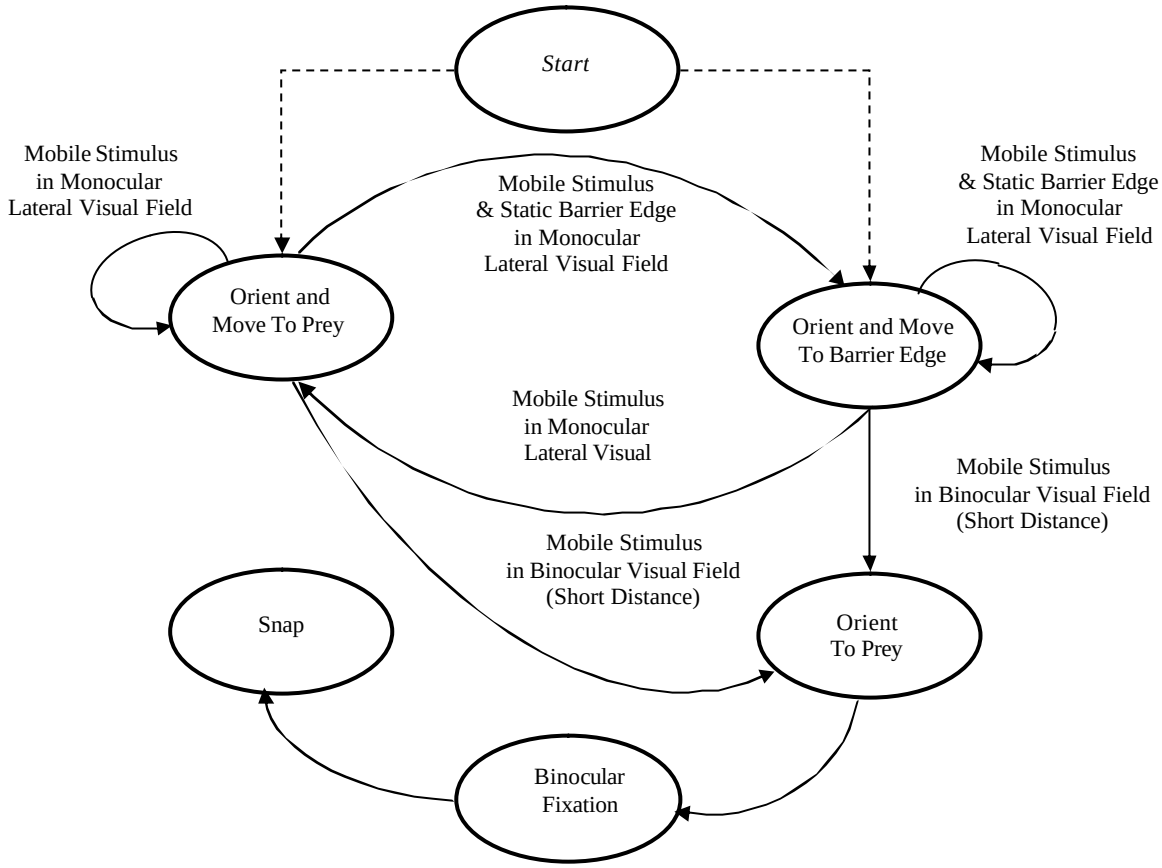


Figure 3. State machine diagram representing frog prey acquisition and prey acquisition with detour corresponding to diagram shown in Figure 2. Depending on the scenario, behavior will initiate in "Orient and Move To Prey" or "Orient and Move to Barrier Edge".

2.2 Schemas

The *schema* model defines a computational hierarchy representing a distributed model for action-perception control [3, 47]. For example, in Figure 4 we show a *rana computatrix* schema-based ethogram for prey acquisition, predator avoidance and static object avoidance. Blocks in the figure represent schemas, with *PS* standing for *Perceptual Schemas* and *S* for *Sensorimotor Schemas*. In the figure, the three sensorimotor schemas, prey acquisition, predator avoidance and static object avoidance, are activated by different perceptual schemas, a moving object, either prey or predator, or a static object, respectively. Communication between schemas can be of a cooperative or competitive nature resulting in schema weighing or *asserting*. Schema assertion takes place when schema activity surpasses certain threshold producing output indicating enough confidence on that particular schema.

Lines in Figure 4 denote direction of data transmission, with arrows indicating positive input and solid circles indicating negative feedback. In this particular ethogram, a frog will distinguish among two kinds of visual input, moving and static objects where the presence of a moving object will inhibit static object recognition, while in the case of moving objects, there exists mutual inhibition between prey and predator.

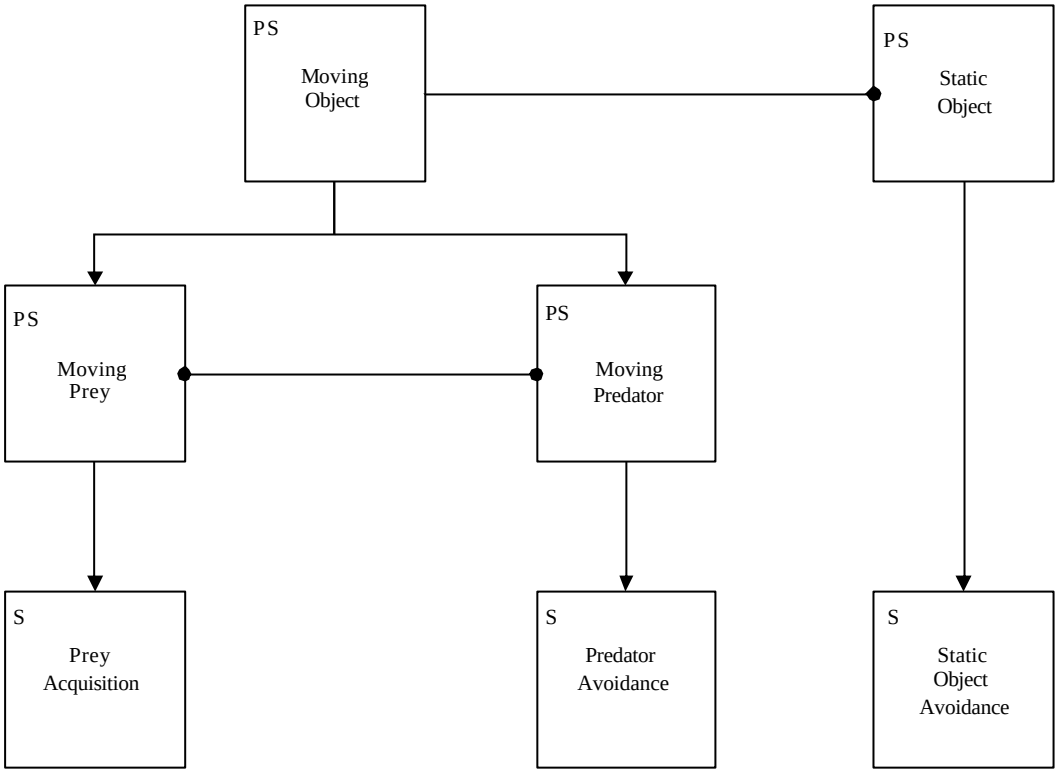


Figure 4. *Rana Computatrix* ethogram showing four perceptual schemas (PS): Moving Object, Static Object, Moving Prey and Moving Predator; and three sensorimotor schemas (S): Prey Acquisition, Predator Avoidance and Static Object Avoidance schemas. Arrows represent information flow from one schema to the next one, while solid circles at the end of lines represent negative feedback.

Schemas may be decomposed into more refined ones the same way as complex behaviors may be split into more specialized ones. Thus, the schema computational model can be seen as a hierarchical or multi-level computational model, as shown in Figure 5. In the top portion of the diagram, a higher-level schema is decomposed into two lower level schemas where the three schemas together form what is known as a schema *aggregate* or *assemblage*.

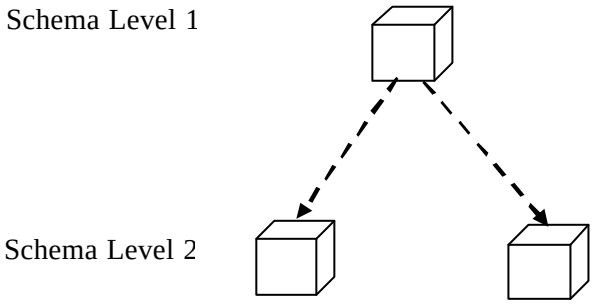


Figure 5. The schema computational model is based on hierarchical interconnected schemas. A schema at a higher level (level 1) can be decomposed (dashed lines) into additional subschemas (level 2).

For example, the Prey Acquisition schema shown in Figure 4 may be decomposed into a Prey Approach and Prey Snap *subschemas* as shown in Figure 6. Note the use of dashed arrows to represent such decomposition. The two *lower level* are mutually inhibited corresponding to the fact that in this particular example both schemas cannot be active at the same time; prey

approach is active at a far away distance while prey snap is active only when the frog is close enough for its tongue to be able to reach the prey.

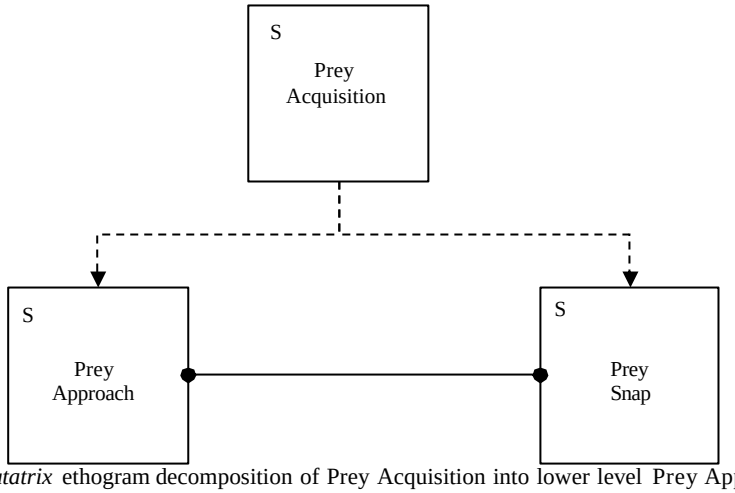


Figure 6. *Rana Computatrix* ethogram decomposition of Prey Acquisition into lower level Prey Approach and Prey Snap schemas. The diagram shows mutual inhibition between Prey Approach and Prey Snap schemas. The two schemas cannot be active at the same time since Prey Approach is instantiated at a far away distance from the prey while Prey Snap is active when close enough for the frog to snap at the prey with its tongue.

The hierarchical ethogram for the frog's prey acquisition, predator avoidance and static object avoidance schemas previously shown separately in Figures 4 and 6 can be represented by a single two-level diagram as shown in Figure 7.

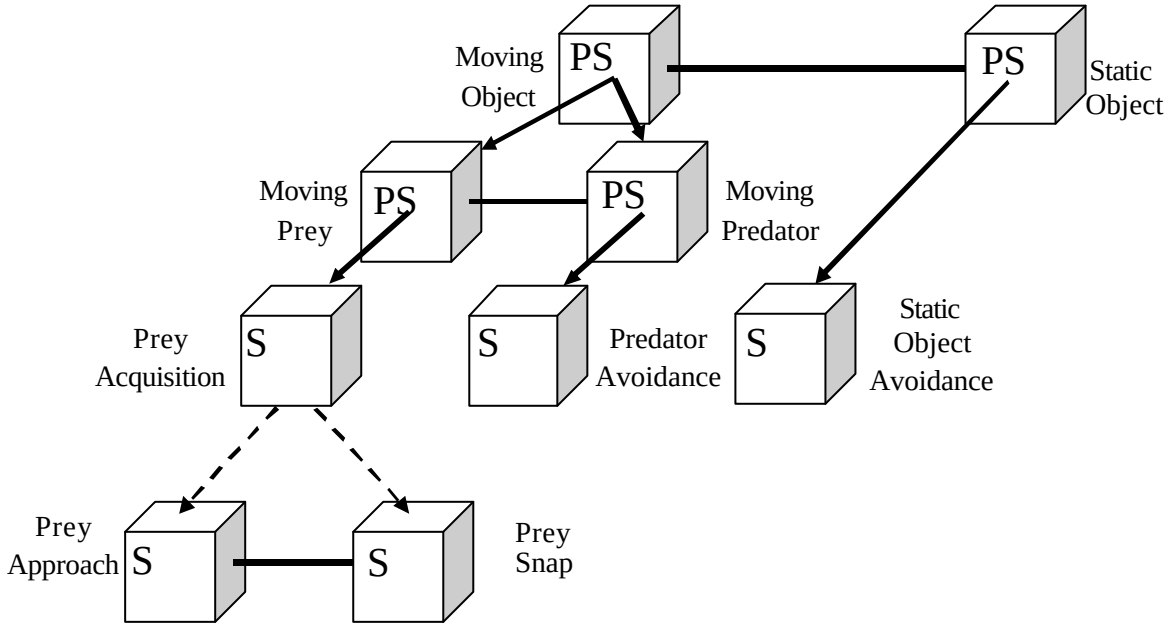


Figure 7. A multi-level schema diagram representing a frog prey acquisition, predator avoidance and static object avoidance with dashed arrows showing a Prey Acquisition assemblage linking to Prey Approach and Prey Snap lower level schemas. The figure integrates Figure 5 and 6 into a single diagram.

Note how the multi-level schema decomposition provides a top-down approach where higher level schemas are initially described at a more abstract way followed by a more detailed lower level schema specification, and a bottom-up approach where individual

schemas are specified in detail and then assembled in creating higher level comprehensive schema systems.

2.3 Neural Networks

Schemas may be designed as higher level specifications or further implemented by specialized processes. Of particular interest to neuroethological modeling are *neural schemas* mapped to biologically neural networks as shown in Figure 8.

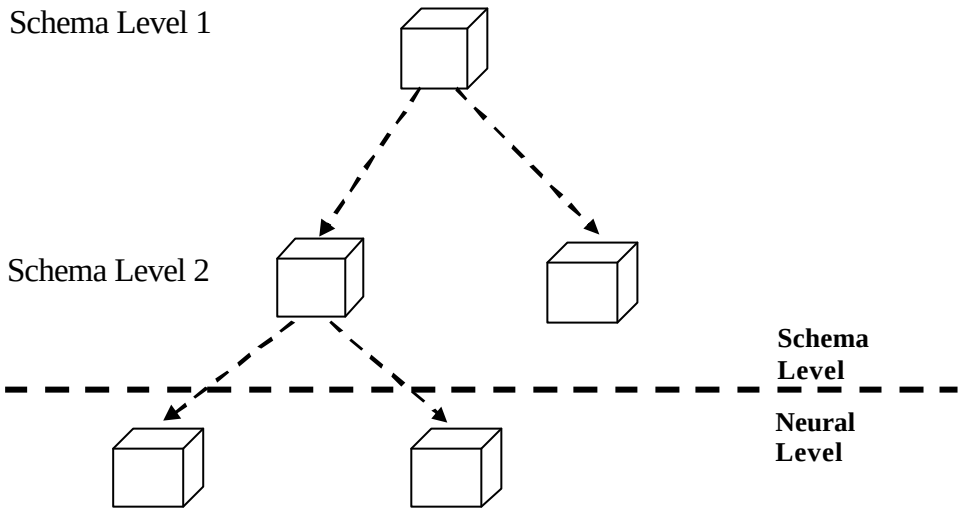


Figure 8. The schema computational model provides mechanisms for different schema implementations, where *neural schemas* are implemented by neural networks.

In general, neural schemas may be implemented by neural networks described at any levels of detail, from very simple neurons to very detailed ones [53], such as compartmental models [44], ion kinetics models [37], or those describing electrochemical mechanisms responsible for phenomena like synaptic plasticity. While there is no restriction to what degree of detail a neural schema may be described, it is usually the case where simpler neural models are used when building large neural networks. A neural network implementation of a neural schema is shown in Figure 9 where neurons are described by simpler elements, each represented by a sphere with connections among neurons represented by solid arrows.

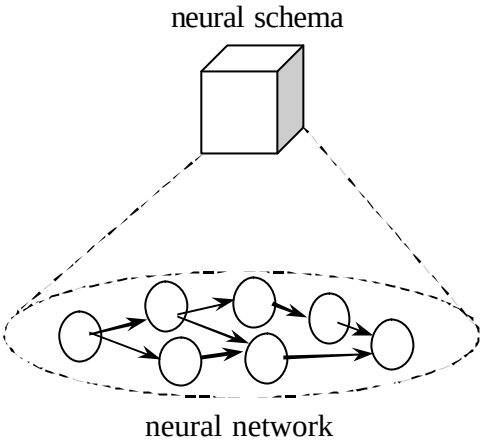


Figure 9. Neural schema implementation by a simple neural network model. Spheres represent neurons while solid arrows represent neural connections.

2.4 Adaptation and Learning

As mentioned in the introduction, an important motivation in developing biologically-inspired robotic systems is the ability of animals (and humans) to adapt and learn from experience in dealing with novel situation. Extensive studies have been done in trying to understand neural plasticity in biological systems. These studies have inspired a number of learning algorithms currently exploited in robotics such as reinforcement learning [48] as well as a large number of artificial network counterparts exemplified by backpropagation [62]. In the schema computational model learning may occur at two levels: (1) The lower level corresponds to "proximal" sensory-motor loop adaptation in neural structures, referred to in this paper as *schemas tuning*; and (2) the higher level corresponds to "distal" learning of new observed behaviors referred to in the paper as *schema learning*. By dividing the learning task at two levels the system can better manage modularity and the learning process itself by performing independent lower-level schema parameter adjusting and dynamic higher-level schema learning (see Corbacho [21][22] for additional discussions on *schema-based learning* in the context of anuran snapping and Nolfi [42] as an example of multi-level learning in the robotic domain).

In the following section we show an example of a multi-level schema computational model exemplifying the schema and neural network hierarchical architecture together with adaptation and learning processes.

3. PREY ACQUISITION WITH DETOUR MODEL

Anurans have shown during experiments advanced visuomotor coordination abilities and adaptive behavior when confronted with stationary objects on their way to a prey or when escaping from a threat. Ingle [39] and Collett [19] have observed that a frog (or toad) approaching a prey or avoiding a predator is affected by stationary objects surrounding the animal. One particular experiment related to stationary objects involves a barrier - a vertical fence - in front of a worm with the worm visible to the frog. Thus, the presence of the barrier affects the frog's trajectory to the worm, where the frog may approach directly towards the fence or detour around it. As is the case with anurans, the worm needs to be visible and moving at all times for the frog to react. Different behavioral responses to different barrier

configurations are shown in Figure 10. The actual frog experimental setup is shown in Figure 11, with the frog in front of a 20cm barrier.

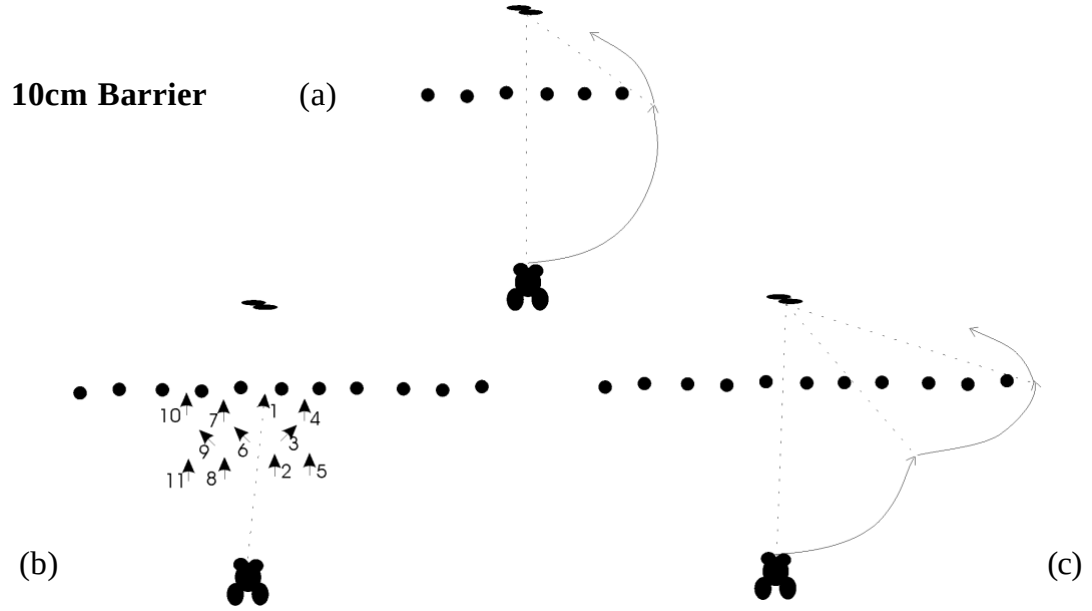


Figure 10. Three different experiments for prey acquisition with detour are described: (a) 10cm barrier where frog immediately detours in reaching the prey; (b) 20cm barrier before learning where frog bumps into barrier many times before detouring and reaching the prey (numbers indicate the succession of the movements); and (c) 20cm barrier after learning where frog after 2 or 3 trials immediately detours in reaching the prey.

We describe the following frog detour experiments [20]:

- **Experiment I**: **Barrier 10cm Wide.** A 10cm wide barrier with the toad starting from a far away distance (15-25cm) in front of the barrier with the worm 10cm behind the barrier. The experiment shows (in 95% of the trials) reliable detour behaviors from the first interaction with the 10cm barrier producing an immediate approach towards one of the edges of the barrier.
- **Experiment II**: **Barrier 20 cm wide.** A 20cm wide barrier where the "naïve" toad (a toad that has not been yet exposed to the barrier) tends to go towards a fencepost gap in the direction of the prey (this was the case for 88% of the trials). The toad initially approaches the fence trying to make its way through the gaps. During the first trials the toad goes straight towards the prey bumping into the barrier. Since the toad is not able to go through a gap it backs-up about 2cm and then reorients towards one of the neighboring gaps.
- **Experiment III**: **Barrier 20 cm wide. After learning.** A 20cm wide barrier where the "trained" toad, after 2 (43%) or 3 (57%) trials, is already detouring around the barrier without bumping into it.

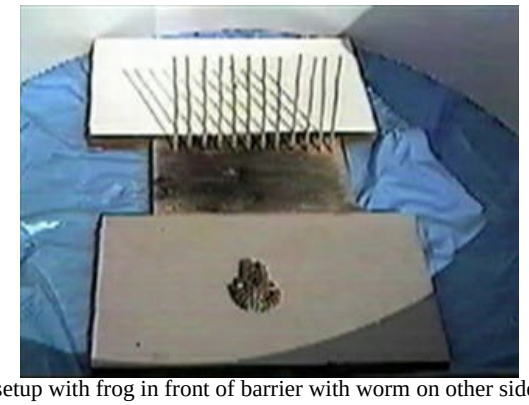


Figure 11. Image of experiment setup with frog in front of barrier with worm on other side.

3.1 Schemas

Following a top-down approach Figure 12 shows a set of schemas modeling the prey acquisition with detour behavior. Note that the predator related schemas are not shown in the diagram. In order to simplify the example and corresponding model, Max Prey Selector has been used instead of the more general Moving Prey and Prey Approach has been used instead of the more general Prey Acquisition schema.

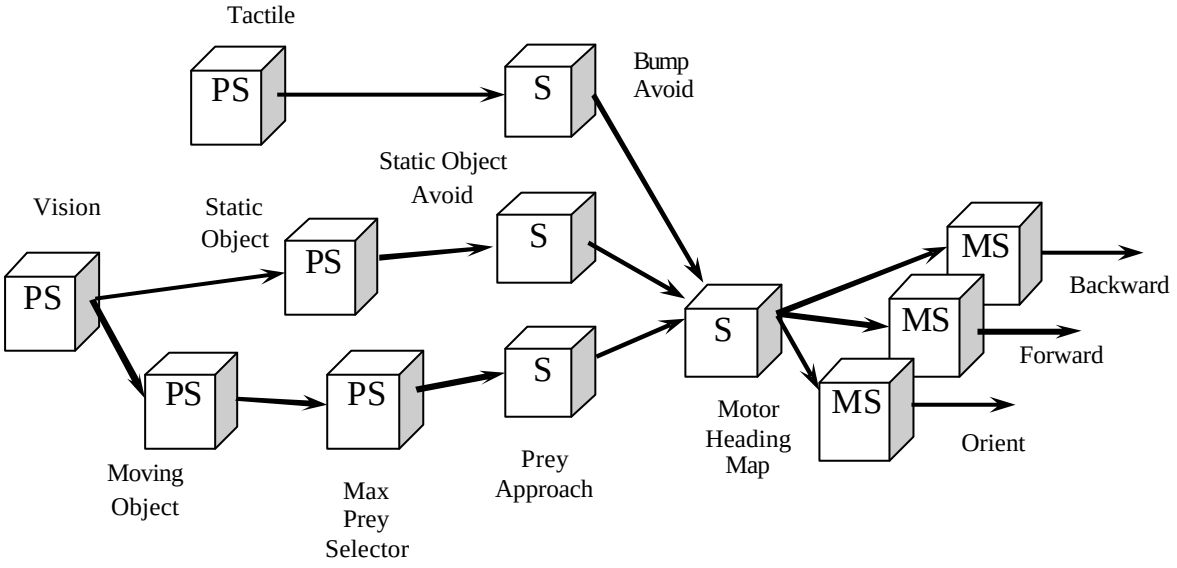


Figure 12. Frog prey acquisition with detour schema architecture consisting of Perceptual Schemas (PS): Visual, Tactile, Moving Object, Static Object, and Max Prey Selector; Sensorimotor Schemas (S): Prey Approach, Static Object Avoid, Bump Avoid, and Motor Heading Map; and Motor Schemas (MS): Orient, Forward and Backup.

The different schemas are labeled according to their type: Perceptual Schemas *PS*, Sensorimotor Schemas *S* and Motor Schemas *MS*.

Perceptual Schemas

Perceptual schemas involve sensor and recognition components:

- **Vision**. Obtains visual information from the environment filtering "interesting" visual elements such as prey and barrier in this particular experiment.
- **Tactile**. Gets triggered when the animal hits an object, such as the barrier.

- **Moving Object**. Recognizes non-stationary objects.
- **Max Prey Selector**. Selects a single prey from multiple ones present. Predators are not considered in the example.
- **Static Object**. Recognizes stationary objects.

Sensorimotor Schemas

Sensorimotor schemas integrate sensory perception and motor action:

- **Prey Approach** Gets instantiated when preys are present. It generates an attractant field whose strength decays proportional to the distance from the prey.
- **Static Object Avoid**. Gets instantiated in the presence of a static object. It generates a repellent vector field associated with each fence post whose strength decays proportional to the distance from the barrier.
- **Bump Avoid**. Triggers reorientation under bumping.
- **Motor Heading Map**. Determines the direction to jump. Projections to the Motor Heading Map (MHM) differ depending on whether a visual stimulus is identified as prey, predator or obstacle. Winner-take-all dynamics over MHM assure the selection of the strongest target angle, upon which a transformation from retinotopic to motor coordinates takes place. This is the input to the different motor schemas. Note that input fields in the motor heading map are weighted before their summation leading the frog to "foveate" towards the prey attractant field generated by PreyApproach unless there are additional fields, such as inhibitions from StaticObjectAvoid or BumpAvoid leading the frog towards a different direction other than prey. Section 5 shows graphical output from the model that further illustrates this summation and corresponding movement direction resulting from the MotorHeadingMap schema.

Motor Schemas

Detour behavior can be seen as the coordination of motor schemas. Motor schemas are implemented as functional components schematizing the intrinsic motor patterns or muscle activations. In the model, MHM contains target location with motor schema selection resulting from the competition of many maps.

- **Orient**. Obtains a direction to reorient, either forwards or backwards.
- **Backward**. Performs backward jumps.
- **Forward**. Performs forward jumps.

3.2 Neural Networks

Biologically inspired neural networks are based on physiological and anatomical neural mappings. For example, Figure 13 shows a diagram of different neural areas involved in the frog's prey acquisition, predator avoidance and static object avoidance [45].

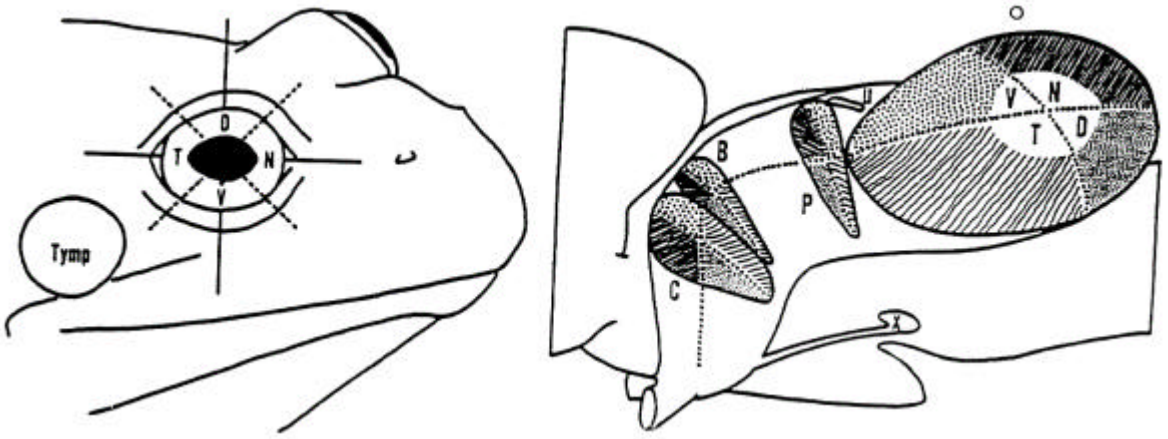


Figure 13. The two illustrations show the most important neural areas involved in the frog's prey acquisition model: Optic Tectum (O), divided into four regions: Temporal (T), Dorsal (D), Nasal (N) and Ventral (V); Thalamic Pretectal Neuropil (P); and Nucleus of Belonci (B), Lateral Geniculate Nucleus (C) and Basal Optic Root (X) [45].

In order to develop a multi-level neural schema model there must be some level of understanding of the correspondence between behavior and neural network structure. This correspondence is achieved by recording from specific neurons in the animal's brain while carrying the behavioral experiments with normal and lesioned animals [30, 39].

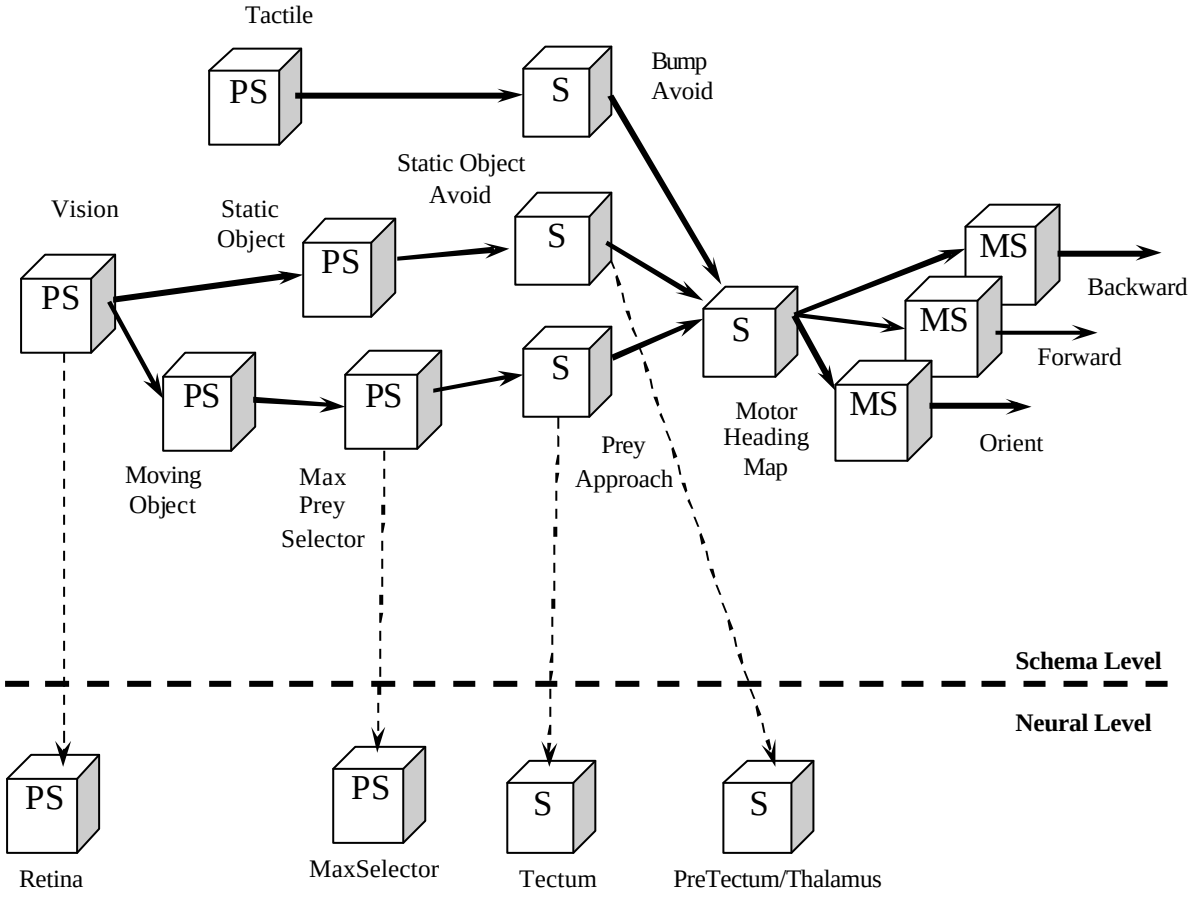


Figure 14. Frog prey acquisition with detour schema architecture with schema and neural level modules. The schema level consists of Perceptual Schemas (PS): Visual, Tactile, Moving Object, Static Object, and Max Prey Selector; Sensorimotor

Schemas (S): Prey Approach, Static Object Avoid, Bump Avoid, and Motor Heading Map; and Motor Schemas (MS): Orient, Forward, and Backup. The neural level consists of four neural schemas: Retina, MaxSelector, Tectum and PreTectum/Thalamus.

Neural Schemas

Ingle [39] offered some clues as to possible neural correlates of the various schemas. Apparently, thalamic and tectal visual mechanism can operate somewhat independently. Monocular frogs without a contralateral optic tectum can quite accurately localize barriers, and while visual input to the pretectal region of the caudal thalamus mediates barrier avoidance behavior, caudal thalamic lesions produce an inability to detour around stationary barriers set in the frog's path during pursuit of prey. In Figure 14 we show a number of neural networks implementing higher level schemas:

- **Retina**. Processes visual stimuli in anurans [49].
- **MaxSelector**. Chooses among multiple stimuli. In some cases the animal will choose an "average" stimulus location [25].
- **Tectum**. Recognizes preys in anurans [14].
- **PreTectum/Thalamus.** Recognizes stationary objects in anurans based on th10 cells [5], continuously discharging in the presence of large dark stationary objects [29].

Neural Model

Simpler neuron models, such as the *leaky integrator* neural model [2], are best suited for large-scale computation, where each neuron is defined in terms of a *membrane potential* with value *mp* representing its previous history, input *s* and output value *mf* represented by a non-linear threshold function over its membrane potential, as shown in Figure 15.

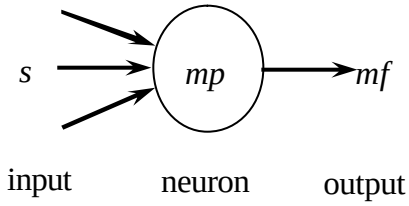


Figure 15. Simple neural model having an input *s*, a neuron body or soma *mp*, and output *mf*.

The *leaky integrator* neural model is described by:

$$d\mathbf{mp}(t)/dt = f(\mathbf{s},\mathbf{mp},t) \quad (1)$$

$$\mathbf{mf}(t) = s(\mathbf{mp}(t)) \quad (2)$$

$$t\ d\mathbf{m}(t)/dt = -m(t) + s \quad (3)$$

Equation 1 describes the membrane potential *mp* in terms of a first order differential equation function *f* that varies in time and depends on its input *s* and previous *mp* value. Equation 2 describes the cell firing *mf* in terms of a threshold function *s* that also varies in time and depends on its membrane potential. Equation 3 describes the leaky integrator model with dependence on an integration constant *t*.

For example, the Maximum Selector [25] neural network, shown in Figure 16, is described using the following leaky integrator equations:

$$t_u \frac{dup_i}{dt} = -up_i + w_u uf_i - w_m vf - h_u + s_i \quad (4)$$

$$uf_i = \begin{cases} 1 & up_i > 0 \\ 0 & up_i \le 0 \end{cases} \quad (5)$$

$$t_v \frac{dvp}{dt} = -vp + w_n \sum_{i=i}^{n} uf_i - h_v \quad (6)$$

$$vf = \begin{cases} vp & vp > 0 \\ 0 & vp \le 0 \end{cases} \quad (7)$$

Equation 4 represents the leaky integrator equation for neurons *u* with weighted input from *u* and *v*, visual input *s*, in addition to parameter *h* and time constant *t*. Equation 5 describes the output of neurons *u* as a "step" threshold function. Equation 6 represents the leaky integrator equation for neuron *v* with weighted input from *v* and *u*, in addition to parameter *h* and time constant *t*. Equation 7 describes the output of neuron *v* as a "ramp" threshold function.

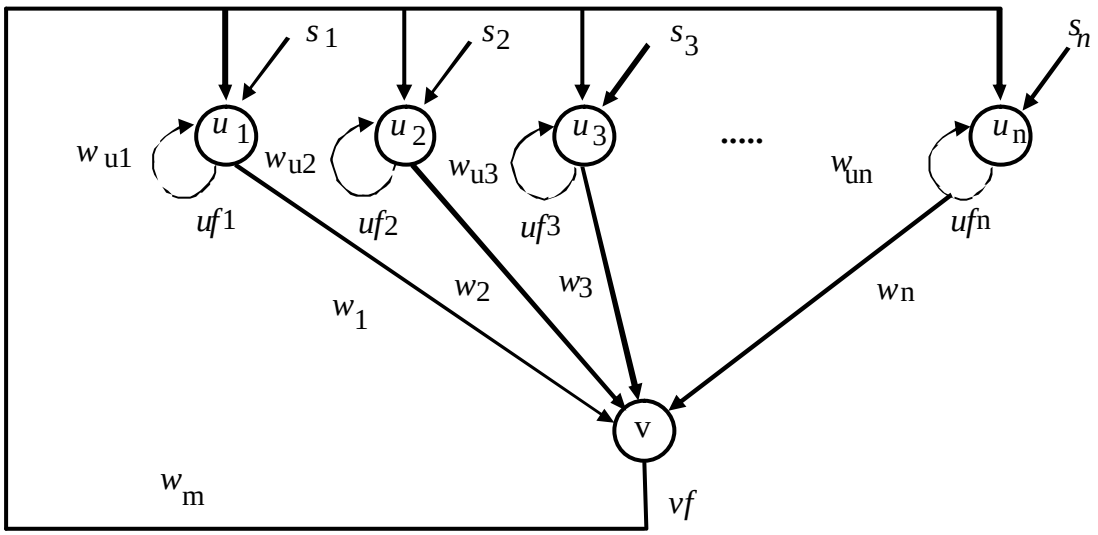


Figure 16. The neural network shown corresponds to the architecture of the Maximum Selector model, where up_i and *vp* represent neural membrane potentials, uf_i and *vf* represent neural firing rates, s_i represent inputs to the network, and w_i represent connection weights. The network is initialized with a number of positive inputs assigned to different cells. After multiple iterations the network stabilizes producing a single "winner", i.e. a single active cell.

3.3 Adaptation and Learning

The detour experiment previously described is quite exemplary in that frogs show advanced adaptation and learning capabilities after performing two or three times a basic detour task. The adaptation and learning process, summarized in Figure 17, includes two schema adaptation stages, one at the beginning as a result of bumping into the barrier, and a second adaptation at the end where the frog adjusts its sidestep magnitude in order to reach the edge of the barrier.

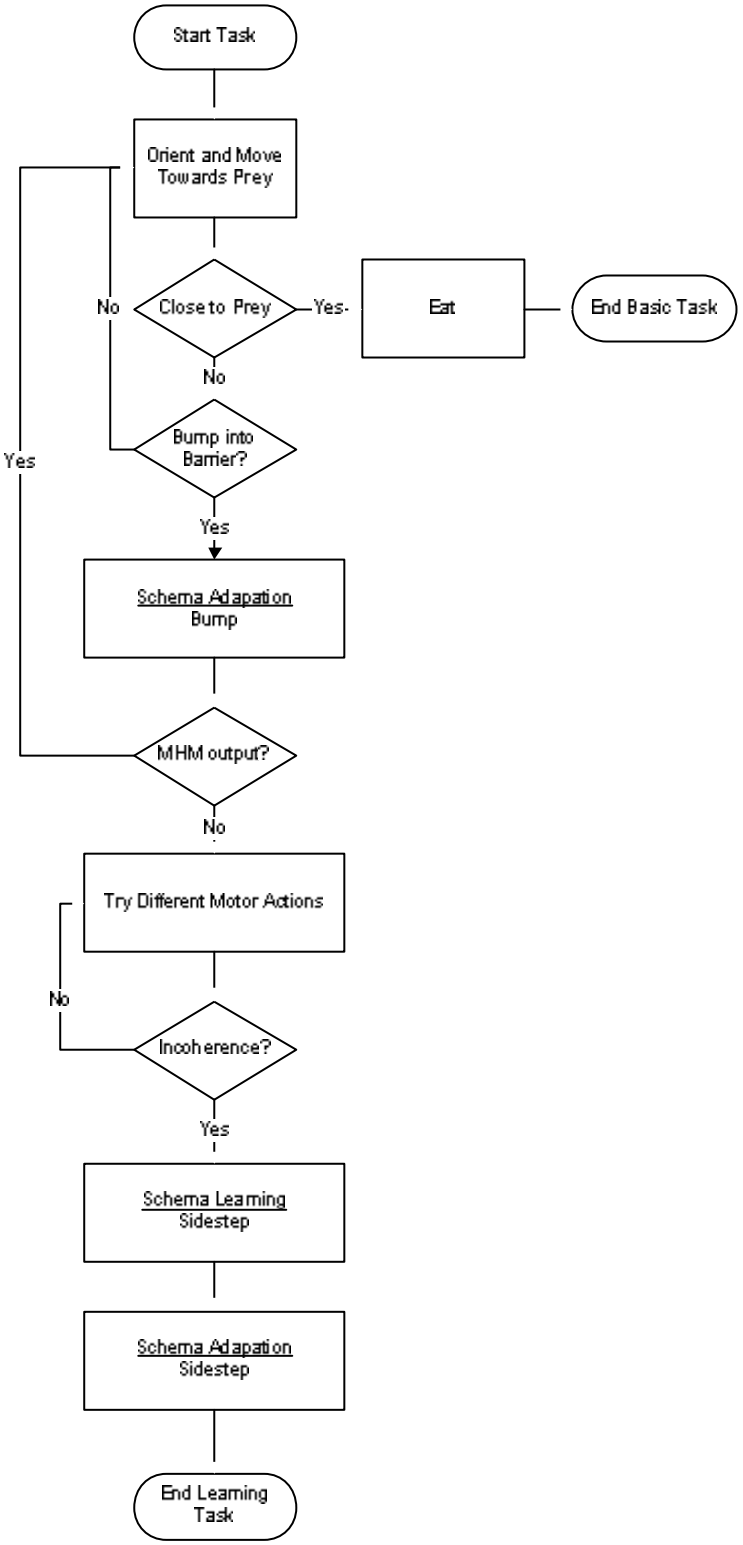
The adaptation and learning process explained in the following section is described in further detail by Corbacho and Arbib [20] and, Corbacho and Weitzenfeld [24].

Schema Adaptation(Bump)

Schema adaptation takes place in response to sensor feedback from the environment. In the frog detour experiment this adaptation is caused from different *modulatory loops* [31]. In particular, the first adaptation is caused in the frog by a tactile reflex resulting in better discrimination between passable and impassable gaps. This adaptation also enables the frog to avoid bumping into the exact same gap repeatedly. Bumping will trigger a backup response followed by a small reorientation that will lead the frog towards a neighboring gap. The schema model performs this reorientation by inhibiting the active target direction while exciting neighboring lateral gap positions. This is achieved by adjusting connections from the static object avoidance schema to the motor heading map (see figure 14). This adaptation takes a very small number of trials letting the frog also discriminate between passable versus non-passable openings with respect to body size. Experiments with frogs in the detour experiment also show that the frog in some cases will try to pass through an otherwise impassable gap by tilting its body. As long as the frog keeps on bumping into the barrier connections between static object avoidance and motor heading map schemas will be further inhibited. This process will continue until the projection of activity between the two schemas is so low that the system does not approach gaps anymore, i.e., the system has recognized the gaps as impassable. At this point, the integrated input activity from prey and static object fields is not enough to make the motor heading map reach any threshold, thus no motor actions will be taken.

Schema Learning (Sidestep)

In their work, Corbacho and Arbib [20] stipulated that the system will switch due to the lack of MHM output into an *exploratory* mode where different motor schemas will be "tried out". This will eventually lead to a combination of forward and orient motor actions, or sidestep, that will let the frog reach the edge of the barrier in its way towards the prey. As a result of the lack of motor actions from the motor heading map, the system will start testing different motor action combinations in trying to reach the prey. Among these, the orient-forward synergy motor pattern combination is the most common strategy observed and will eventually lead the frog towards the prey. After executing this synergy action a couple of times the frog will reach the edge of the barrier. In this location, the prey attraction field will be “free” from barrier inhibition thus producing output in the motor heading map towards the prey. The mismatch between a prediction of no motor heading action and the resulting motor heading output will trigger the learning process [20]. This learning process relates to the ability of animals to anticipate or predict the effect of some of its actions. It is hypothesized that this anticipation occurs by the presence of signals linking motor space back to motor heading map such that action patterns may be modified in advance [35, 27, 26]. Learning will lead to the new detour behavior described in Experiment III. Note that this behavior is taken from the repertoire of already existing behaviors, i.e., the system learns when to perform a new behavior merging from already existing individual abilities. Actually, the behavior shown in Experiment III is similar to that of Experiment I except that in the former the edge of the barrier is outside the frog’s visual field.


Start Task
Orient and Move Towards Prey
Close to Prey
No
Yes
Eat
End Basic Task
No
Bump into Barrier?
Yes
Yes
Schema Adapation
Bump
MHM output?
Yes
No
Try Different Motor Actions
No
Incoherence?
Yes
Schema Learning
Sidestep
Schema Adapation
Sidestep
End Learning Task

Schema Adaptation (Sidestep)

During exploratory behavior, the frog performs two to three short sidesteps before reaching the barrier edge. Yet, after two to three detour trials the frog is performing a single wide sidestep that takes the animal from the starting position to the edge of the barrier. Again, Corbacho and Arbib [20] contended that when the frog moves laterally to the barrier, the prey will not be centered in its visual field resulting in an *eccentricity* mismatch between prey perception and motor actions that would cause the system an additional error. This mismatch may be reduced by increasing the magnitude of the sidestep after learning.

4. THE ASL/NSL AND MIRO ROBOTIC SYSTEM

In order to describe neuroethological models using the schema and neural network multi-level computational system we have developed the Abstract Simulation Language ASL [53] and Neural Simulation Language NSL [56] architecture. We use an integrated ASL/NSL system for distributed simulation [59] with robotic experimentation carried out with the Mobile Internet Robotics MIRO [60].

4.1 ASL

The Abstract Schema Language ASL [53] defines a computational model to express the hierarchical nature of schemas where each schema incorporates its own structure and control mechanisms. The schema computational model follows a tree-like structure (schemas may also communicate between layers making the structure a directed graph). At the higher abstraction levels, schemas are only specified, leaving their detailed implementations to lower level schemas (in some cases schemas may incorporate their own local implementation). The schema interface consists of multiple unidirectional control/data, input and output ports together with a schema implementation body, as shown in Figure 18. Schemas correspond formally to *port automata* with activity variables indicating the degree of confidence [1].

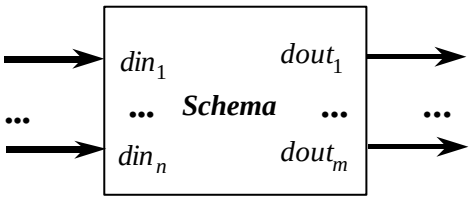


Figure 18. Each schema may contain multiple input, $din_1,...,din_n$, and output, $dout_1,...,dout_m$, ports for unidirectional communication.

Communication between schemas is in the form of asynchronous message passing, hierarchically managed, internally, through anonymous port reading and writing, and externally, through dynamic port *connections* and *relabelings*. Schemas are interconnected by matching schema interfaces, in other words, *connections* (solid arrows in Figure 14) are done by linking *output ports* from one schema to *input ports* in other schemas. On the other hand, *relabelings* (dashed arrows in Figure 14) are done by linking ports of similar type (input or output) among different schemas usually at different levels in the hierarchy. In such a multi-level hierarchy, we consider that higher-level schemas to *delegate* the task to lower-level ones, such as in the case of neural schemas. The hierarchical port management methodology enables the development of distributed architectures where schemas may be designed in a top-down

and bottom-up fashion implemented independently and without prior knowledge of the complete model or their final execution environment, encouraging component reusability.

In Figure 19 we show the ASL top-level schema architecture for the frog's prey acquisition with detour model [24]. In the diagram we only show higher level schema interconnections. Note the direction of such connections, where arrows originate at input ports and end at output ports. All *ps* ports represent weight interconnections among schemas to assert their activity.

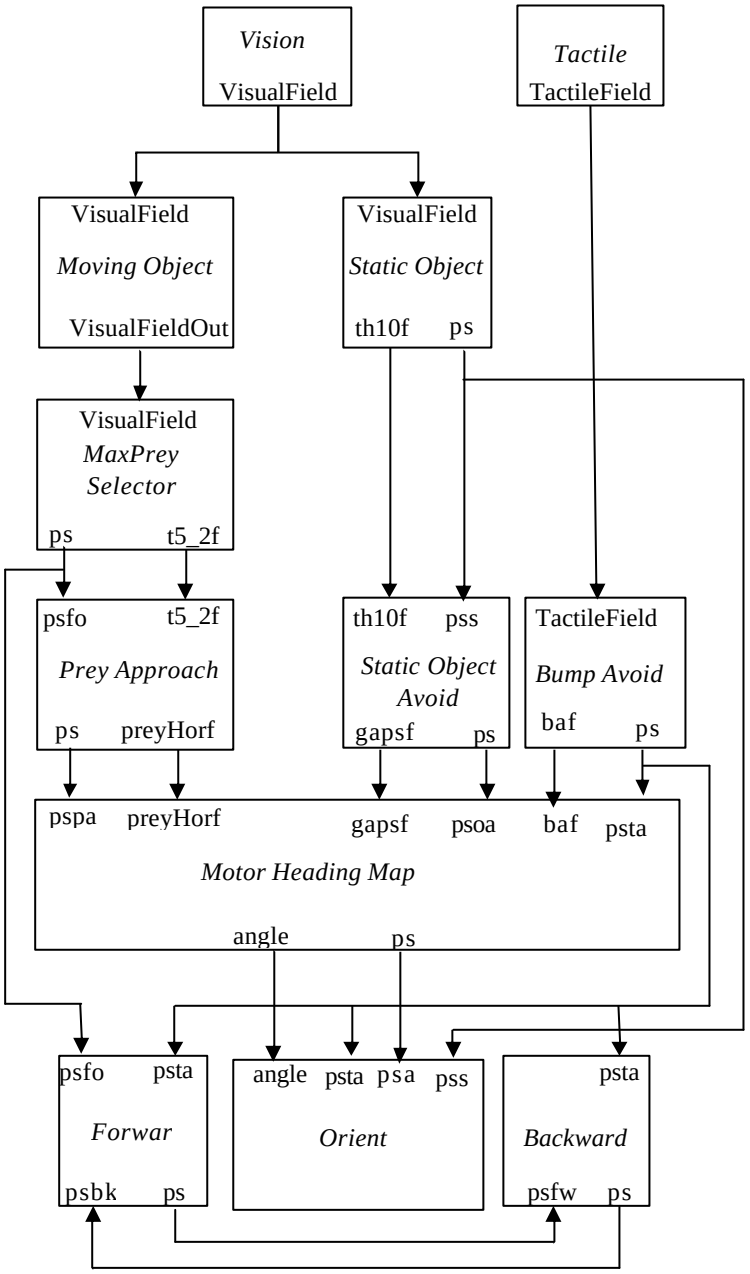


Figure 19. Schema interconnection for Frog Prey Acquisition with Detour model. Arrows connect output ports from one schema to input ports from another schema.

In the following piece of code we show portions of the schemas defining the complete model. ASL uses a Java-like syntax where schemas are declared and instantiated (not seen in the code), with a specially method *makeConn* for the specification of port connections. Connections are specified hierarchically where for example the connection between ports as in belonging to MovingObject and MaxPreySelector are specified in the *makeConn* method belonging to PreyAcquisitionDetourModel as shown below.

```
nslModule PreyAcquisitionDetourModel extends NslModule
{
        …
        public MovingObject v;
        public MaxPreySelector mp;
        public PreyApproach pa;
        …

        public void makeConn(){
                ….
                nslConnect(v.VisualFieldOut, mp.VisualField);
                nslConnect(mp.t5_2f, pa. t5_2f);
                nslConnect(mp.ps, pa.psfo);
                ….
        }
}
```

In the following piece of code we show a portion of MaxPreySelector where it *relabels* higher-level ports defined in MaxPreySelector to lower-level ports defined in the MaxSelector neural schema described in the next section. Once ports are relabeled, a higher level schema is considered to *delegate* its task to the lower level one.

```
nslModule MaxPreySelector extends NslModule
{
        public MaxSelector m;
        …

        public void makeConn(){
                nslRelabel(VisualField, m.in);
                nslRelabel(m.out, t5_2f);
        }
}
```

4.2 NSL

The Neural Simulation Language NSL [56] is integrated with the ASL system to model individual neural schemas. The MaxSelector neural schema includes instantiation variables, *sizeX* and *sizeY* as part of its declaration as shown in the next piece of code. These two variables are passed to the different bcal declarations. The schema also contains an *in* and *out* NslDinDouble2 and NslDoutDouble2 input and output ports, respectively. Additionally, there are declarations for internal variables, *up*, *uf*, *vp*, *vf*, *hu*, *hv* and *tau*, used in the *simRun* method for neural processing. Note the correspondence with equations 4 to 7. The other method, *initRun*, is used for constant and variable initializations.

```
nslModule MaxSelector (int sizeX, int sizeY) extends NslModule
{
        public NslDinDouble2 in(sizeX,sizeY);
        public NslDoutDouble2 out(sizeX,sizeY);

        public NslDouble2 up(sizeX,sizeY);
        public NslDouble2 uf(sizeX,sizeY);
        public NslDouble0 vp, vf;
        public NslDouble0 hu, hv, tau;

        public void initRun(){
                up = 0; uf = 0; vp = 0; vf = 0;
                hu = 0.1; hv = 0.5; tau = 1.0;
        }
        public void simRun(){
                up = nslDiff(up,tau, -up + uf  - vf – hu + in);
                uf = nslStep(up,0.1,0.1.0);
                vp = nslDiff(vp,tau,-vp + nslSum(uf) – hv);
                vf = nslRamp(vp);
                out = uf;
        }
}
```

A more complete model explanation can be found in [24] with the full model code available for download from the NSL web site [43].

4.3 MIRO

The ASL and NSL systems have been integrated with the MIRO [60] robotics architecture into a unified simulation and robotics experimental environment, as shown in Figure 20. MIRO provides remote sensory input to the ASL/NSL system as well as motor actions to and from the robot, respectively. MIRO performs preliminary visual processing such as blob formation and segmentation on input images, from either the simulated or real camera, while performing motor actions generating simulation or real robot commands.

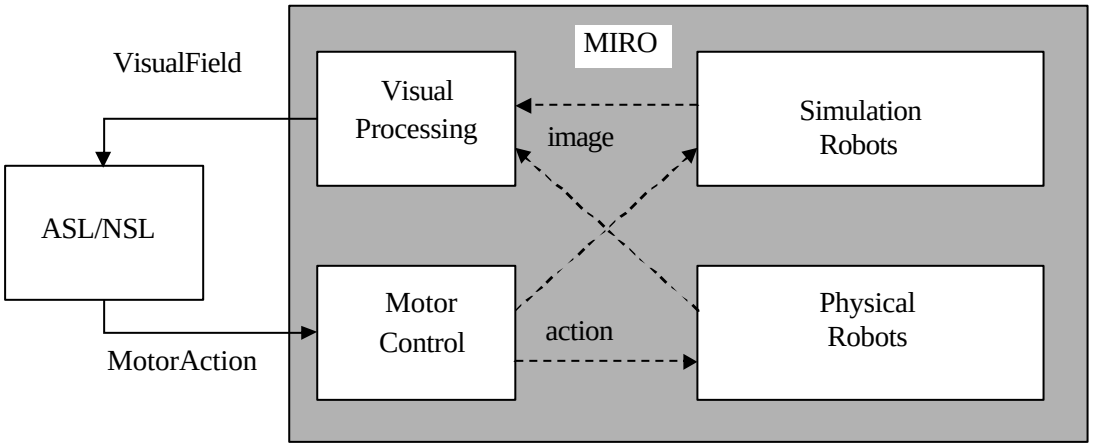


Figure 20. Block diagram for the integrated ASL/NSL and MIRO robot architecture. ASL/NSL is responsible for schema and neural network processing while MIRO provides robotic control from the simulated or physical robot environment.

The ASL/NSL and MIRO architecture performs remote ASL/NSL schema and neural network processing in a PC connected to the robot via wireless communication. Although the robot serves only as input/output device, the full system is completely autonomous and multiple robots can execute concurrently. The system supports multiple types of robots and includes Internet control and monitoring, as shown in Figure 21. The robot shown in Figure 21, Kroax, was built in our laboratory and incorporates a wireless camera transmitting analog video with a frame grabber and wireless receptor at the PC end. The robot also incorporates a transceiver for sensor reading (barrier bumping). Motor actions are controlled by a local OOPIC processor doing limited local computation.

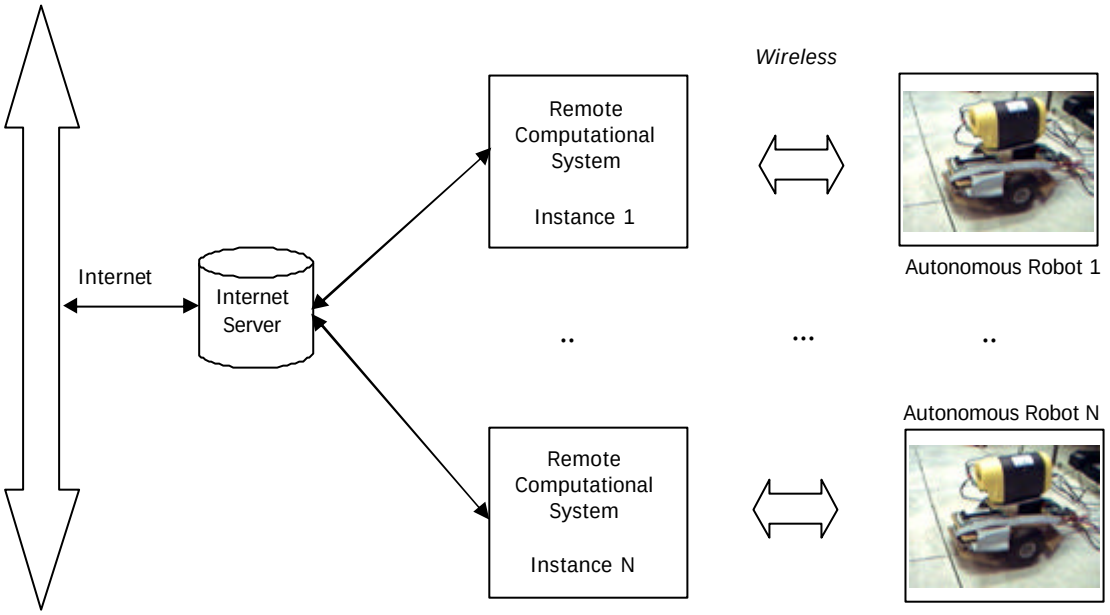


Figure 21. MIRO embedded robotic architecture consisting of multiple autonomous robots linked to their own instance of the distributed neural ASL/NSL computational system. All such instances are connected to Internet for remote monitoring.

In Figure 22, we show a sample processing cycle in the integrated ASL/NSL/MIRO architecture.

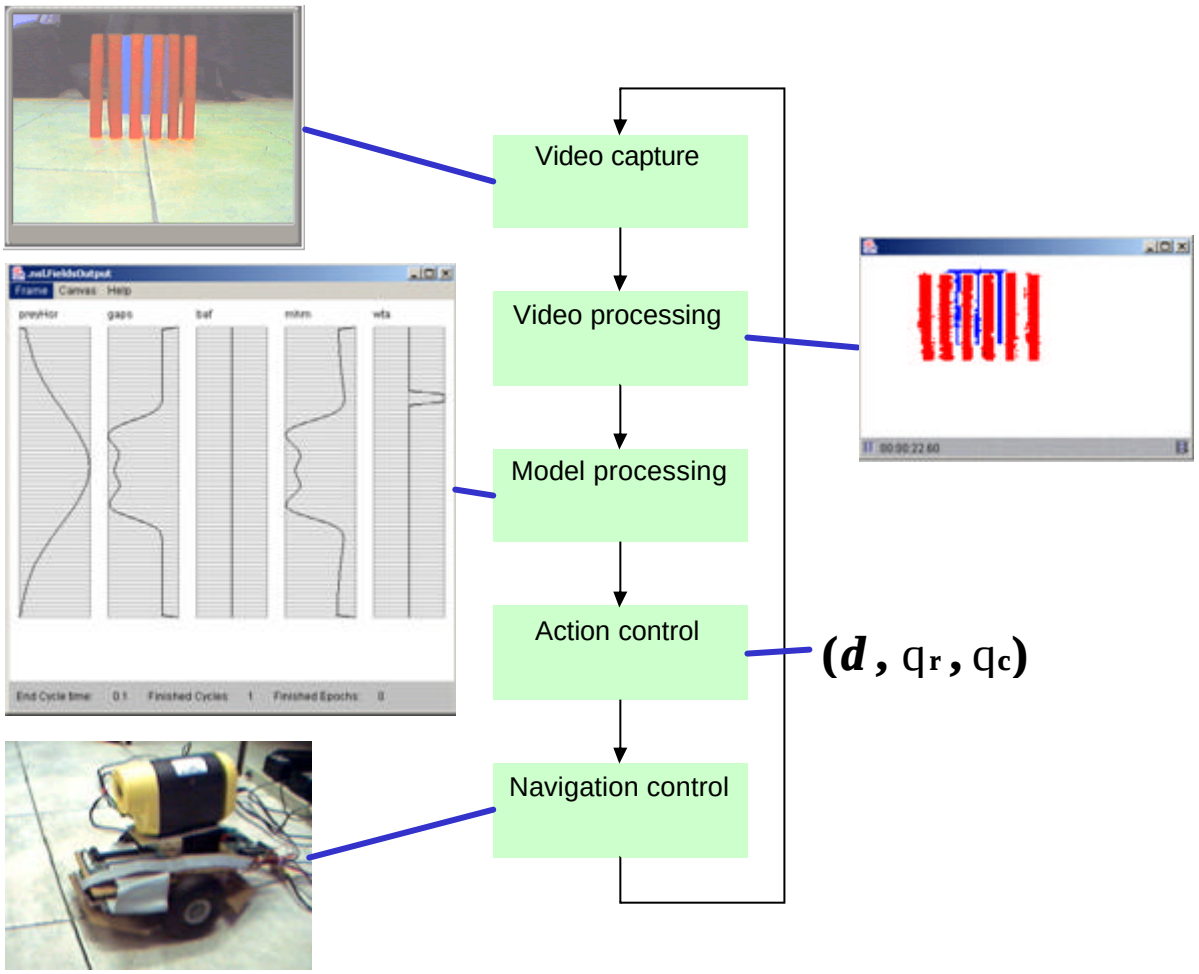


Figure 22. Cycle of computation for the different prey acquisition with detour experiments using the ASL/NSL and MIRO architecture. We use in this example a blue object to represent a prey and a red object represents a fencepost. The model output consists of a robot distance *d* to advance, a robot orientation *qr* and a camera orientation *qc*.

The processing stages are as follows:

- **Video Capture**. Images are obtained from the simulated or real robot camera via wireless transmission.
- **Video Processing**. Objects are recognized by color, with blue corresponding to a prey and red to a fencepost.
- **Model Processing**. Curves represent prey and static object attractant and repellent fields. Starting from the left in Figure 22, the first curve *preyHor* represents a prey attractant field, *gaps* represents the barrier repellent field, *baf* becomes active when the robot bumps into the barrier, *mhm* represents the summations of the different attractant and repellent fields, and *wta* produces an orientation angle for either forward or backward motions, the latter only after bumping has been produced.
- **Action Control**. Robot displacement d, orientation q_r for the robot and q_c for the camera are computed from model processing output. Note than in Kroax robot design, the camera can rotate independently from robot body.
- **Navigation Control**. The robot is controlled remotely via wireless transmission.

In Figure 23, we show a sample display as visualized directly from an Internet browser. Two images are displayed, the left one corresponding to an aerial camera and the one to its right corresponding to the local robot camera. In the image two blue cylinders are displayed corresponding to two preys. The top-right frame corresponds to the ASL/NSL console where both input and output can be controlled. The lower graphs represent model output from the two preys. Note how the *wta* curve shows a single "winner" from the two preys.

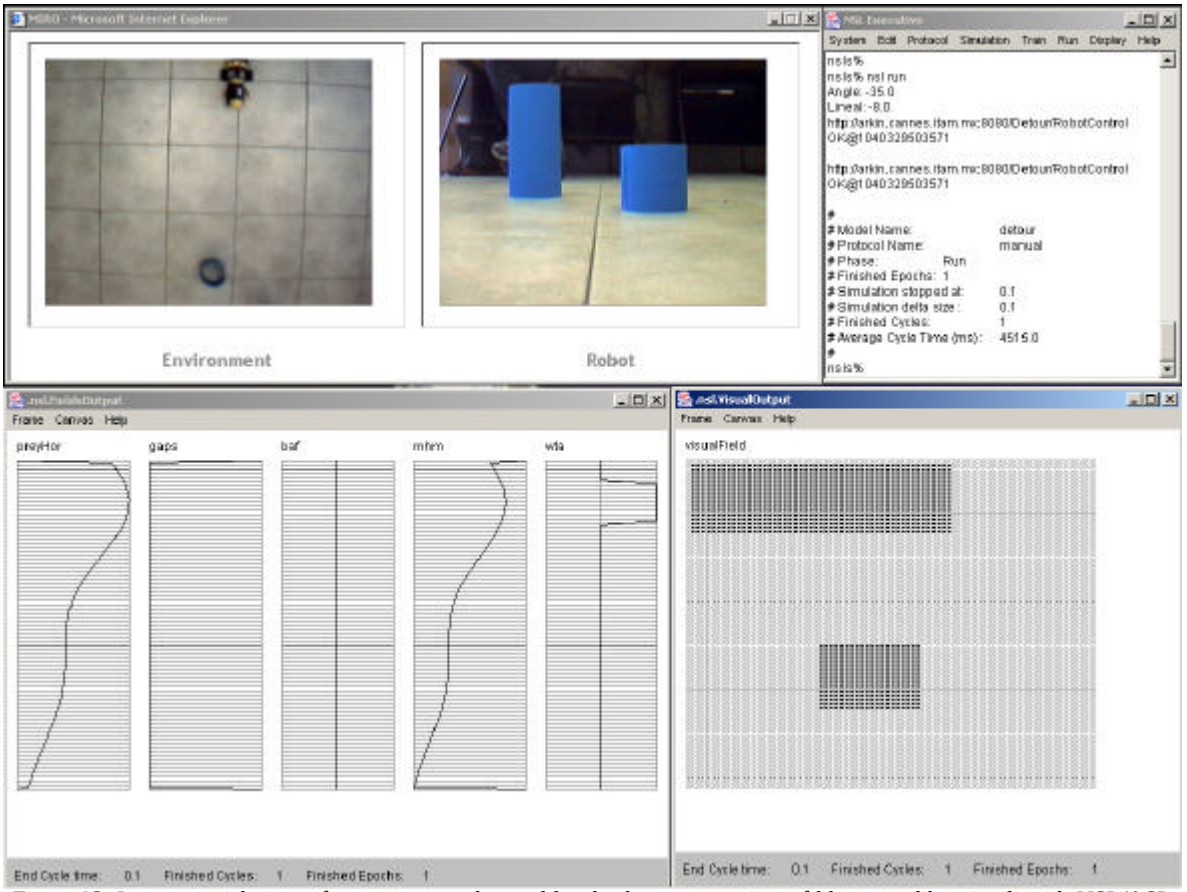


Figure 23. Internet aerial view of autonomous robot and local robot camera view of blue prey-like stimuli with NSL/ASL frames showing results from different visual and neural modules in a basic prey acquisition robot experiment.

5. SIMULATION AND EXPERIMENTAL RESULTS

The model is tested under simulation and robotic experimentation with each detour experiment run independently and autonomously. Learning from Experiment II to Experiment III is managed externally in order to keep the two experiments independent.

5.1 Model Simulation

The three experiments described in section III were simulated varying both barrier size (10cm and 20cm) as well as learning conditions applied to the 20cm barrier experiment. These are described next.

Experiment I

A number of internal fields are shown in Figure 24 for experiment I consisting of a 10cm wide barrier. The top graph describes the barrier repellent field *gaps*, followed by the prey *prey_hor*

attractant field, the *mhm* field integrating prey and barrier fields, followed by the orientation field *wta* and the bumping activity *baf*. (Note that these graphs are displayed horizontally as opposed to the vertical display from Figure 22.)

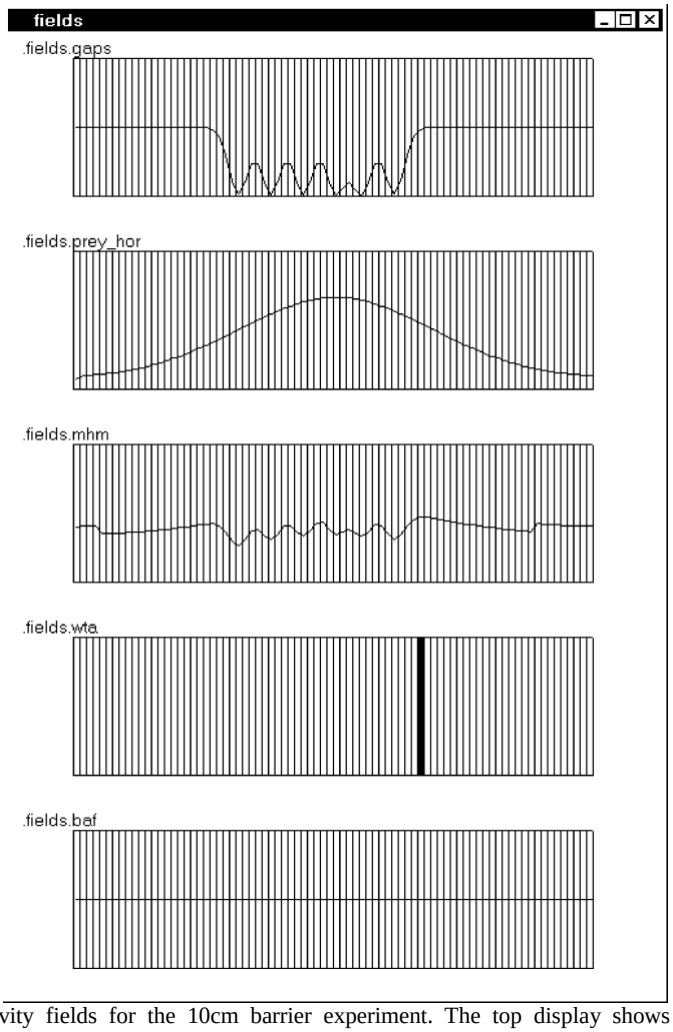


Figure 24. Different activity fields for the 10cm barrier experiment. The top display shows the repulsive field *gaps* generated from the barrier. The next display down represents the attraction field *prey_hor* generated from the prey. The next display down *mhm* represents the combined *gaps* and *prey_hor* fields. The next display down *wta* represents the heading or frog's orientation when moving forwards. The last display *baf* is currently empty and represents activity due to barrier bumping

The movement direction is obtained from the weighted sum of prey attraction and barrier repulsion fields. In this experiment the direction of movement is towards the right side of the barrier falling within the visual field of the frog. The resulting simulated motion trajectory is shown in Figure 25.

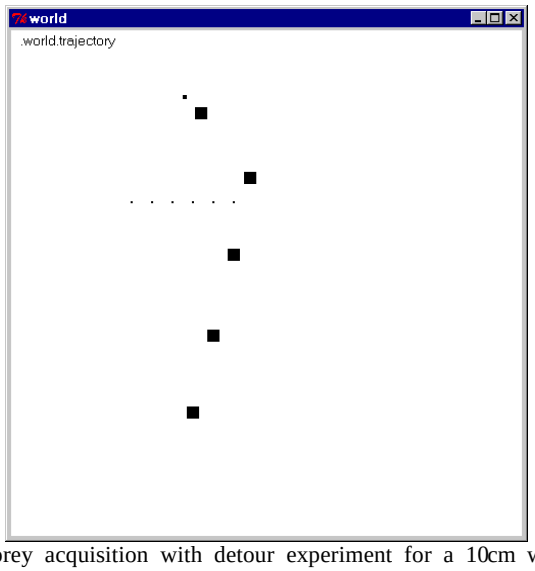


Figure 25. *Rana Computatrix* prey acquisition with detour experiment for a 10cm wide barrier. The different dots correspond to the frog's trajectory. Note how the frog heads directly towards the edge of the barrier.

Experiment II

A number of internal fields are shown in Figure 26 for experiment II consisting of a 20cm wide barrier. The fields displayed in the graph are the same as those displayed in Figure 24.

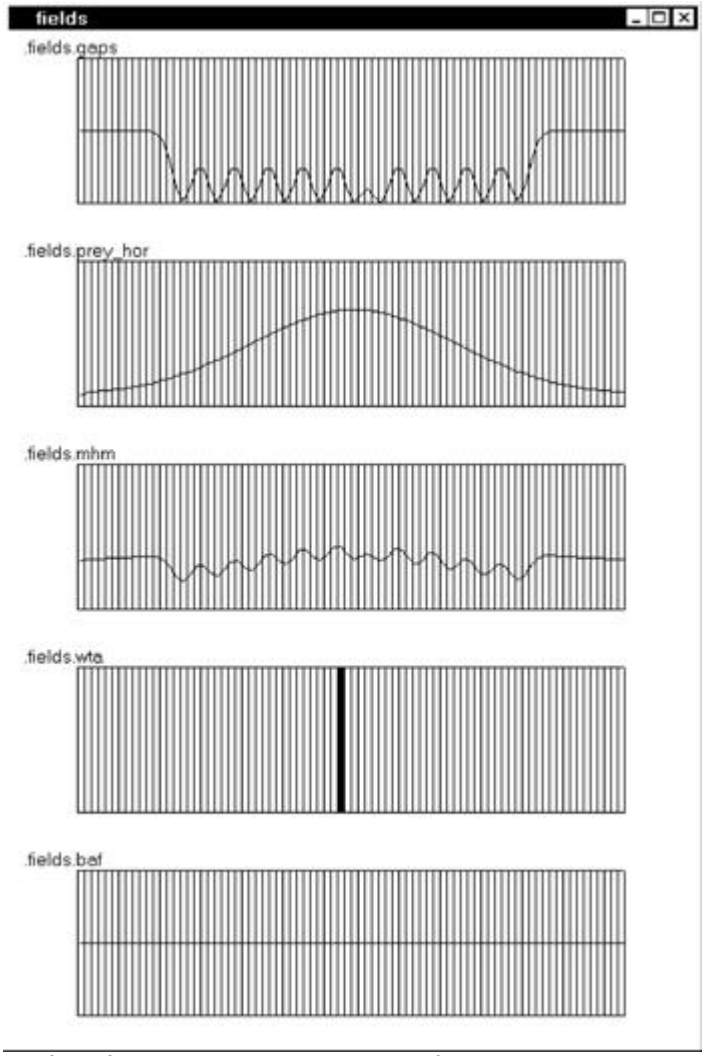


Figure 26. Different activity fields for the 20cm barrier experiment before bumping into the barrier. The top display shows the repulsive field *gaps* generated from the barrier. The next display down represents the attraction field *prey_hor* generated from the prey. The next display down *mhm* represents the combined *gaps* and *prey_hor* fields. The next display down *wta* represents the heading or frog's orientation when moving forwards. The last display *baf* is currently empty and represents activity due to barrier bumping.

Frog movement direction results again from the combination of prey attraction and barrier repulsion fields. In this experiment the direction of movement is towards the middle of the barrier resulting in the frog bumping into the barrier. Once the frog hits the barrier a bumping field is generated making the frog backup and redirecting its motion. The resulting field after bumping is shown in Figure 27.

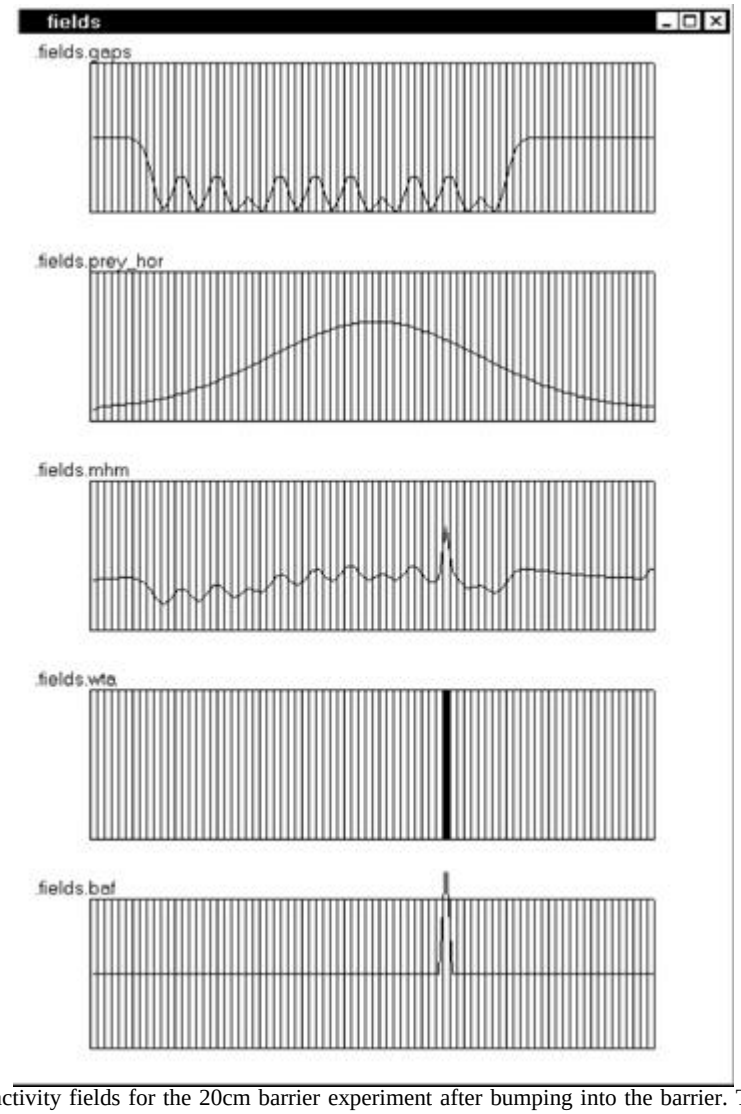


Figure 27. Different activity fields for the 20cm barrier experiment after bumping into the barrier. The top display shows the repulsive field *gaps* generated from the barrier. The next display down represents the attraction field *prey_hor* generated from the prey. The next display down *mhm* represents the combined *gaps*, *prey_hor,* and *baf* fields. The next display down *wta* represents the heading or frog's orientation when moving forwards. The last display *baf* shows activity due to barrier bumping.

The resulting motion trajectory is shown in Figure 28. After hitting the barrier several times the frog finally perceives the edge of the barrier and is able to advance towards the prey. The different steps are numbered to identify intermediate trajectories. Note that three backups are performed in this experiment by the simulated frog before reaching the edge of the barrier:

1. Forward: 1? 2 ? 3 ? 4 (bumping)
2. Backward: 4 ? 5 (reorient)
3. Forward: 5 ? 6 ? 7 (bumping)
4. Backward: 7 ? 8 (reorient)
5. Forward: 8 ? 9 ? 10 (bumping)
6. Backward: 10 ? 11 (reorient)
7. Forward: 11 ? 12 (mismatch)
8. Forward: 12 ? 13 ? 14 ? 15 ? 16 (prey acquisition)

As discussed in the adaptation section in 3.3, the frog avoids hitting the same gap repeatedly. We implemented this adaptation by doing a simplified backup and reorientation action.

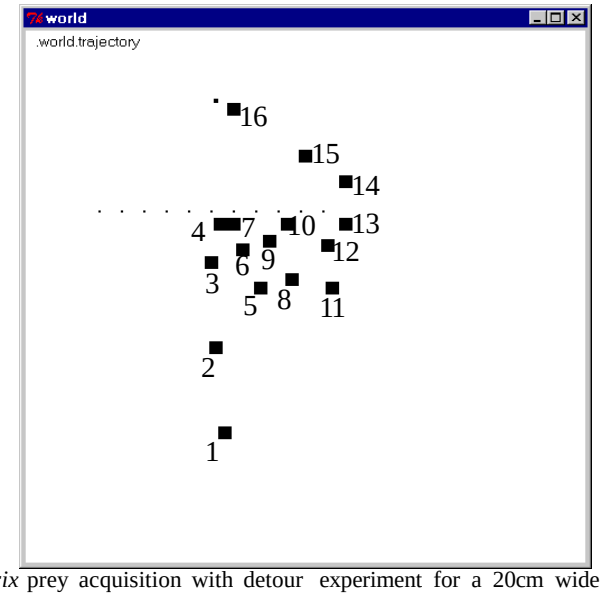


Figure 28. *Rana Computatrix* prey acquisition with detour experiment for a 20cm wide barrier before learning. The different dots correspond to the frog's trajectory until it finally reaches the prey. Note how the frog heads towards the edge of the barrier only after having bumped into it several times. We have added numbers corresponding to the frogs position in time. In this experiment the frog hits the barrier three times before perceiving the side of the barrier.

Experiment III

The model internal fields for experiment III are shown in Figure 29 consisting of a 20cm wide barrier where the frog has previously performed the task two to three times and learned how to approach the barrier edge, falling outside its visual field.

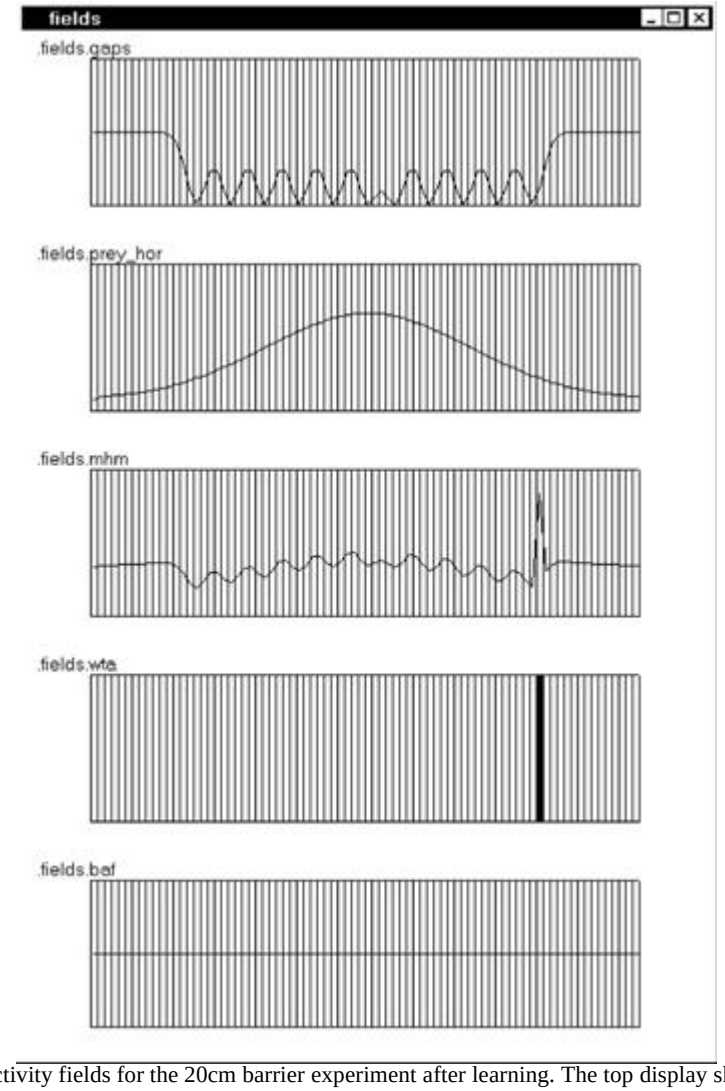


Figure 29. Different activity fields for the 20cm barrier experiment after learning. The top display shows the repulsive field *gaps* generated from the barrier. The next display down represents the attraction field *prey_hor* generated from the prey. The next display down *mhm* represents the combined *gaps*, *prey_hor*, and *baf* fields. The next display down *wta* represents the heading or frog's orientation when moving forwards. The last display *baf* is empty and represents activity due to barrier bumping. The frog directly redirects itself towards the edge of the barrier. Note how an extra field added to *mhm* to orient the frog directly towards the edge.

Note that although no bumping occurs, the *mhm* field generates a new field caused by learning. The new field in *mhm* field also causes a larger sidestep towas discussed in section 3.3 caused by the second adaptation stage. The resulting behavior is shown in Figure 30.

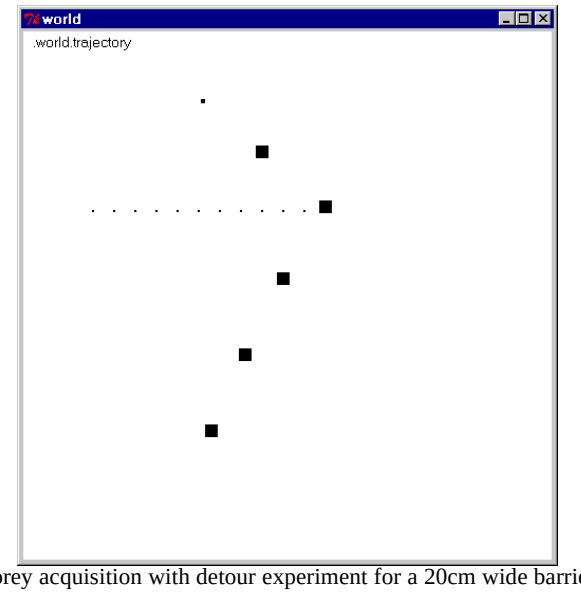


Figure 30. *Rana Computatrix* prey acquisition with detour experiment for a 20cm wide barrier after learning. The different dots correspond to the frog's trajectory. Note how the frog heads directly towards the edge of the barrier.

5.2 Robotic Experimentation

The following experiments were run with the Kroax robot described in Section 4 having various barrier sizes (10cm and 20cm) [61].

Experiment I

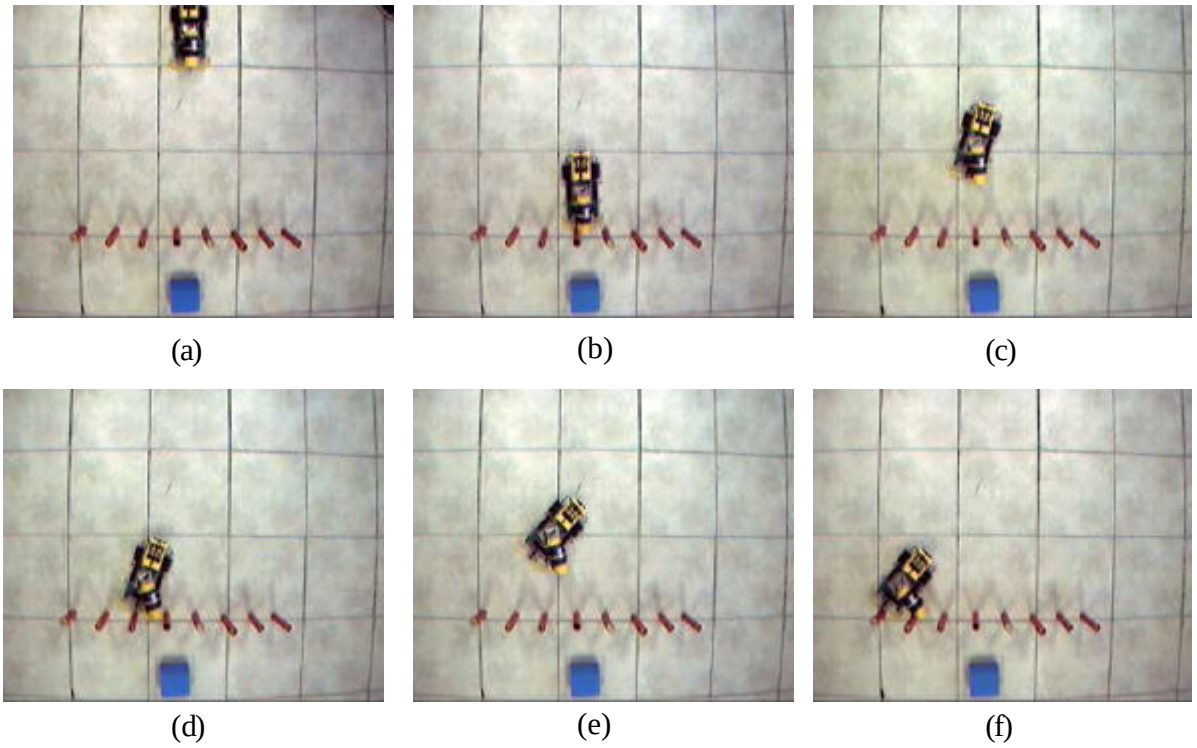
In Figure 31, a sequence of pictures is shown for experiment I. Note in (b) and (c) how the camera rotates independently from the robot body in keeping a view of the prey at all times. The edge of the barrier falls within the visual field of the robot at all times thanks to the camera rotation.

(a) (b) (c)

Figure 31. Results from prey acquisition experiment for 10cm barrier with direct detour around barrier: (a) robot starts looking at prey with barrier in front, (b) robot moves directly to the edge of the barrier, (c) robot completes detour.

Experiment II

In Figure 32, a sequence of pictures is shown for experiment II. In this experiment the robot bumps into the barrier several times before being able to eventually circle around the barrier. In the set of images from (a) to (f) the robot bumps into the barrier 3 times. The results correspond very close to those obtained in Experiment II in the simulation section. Yet, there were some discrepancies in robot movement primarily due to slippage and coarse control on actual movement actions. The edge of the barrier falls outside the visual field of the robot.



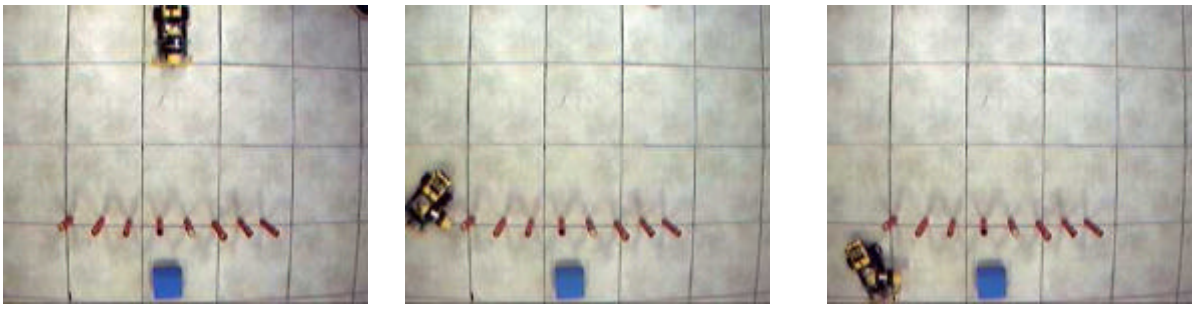
(a) (b) (c)

(d) (e) (f)

Figure 32. Results from prey acquisition experiment for 20cm barrier before learning: (a) robot starts looking at prey with barrier in front, (b) robot moves directly towards the barrier bumping into it, (c) robot backs up from the barrier, (d) robot moves again towards the barrier bumping into it, (e) robot backs up, (f) robot moves again towards the barrier bumping into it. Note that in the last bump the robot is very close to the edge of the barrier. Eventually the robot perceives the edge of the barrier and completes the detour.

Experiment III

In Figure 33, a sequence of pictures is shown for experiment III. In this experiment the robot has already learned from the previous trials executed in experiment II and is able to circle around the 20cm barrier. In the current robot experiment the robot reaches the edge after a single intermediate step similar to the frog experiment. This corresponds to the sidestep adaptation process described in section 3.3. Note that the actual experiment is managed independently from experiment II. The human operator relocates the robot in its initial position and restarts to experiment execution. The edge of the barrier falls outside the visual field of the robot.



(a) (b) (c)

Figure 33. Results from prey acquisition experiment for 20cm barrier after learning: (a) robot starts looking at prey with barrier in front, (b) robot moves directly to the edge of the barrier, (c) robot completes detour.

Figure 34 summarizes the different trajectories followed by the robot during the three experiments, where (a) corresponds to the 10cm trajectory, (b) corresponds to the 20cm trajectory before learning, and (c) corresponds to the 20cm after learning. Note that the picture in (a) is a "amplified" from those in (b) and (c) yet the trajectory is similar to that in (c). Thus, the robot in (c) is performing a similar detour behavior after learning to that when the edge of the barrier fell within its visual field. Learning in this experiment duplicates the frog's ability to learn from experience and call upon in a novel situation, i.e, where the frog does not see the edge of the barrier, behaviors from its repertoire of capabilities.

(a) (b) (c)

Figure 34. Results from the three prey acquisition experiments showing the different trajectories followed by the robot:: (a) robot trajectory for 10cm barrier, (b) robot trajectory for 20cm barrier before learning, (c) robot trajectory for 20cm barrier after learning.

6. CLOSING THE RESEARCH LOOP

In the introduction of the paper we motivated this work with our long term vision of tightly coupled research involving neuroscientific experimentation, theoretical modeling, simulation and robotics experimentation as shown in Figure 1. The work presented in the paper has dealt with the "forward" cycle in this research framework, from neuroscience to robotics. Yet, the "backward" cycle is crucial in pursuing these goals. While some advances have been done in this direction by our group and others (see also our work in rat learning to navigate working closely with neurophysiologists [10, 11]), there is much to be done to this respect. In this section we discuss some shortcomings found during robot experimentation that requires further studies with frogs:

1. Although during simulation this was not an important factor, during robotic experimentation we found that the robot would keep loosing sight of the prey after a slight reorientation. This was caused by the very limited visual field of the cameras used in the robot. Although we compensate for this shortcoming by reorienting back the camera in order to have the prey back in sight (see Flores and Weitzenfeld [33] for a model inspired on monkey saccades), we need further understandings on how the frog would behave when close to loosing track of the prey. Since in general the frog only reacts to the presence of a (moving) stimulus it is important to design experiment that will force the frog to a similar situation. The experimental setup shown in Figure 35 (a) has been designed for that purpose, where the frog will need to rotate 180 degrees before being able to reach the prey. The length and angle on the side barriers need to be adjusted during experimentation.

2. Another aspect that needs further exploration is how much can the frog adapt and learn from similar but yet more complex set of barrier-related experiments. In Figure 35 (b) we have designed an example of a two set of parallel barriers adding more complexity into the experiment. The goal is to find how complex can the original barrier experiment become and how many trials will the frog take in learning them.

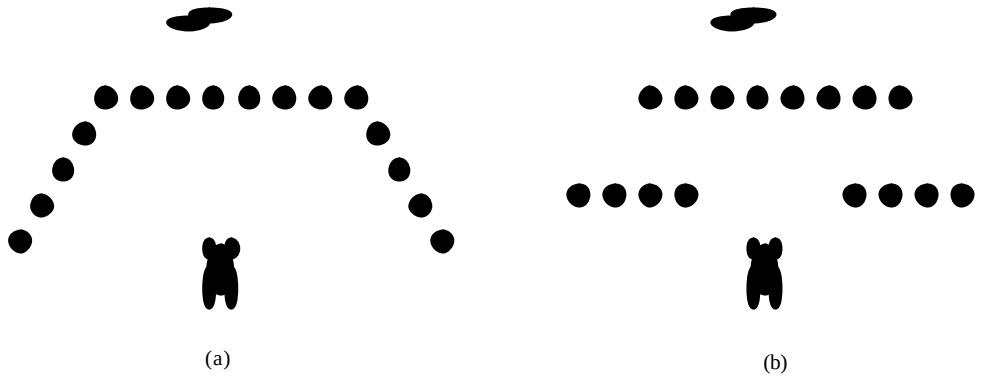


Figure 35. Setup for new neuroscientific behavioral experiments to help understand the frog's capability: (a) to keep track of the prey while having to rotate back 180 degrees, and (b) learn more complex detour tasks.

These are set of experiments we are starting to do and exemplify the "backward" research path from robotics back to experimental neuroscience.

7. CONCLUSIONS AND DISCUSSION

The work presented in this paper reviews the challenges and complexity in modeling robotic systems inspired by neuroethological animal studies. The motivation behind this work is to provide neuroscientists a testbed for robotic experimentation and to provide roboticists tools and methodologies in the development of biologically inspired systems at different levels of granularity. It should be noted that the frog prey acquisition with detour model is only one example of a multi-level schema and neural network application developed using ASL/NSL and tested in real robots using the MIRO architecture. In the rest of this section we analyze the scope of present and future work by our group related to multi-level biological robotics architectures.

Schema and neural network modeling

In terms of neuroethological modeling, we manage complexity by using a multi-level approach emphasizing both top-town and bottom-up designs through different levels of granularity. At the top-level, behaviors are described in terms of schema models such as the frog prey acquisitions, predator and static object avoidance. Schemas may be refined in a hierarchical manner until reaching lower-level neural networks for their implementations. Neural implementation may include simpler or more detailed neuron models, such as those requiring synaptic plasticity, with systems such as GENESIS [13] and NEURON [36]. We would like in the future to be able to "reuse" neural modules in more comprehensive neuroethological robotic architectures. Yet, a major challenge is to be able to link independently developed neural schemas where input and output specifications do not necessarily match. For example, the original neural models for MaxSelector, Tectum and Pretectum incorporated coarse visual input instead of more detailed R2, R3, and R4 retina class cells in the Retina [23] requiring a redesign of the more general VisualField found in the model and in other modules such as

Depth Perception [55]. Also, the logic of one module may be based on different assumptions from those of other modules, for example, parameters, time frequencies or even the particular experiment used to design the neural module.

Prey acquisition with detour

The frog prey acquisition with detour model presented in this paper explains basic characteristics about detour behavior. In experiment I, if the retinotopic representation of the barrier edge falls within the prey visual field, then the summation of activity from the prey attractant field and the barrier repellent map will result in a detour of the frog around the barrier. In experiment II, in the case of wide barriers, the prey attractant field falls beyond the frog visual field, thus triggering an approach to a point within the barrier map so long as the peak of prey attraction exceeds the barrier inhibition. Thus, the model predicts that the naive frog would approach wide barriers rather than detour around them. In experiment III, an artificial detouring field is added at the edge of the barrier to make the frog move around the barrier without the need to perceive the actual edge. In terms of visual processing it is important to understand that in the current experimental results only monocular vision is modeled. We plan in the future to extend the model by distinguishing monocular optic flow when far from the prey from binocular depth perception when ready to snap at the prey [40, 41].

Simulation versus robot embodiment

Most brain modeling is currently done through simulation, but simulation is not quite the same as real-world robotic experimentation. In particular, many shortcuts are taken during simulation. For example, simulated cameras can project very large visual fields with "perfect" images and object sizes and without worrying about lighting conditions. Once models are experimented under real world conditions objects become less "perfect" and harder to recognize. These aspects need to be treated with real robot experimentation. Another constraint with robotic experimentation that appeared in our prey acquisition with detour experiment, relates to "losing" the prey once the robot orients towards the edge of the barrier. In the simulated version the robot perceived the prey as well as barrier due to the laerger size of the visual field. While frog visual fields are quite large the actual model had to be modified in dealing with the smaller visual field. Initially we solved this problem using geometric corrections to orient back the robot camera to its original viewing angle. A different approach we later modeled was inspired in the use of "memory" saccades as in the monkey oculomotor system [33]. Yet, the detour model is based on toad and frog studies, while the oculomotor system is based on monkey studies. To neurobiologist this is quite significant. On the other hand, to robotic designers this may not be necessarily critical.

Distributed Neural Processing

Large neural network models produce great amounts of data and consume extensive processing cycles. A typical computation cycle starts by obtaining sensory input (visual and tactile) and ends by producing motor output. Cycles continue indefinitely or until some specific task is achieved, such as reaching the prey. For example, a "typical" retina model [49] may consist of more than 100,000 neurons and half a million interconnections requiring many hours of simulation to complete these cycles. The expensive nature of neural computation is further exacerbated by the fact that a comprehensive schema-neural model includes multiple neural networks. This becomes even worse in the case of higher-level animals involving extensive behaviors and other brain regions [4]. By taking advantage of the parallel and distributed nature

of neural network computation [54] we have extended the original NSL/ASL simulation system to distributed architectures [59].

Distributed Robot Architecture

In recent years increasing effort is spent in embedding mobile robotic systems into computer networks via wireless communication [28]. This approach makes it possible not only to control and monitor remote robots, such as with the Mars explorers, but also enhance their capabilities by linking the robot to remote computational resources, such as image processing or neural processing. These efforts have highlighted the benefits of embedded systems in robotics via Internet [34, 46]. In such a way, processing is distributed between local robotic hardware and remote computational system. Although it is possible to share robot "intelligence" among multiple robots where applications could easily take advantage of information sharing (see [9] for a discussion on distributed versus centralized robotic systems), we are particularly interested in keeping a truly distributed autonomous robot architecture where neuroethological experimentation can be conducted. Under the MIRO architecture: (i) time-consuming processes are carried out in the (neural) computational system, implemented using the NSL/ASL system while (ii) sensory input, motor output and other limited tasks are carried out in the robot hardware. In such a way, the computational system provides the robot's "intelligence", while the robot does limited processing. The most important challenge in this type of architectures is how to achieve real-time performance. We are at this time assessing the efficiency of the MIRO embedded architecture. A number of issues we are addressing relates to what happens when communication between the robot and computational system fails or becomes extremely slow or unreliable. When such situation arises, the robot can respond in a number of ways, do nothing until communication is restored, end its mission, or perform limited tasks that may put it back in action such as actively searching for a location where communication can be reestablished.

Multi-robot systems

Additionally, most of our experiments have involved single robots mostly due to the underlying complexity of neuroethological models. Current work relates to multi-robot neuroethological experimentation not only in frogs but also with other animals such as wolves in group hunting behaviors [50].

Adaptation and Learning

This is an important area in both biological studies and robotic architectures. The frog detour model is quite significant to roboticists in that learning occurs after only a couple of trials. As opposed to many learning techniques such as reinforcement learning [48] and backpropagation [61] having large convergence times due in particular to their lack of structure in the weight space, learning in this model is very focused in terms of a "simple" detour task. As previously mentioned, learning is restricted to the repertoire of existing capabilities where these capabilities are adapted to the task at hand.

8. REFERENCES

[1] Arbib, M.A., Levels of Modelling of Mechanisms of Visually Guided Behavior, *Behavior Brain Science* 10:407-465, 1987.

[2] Arbib, M.A., *The Metaphorical Brain 2*, Wiley, 1989.

[3] Arbib, M.A., Schema Theory, in *The Handbook of Brain Theory and Neural Networks*, 2nd Edition, Editor Michael Arbib, 993-998, MIT Press, 2002.

[4] Arbib, M.A., Erdi, P. and Szentagothai, J., *Neural Organization: Structure, Function and Dynamics*, MIT Press, 1998.

[5] Arbib, M.A. and Lee, H. B., Anuran visuomotor coordination for detour behavior: from retina to motor schemas, Proceedings From Animals to Animats III, Editors Cliff, D., Husbands, P., Meyer, J.M. and Wilson, S.W., MIT Press, 1994.

[6] Arkin, R.C., Behavioral based Robotics, MIT Press, 1998.

[7] Arkin, R.C., Ali, K., Weitzenfeld, A., and Cervantes-Perez, F., Behavior Models of the Praying Mantis as a Basis for Robotic Behavior, *in Journal of Robotics and Autonomous Systems*, 32(1), pp 39-60, Elsevier, 2000.

[8] Arkin, R.C., Cervantes-Perez, F., and Weitzenfeld, A., "Ecological Robotics: A Schema-Theoretic Approach", "Intelligent Robots: Sensing, Modelling and Planning", eds. R.C.Bolles, H.Bunke, and H.Noltemeier, pp 377-393, World Scientific, 1997.

[9] Balch, T. and Arkin, R.C., Communication in Reactive Multiagent Robotic Systems, Autonomous Robots, 1, pp 1-25, 1994.

[10] Barrera, A., and Weitzenfeld A., Return of the Rat: Biologically-Inspired Robotic Exploration and Navigation, BioRob 2006, Feb 20-22, Pisa, Italy, 2006.

[11] Barrera, A., and Weitzenfeld A., Bio-inspired Model of Robot Adaptive Learning and Mapping, IROS – International Robots and Systems Conference, Beijing, China, Oct 9-13, 2006.

[12] Beer, R. D., Intelligence as Adaptive Behavior: An Experiment in Computational Neuroethology, San Diego, Academic Press, 1990.

[13] Bower, J.M., and Beeman, D., The Book of GENESIS, Exploring Realistic Neural Models with the General Neural Simulation System, Telos, Springer-Verlag, 2nd Edition, 1998.

[14] Cervantes-Perez, F., Lara, R., and Arbib, M.A., A neural model of interactions subserving prey-predator discrimination and size preference in anuran amphibia, Journal of Theoretical Biology, 113, 117-152, 1985.

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

[16] Cervantes-Perez, F., Herrera, A., and García, M., Modulatory effects on prey-recognition in amphibia: a theoretical 'experimental study', in *Neuroscience: from neural networks to artificial intelligence*, Editors P. Rudomin, M.A. Arbib, F. Cervantes-Perez, and R. Romo, Springer Verlag Research Notes in Neural Computing vol 4, pp. 426-449, 1993.

[17] Cliff, D., Neural Networks for Visual Tracking in an Artificial Fly, in *Towards a Practice of Autonomous Systems: Proc. of the First European Conference on Artificial Life (ECAL 91)*, Editors, F.J., Varela and P. Bourgine, MIT Press, pp 78-87, 1992.

[18] Cobas, A., and Arbib, M.A., Prey-catching and Predator-avoidance in Frog and Toad: Defining the Schemas, *J. Theor. Biol* 157, 271-304, 1992.

[19] Collett, T., Picking a route; Do toads follow rules or make plans? *(Advances in Vertebrate Neuroethology*, J.P. Ewert, R.R. Capranica and D.J. Ingle, Eds), pp.321 – 330, 1983.

[20] Corbacho, F., and Arbib M. Learning to Detour, *Adaptive Behavior*, Volume 3, Number 4, pp 419-468, 1995.

[21] Corbacho, F., Nishikawa, K.C., Weerasuriya, A., Liaw, J.S. and Arbib, M.A., Schema-based learning of adapatable prey-catching in anurans I. The basic architecture, Biological Cybernetics, 93:391-409, 2005.

[22] Corbacho, F., Nishikawa, K.C., Weerasuriya, A., Liaw, J.S. and Arbib, M.A., Schema-based learning of adapatable prey-catching in anurans II. Learning after lesioning, Biological Cybernetics, 93:410-425, 2005.

[23] Corbacho, F., and Weitzenfeld, Retina, in *The Neural Simulation Language NSL, A System for Brain Modeling*, pp 189-206, MIT Press, July 2002.

[24] Corbacho, F., and Weitzenfeld, Learning to Detour, in *The Neural Simulation Language NSL, A System for Brain Modeling*, pp. 319-342, MIT Press, July 2002.

[25] Didday, R.L., A model of visuomotor mechanisms in the frog optic tectum, *Math. Biosci.* 30:169-180, 1976.

[26] Dominey, P., and Arbib, M.A., A cortico-subcortical model for generation of spatially accurate sequential saccades, Cerebral Cortex, 2, pp 153-175, 1992.

[27] Droulez, J. & Berthoz, A. A neural network model of sensoritopic maps with predictive short-term memory properties. *Natl. Acad. Sci. USA*, 88: 9653-9657, 1991.

[28] Estrin, D., Govidian, R., and Heidemann, J. (Eds.) Special issue on Embedding the Internet, Communication of the ACM, 43(5), May 2000

[29] Ewert, J. P., Single unit response of the toad's (Bufo americanus) caudal thalamus to visual objects. Z. vergl. Physiol. 74, 81-102, 1971.

[30] Ewert, J.P, Neuroethology, an introduction to the neurophysiological fundamentals of behavior, Springer-Verlag, 1980.

[31] Ewert, J. P. Neuroethology of releasing mechanisms: Prey-catching in toads, Behavior Brain Science 10, 337–405, 1987.

[32] Fagg, A., King, I., Lewis, A., Liaw, J., Weitzenfeld, A., A Testbed for Sensorimotor Integration, Proceedings of IJCNN '92, Baltimore, MD, 1:86-91, 1992.

[33] Flores Ando, F., and Weitzenfeld, A., Visual Input Compensation using the Crowley-Arbib Saccade Model, Proc. International Conference on Advanced Robotics ICAR, Seattle, USA, July 17-20, 2005.

[34] Goldberg, K., and Siegwert, R., (eds), Beyond Webcams: An Introduction to Online Robots, MIT Press, 2002.

[35] Grobstein, P. Directed movement in the frog; motor choice, spatial representation, free will? In *Neurobiology of motor programme selection* (Eds. J. Kien, C. R. McCrohan, & W. Winlow). Pergamon Press: Oxford, 1992.

[36] Hines, M., and Carnevale, T., The NEURON Simulation Environment, *Neural Computation*, 9:1179-1209, 1997.

[37] Hodgkin, A.L. and Huxley, A.F., A quantitative description of membrane current and its application to conduction and excitation in nerve, *Journal of Physiology*, 117, 500-544, 1952.

[38] House, D., Depth Perception in Frogs and Toads: A study in Neural Computing, Lecture Notes in Biomathematics 80, Springer-Verlag, 1985.

[39] Ingle, D., Brain mechanisms of visual localization by frogs and toads. (*Advances in Vertebrate Neuroethology*, J. -P. Ewert, R. R. Capranica and D. J. Ingle, Eds), 177 – 226, 1983.

[40] Lecorne, S., and Weitzenfeld, A., Robot Navigation using Stereo Vision, Proc. 1st IEEE-RAS Latin American Robotics Symposium, ITAM, Mexico City, October 28-29, 2004.

[41] Lock, A. & Collett, T. A toad's devious approach to its prey: a study of some complex uses of depth vision. *J. Comp. Physiol.*, 131: 179-189, 1979.

[42] Nolfi, S., Using emergent modularity to develop control system for mobile robots. Adaptive Behavior, 5, 1997.

[43] NSL Web Sites, http://www.neuralsimulationlanguage.org/ and http://nsl.usc.edu.

[44] Rall, W., Branching dendritic trees and motoneuron membrane resistivity, *Exp. Neurol.*, 2:503-532, 1959.

[45] Scalia, F., and Fite., K.V., A retinotopic analysis of the central connections of the optic nerve in the frog, J. Comp. Neurol., 158:455-478, 1974.

[46] Sukhatme, G.S., and Mataric, M.J., Embedding Robots Into the Internet, Communication of the ACM, 43(5) pp 67-73, Special issue on Embedding the Internet, D. Estrin, R. Govidian, and J. Heidemann, eds., May 2000.

[47] Sun, R., On schemas, logics, and neural assemblies, Applied Intelligence, 5(2):83-102, 1995.

[48] Sutton, R., and Barto, A., Reinforcement Learning: An Introduction, MIT Press, 1998.

[49] Teeters, J.L., and Arbib, M.A., A model of the anuran retina relating interneurons to ganglion cell responses, *Biological Cybernetics*, 64, 197-207, 1991.

[50] Vallesa, A., and Flores, H., Weitzenfeld, A., 2006, A Biologically-Inspired Wolf Pack Multiple Robot Hunting Model, LARS 2006, Santiago Chile, Oct 26-27.

[51] Webb, B. What does robotics offer animal behaviour? Animal Behaviour, 60, 545-558, 2000.

[52] Webb, B. and Harrison, R. Integrating sensorimotor systems in a robot model of cricket behaviour, Sensor Fusion And Decentralized Control In Robotic Systems III November 5-8, Boston. (Editors: McKEE GT, Schenker PS) Proceedings of the Society of Photo-Optical Instrumentation Engineers (SPIE), 4196, 113-124, 2000.

[53] Weitzenfeld, A., ASL: Hierarchy, Composition, Heterogeneity, and Multi-Granularity in Concurrent Object-Oriented Programming, *Proceedings of the Workshop on Neural Architectures and Distributed AI: From Schema Assemblages to Neural Networks*, USC, October 19-20, 1993.

[54] Weitzenfeld, A., Arbib, M., 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 1991.

[55] Weitzenfeld, A., and Arbib, M.A., Depth Perception, in *The Neural Simulation Language NSL, A System for Brain Modeling*, pp 171-187, MIT Press, July 2002.

[56] Weitzenfeld, A., Arbib, M., Alexander, A., *NSL - Neural Simulation Language: A System for Brain Modeling*, MIT Press, July 2002.

[57] Weitzenfeld A., "A Multi-level Approach to Biologically Inspired Robotic Systems", *Proc of NNW 2000 10th International Conference on Artificial Neural Networks and Intelligent Systems*, Prague, Czech Republic, Juy 9-12, 2000.

[58] Weitzenfeld, A., Cervantes, F., Sigala, R., NSL/ASL: Simulation of Neural based Visuomotor Systems, in *Proc. of IJCNN 2001 International Joint Conference on Neural Networks*, Washington DC, July 14-19, 2001.

[59] Weitzenfeld A., Gutierrez-Nolasco, S., "ASL/NSL: A Multi-level Computational Model for Distributed Neural Simulation", in Proc of SCSC 2000 Summer Computer Simulation Conference, Vancouver, Canada, July 16-20, 2000.

[60] Weitzenfeld A., Gutierrez-Nolasco S., and Venkatasubramanian N., MIRO: An Embedded Distributed Architecture for Biologically inspired Mobile Robots, Proc ICAR-03, 11th International Conference on Advanced Robotics, June 30 – July 3, Coimbra, Portugal, 2003.

[61] Weitzenfeld A., Frog Prey Acquisition with Detour "Videos", http://www.cannes.itam.mx/videos/detour, 2005.

[62] Werbos, P., The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting, Wiley, 1994.