Visualization of Multi-level Neural-based Robotic Systems

Alfredo Weitzenfeld
ITAM
Río Hondo #1, CP 01000
México, DF, MEXICO
alfredo@lamport.rhon.itam.mx

Ronald Arkin
Georgia Tech
Atlanta, GA 30332-0280
USA
arkin@cc.gatech.edu

Francisco Cervantes
ITAM
Río Hondo #1, CP 01000
México, DF, MEXICO
cervante@lamport.rhon.itam.mx

José Francisco Peniche
ITAM
Río Hondo #1, CP 01000
México, DF, MEXICO
cj@sutherland.rhon.itam.mx

Abstract[1]

Autonomous biological systems are very complex in their nature. Their study, through both experimentation and computation, provides a means to understand the underlying mechanisms in living systems while inspiring the development of technological applications. Experimentation, consisting of data gathering, generates predictions to be validated by experimentation on artificial systems. Computational models provide the understanding for the underlying dynamics, and serve as basis for simulation and further experimentation. The work presented here involves analyzing how predictive models can be generated from biological systems and then be used to drive robotic experiments; and conversely, how can results from robotic experiments drive additional neuroethological data gathering. This process requires a variety of visualization techniques in modeling and simulation of increasingly complex systems.

1 Introduction

The study of autonomous biological systems comprises a cycle of biological experimentation, computational modeling and robotics experimentation, as depicted in Figure 1. This cycle serves as framework for the study of the underlying neural mechanisms responsible for behavior in animals and the inspiration for designing autonomous robotic systems.

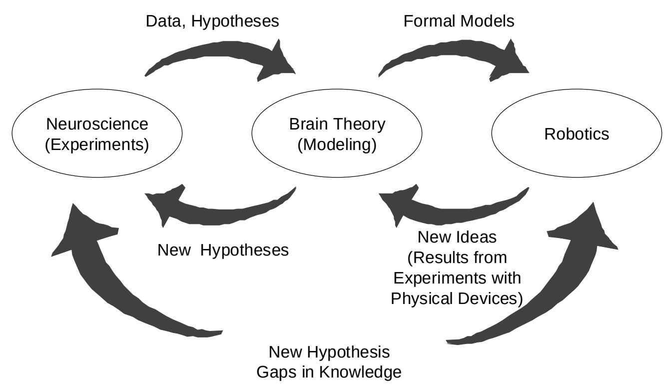

Figure 1. Framework for the study of living organisms through cycles of biological experimentation, computational modeling, and robotics experimentation.

While much work has been done on experimentation on living animals and the development of biological and artificial neural models in developing behavioral models for robotics; there exists a very limited effort to integrate across the different modeling levels currently applied to the study biological systems in a single unified approach. Two existing projects, "Ecological Robots: A Schema-theoretic Approach" [Arkin *et al*, 1997] and "Multi-level Simulation Methodology: A Computational and Experimental Approach to Neural Systems" [Weitzenfeld *et al*, 1998b], have the goal to develop a multi-level simulation methodology to answer some of the questions arising in highly complex neural systems which single level

[1] This research is supported by the National Science Foundation in the U.S. (grant #IRI-9505864) and CONACyT in Mexico (grants #546500-5-C006-A and #546500-5-C018-A)

models cannot. These biological neural systems are studied primarily with respect to four different levels of analysis: autonomous robotic agents, behavior, neural networks, and detailed neurons, as shown in Table 2.

Levels of Analysis	**Theoretical Approach: Simulation Tool**
1. Autonomous robotic agents	Sensors/Actuators: MissionLab (GeorgiaTech)
2. Behavior	Schemas: ASL (ITAM), MissionLab (GeorgiaTech)
3. Neural Networks	Neural Elements: NSL (ITAM-USC)
4. Neurons	Compartmental models, Cable Theory, Hodgkin-Huxley, ion channels: GENESIS [Bower and Beeman, 1994], NEURON [Hines, 1994]

Table 2. Multi-level analysis for the understanding of autonomous biological systems.

- At the highest level, autonomous robotic agents are designed to interact with the world via sensors and actuators. These agents are simulated in virtual robots and implemented in real robots with systems such as *MissionLab* [MacKenzie *et al*, 1997]. Autonomous robotic agents are exemplified by biologically inspired systems, such as the computational frog (*rana computatrix*) [Arbib, 1987], the computational praying mantis [Cervantes *et al*, 1993a], the computational cockroach [Beer, 1990], and the computational hoverfly [Cliff, 1992].
- At the behavior level, neuroethological data from living animals is gathered to generate single and multi-agent systems to study the relationship between an agent and its environment, giving emphasis to aspects such as cooperation and competition between agents. In our project, agent behavior is described in terms of perceptual and motor *schemas* [Arbib, 1992] decomposed and refined in a recursive fashion. Behaviors, and their corresponding schemas, are simulated via the abstract simulation language ASL [Weitzenfeld, 1993]. Examples of behavioral models include the praying mantis *Chantlitlaxia* (search for a proper habitat) [Cervantes et al, 1993a] and the frog and toad prey acquisition and predator avoidance models [Cobas and Arbib, 1992].
- At the neural network level, neuroanatomical and neuronphysiological data are used to generate perceptual and motor neural network models corresponding to the schema models developed at the behavioral level. These models try to explain the underlying mechanisms for sensorimotor integration. Neural networks are simulated via the neural simulation language NSL [Weitzenfeld and Arbib, 1994][Weitzenfeld *et al*, 1998a]. Neural network models are exemplified by the prey acquisition and predator avoidance neural models [Cervantes et al, 1993b]. These models have the particular characteristic of incorporating adaptation and learning, such as the retino-tectal-pretectal neural circuitry modulated by learning processes responsible for habituation [Flores, 1997].
- At the detailed neural level, electrochemical neural mechanisms are studied to understand different neural phenomena, such as presynaptic inhibition in the control of synaptic selectivity [Eguibar *et al*, 1994]. These detailed neural model intends to provide refined neural mechanism where simplified ones are not enough, such as in gating networks [Jacobs *et al*, 1991].

While single level modeling involves by itself a great complexity, the grand challenge in this multi-level methodology is to integrate across the different modeling levels in order to explain phenomena which single levels cannot. From a system's standpoint, it is necessary to integrate between the different simulation and experimentation tools [Weitzenfeld *et al*, 1998c]. From a modeler's perspective, it is essential to comprehend the simulation's response, particularly challenging when involving highly complex models. Tools currently used to support this process include compiled languages for modeling, scripting languages for simulation, visual programming languages, graphical interfaces, visualization techniques, concurrent and distributed processing, numerical methods, analysis tools and simulation methodologies. Visualization plays a critical role both in synthesizing new models, using a top-down and bottom-up approach, and in analyzing the model's simulation results. Each analysis level involves its own complexity, requiring appropriate visualization techniques. Furthermore, there is the additional complexity of integrating across the different modeling levels.

2 Modeling

2.1 Autonomous Robotic Agents

Autonomous robotic agents can be either simulated in a virtual world or executed in the real world. The MissionLab architecture is specially suited for this task, since the model built needs only to be bound to the corresponding environment without any changes to the model itself. This is achieved by performing sensors and actuator binding to software or hardware devices, independent from model construction. MissionLab, as a simulation and execution system, incorporates graphical user interfaces, reusable software libraries, a simulation facility, and the capability to download executable robot code for a range of real mobile platforms, ranging from teams of small robots to human sized vehicles [Arkin and Balch, 1997], as shown in Figure 3.

Figure 3. Variety of real world autonomous robotic agents supported by the MissionLab system.

In terms of virtual worlds, Figure 4 shows an example of computational frog (*rana computatrix*) pursuing a prey (worm), interposed by a barrier.

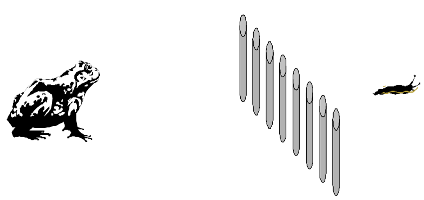

Figure 4. Computational frog in a prey and barrier set up.

2.2 Behaviors

Behaviors are generally described by ethogram, as the one shown in Figure 4, corresponding to the praying mantis' *Chantlitlaxia* [Cervantes *et al*, 1993a]. This conduct takes place when exploring the praying mantis explores its environment (when not mating, hunting, etc.).

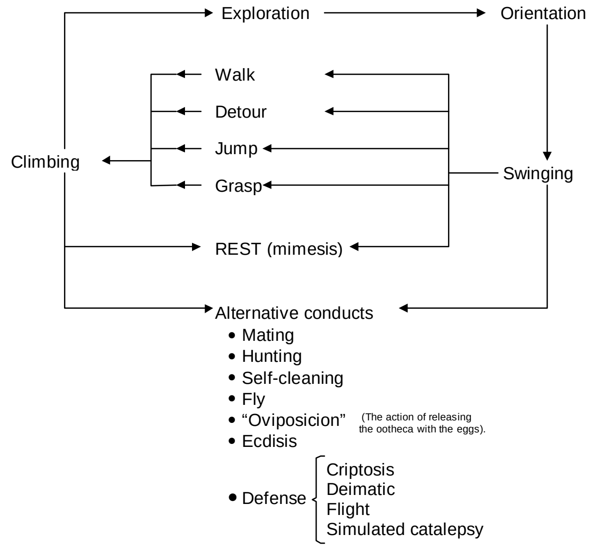

Figure 5. Praying Mantis' *Chantlitlaxia* Ethogram.

Another example of behavior is shown in Figure 6. These two figures are taken from the prey acquisition with detour model [Corbacho and Arbib, 1995], corresponding to Figure 4. When the barrier in front of the prey is wide and high enough, the agent (in this case the toad) will advance directly into the barrier in an attempt to catch the prey. After many trials learning takes place and the agent is able to detour directly around the barrier.

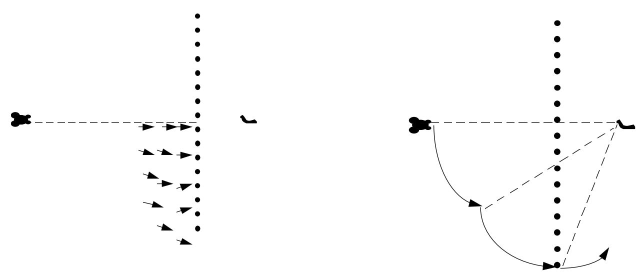

Figure 6. Left: Toad advances directly to middle of barrier, trying repeatedly to go through one of the gaps between adjacent posts. This continues until the toad reaches the edge of the barrier from where it advances directly to the prey. Right: After learning, toad advances to barrier's edge, avoiding hitting the barrier, and thus, successfully completing the detour behavior.

2.3 Schemas

Schemas are the primitive entities for modeling behaviors in autonomous robotic agents. In order to support complex adaptive behaviors, schemas define a hierarchical distributed model for action-perception control, where each schema incorporates its own structure and control mechanisms. Such a schema model is supported by MissionLab, with particular emphasis on autonomous robotic agents (virtual or real), and the more general Abstract Schema Language (ASL). ASL provides a multithreading distributed architecture for the execution of a large number of schemas described via compiled code, an interactive shell console and visualization tools [Calderas and Mármol, 1996]. In particular, ASL incorporates the ability to integrate with neural networks processing through its integration to the Neural Simulation Language NSL. Both in MissionLab and ASL, schemas correspond more to a specification language rather than to an explicit programming language. At the higher abstraction levels, the detailed schema implementation is left unspecified, only specifying what is to be achieved. At a lower level, schema are implemented, where different implementations may correspond to a single schema, in particular neural networks. The schema computational model is shown in Figure 7.

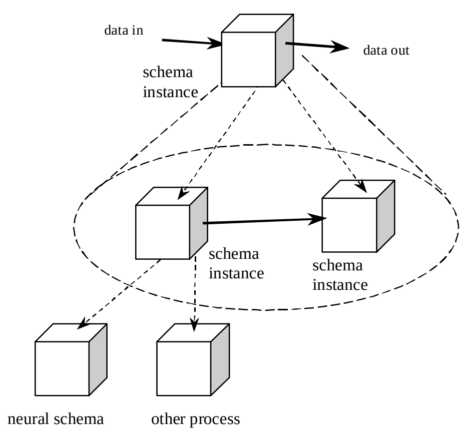

Figure 7. Schema hierarchical computational model.

At the top, a high level schema is shown decomposed into two lower level schemas. This hierarchical modeling is the basis for composition, known as schema *aggregates*, or *assemblages*. When at the same level, schemas are interconnected (solid arrows), or when at different levels, have their task delegated (dashed arrows). Schema interface consists of multiple unidirectional control/data, input and output ports, and a body where schema behavior is specified. Communication 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. When doing connections, output ports from one schema are connected to input ports from other schemas, and when doing relabelings, ports of similar type from schemas at different levels are linked to each other. The hierarchical port management methodology enables the development of distributed architectures where schemas may be designed and implemented independently and without prior knowledge of the complete model or their final execution environment, encouraging component reusability. This approach supports both top-down and bottom-up system design.

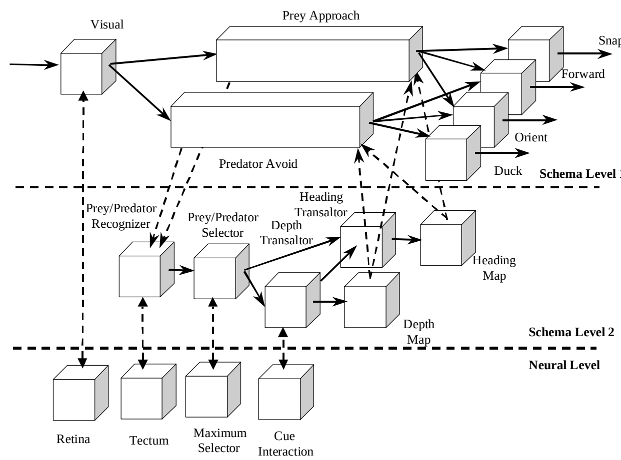

Figure 8. Schema model hierarchy for the toad's prey acquisition and predator avoidance behaviors.

Figure 8 shows the schema model hierarchy corresponding to the toad's prey acquisitions and prey avoidance model [Cobas and Arbib, 1992]. The highest level, schema level 1, describes the different behaviors being modeled, *prey approach* and *predator avoid*, and the perceptual and motor schemas, in this case, visual input and four types of motor action: forward, orient, snap and duck .. Tasks at this level are delegated to the next level down, schema level 2, where schemas perform more refined tasks. In this model, both the prey approach and the predator avoid schemas, delegate their tasks to a schema assemblage composed of a prey/predator recognizer, a prey/predator selector, depth and heading translators and maps. Next level down, the different neural networks implement the neural tasks by means of neural processing. Schemas delegating to neural processes are known as neural schemas. In particular the neural schemas in this model are implemented by a *Retina* [Teeters and Arbib, 1991], *Tectum* [Cervantes *at al*, 1985], *Maximum Selector* [Didday, 1976], and *Cue Interaction* [House, 1989] neural model.

Schema model complexity depends most importantly on the intrinsic complexity of the system being model. This complexity can be managed by modularizing the model into as many schemas and abstraction levels as desired. The key challenge for the modeler is to be able at all times to comprehend the complete model and its detailed components by being able to interact with them as a group as well as individually. The following set of figures is an example of the inherent complexity of some of these models. The model described in Figure 9 describes the highest schema level corresponding to the control of saccades in primate oculomotor behavior [Dominey and Arbib, 1992].

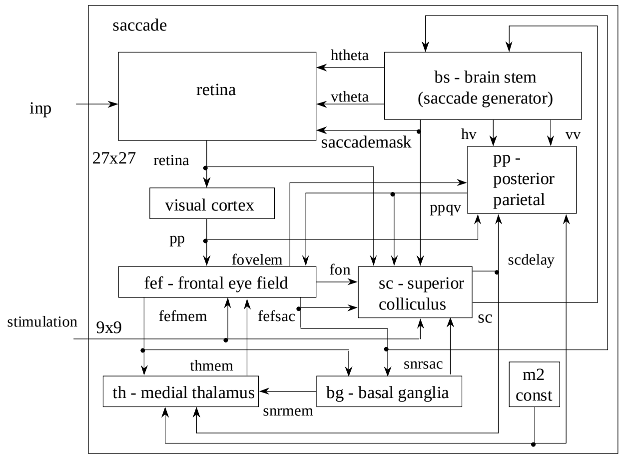

Figure 9. Schemas corresponding to multiple brain regions involved in sequential saccade generation: retina, brain stem, visual cortex, posterior parietal cortex, front eye fields, superior culliculus, medial thalamus, and basal ganglia.

Figure 10 describes more detailed schemas for the brain stem saccade generator.

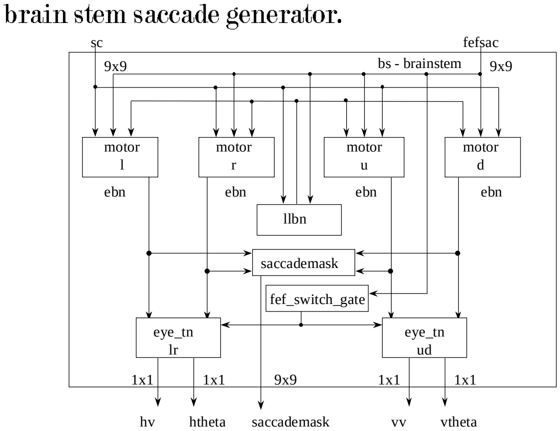

Figure 10. Schemas corresponding to brain stem saccade generator one level down: motor schemas for left, right, up and down eye movements, long lead burst neurons (llbn), frontal eye field gates (fef_switch_gate), and tonic neurons (eye_tn) to generate actual eye movement.

Figure 11 describes more detailed schemas corresponding to the motor schema belonging to the brain stem saccade generator, two levels down from the top schema level. Figure 12 describes schemas corresponding to the motor schema *trig_pause* belonging to the brain stem saccade generator, three levels down from the top schema level.

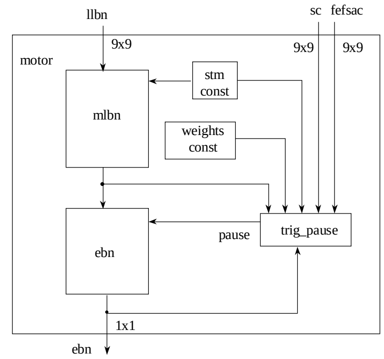

Figure 11. Schemas corresponding to each motor belonging to the brain stem saccade generator: excitatory burst neuron (ebn), motor lead burst neuron (mlbn), and a pause trigger (trig_pause). This diagram is two levels down from the top level.

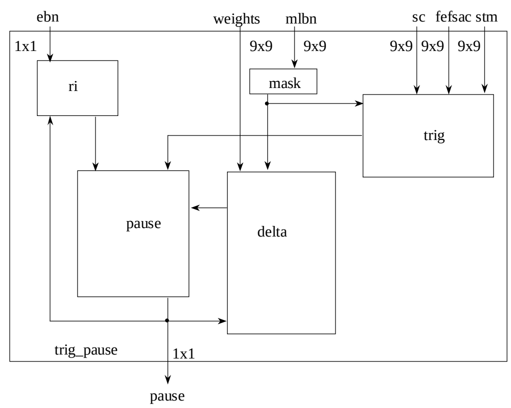

Figure 12. Schemas corresponding to pause trigger (trig_pause) in motor schema. This diagram is three levels down from the top level.

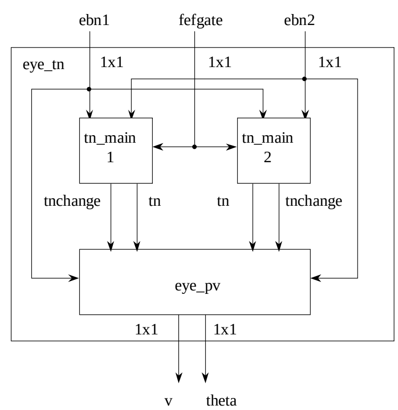

Figure 13. Schemas corresponding to brain stem saccade generator tonic neurons (eye_tn), two levels down from main schema level.

Figure 13 describes more detailed schemas corresponding to the tonic neuron schemas belonging to the brain stem saccade generator, three levels down from the top schema level. Figure 14 describes the main tonic neuron (tn_main) schemas, four levels down from the top schema level.

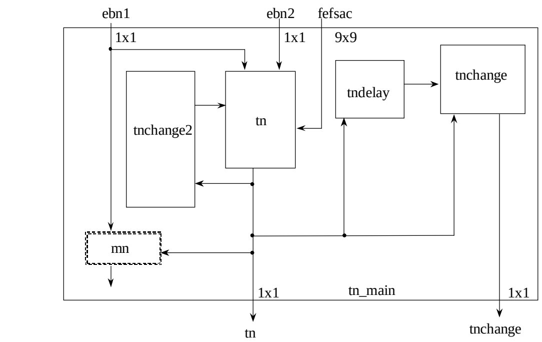

Figure 14. Schemas corresponding to brain stem saccade generator main tonic neurons (tn_main) from eye tonic neurons (eye_ tn), four levels down from main schema level.

Figure 15 describes more detailed schemas corresponding to the front eye field schema, while Figure 16 describes more detailed schemas corresponding to the superior culliculus.

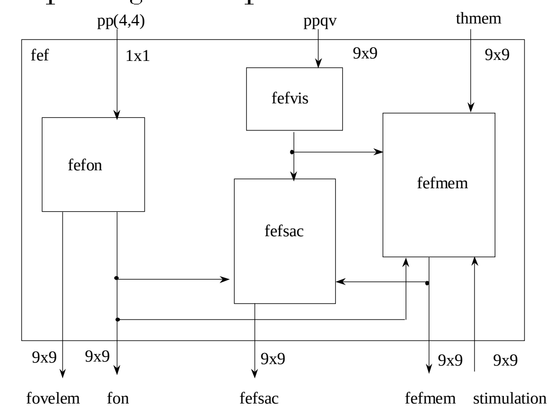

Figure 15. Schemas corresponding to front eye field (fef), one level down from main level.

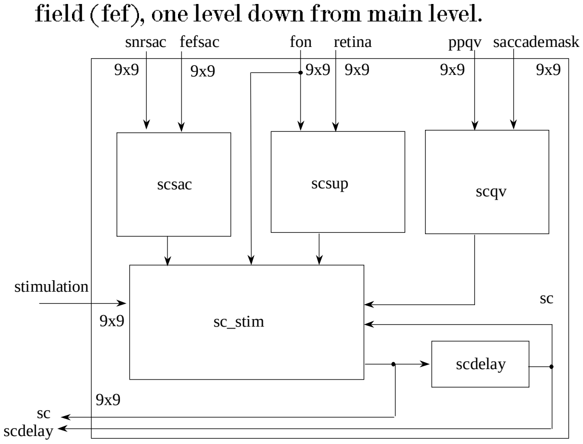

Figure 16. Schemas corresponding to the superior culliculus (sc), one level down from main schema level.

The previous diagrams are an example of some of the intrinsic complexity in schema modeling. The actual model for the control of saccades in primates is more complex. Complexity is much more significant when considering more behaviors and other brain regions [Arbib, et al, 1998].

2.4 Neural Networks

Neural schemas delegate their implementation to neural networks processing, as shown in Figure 17:

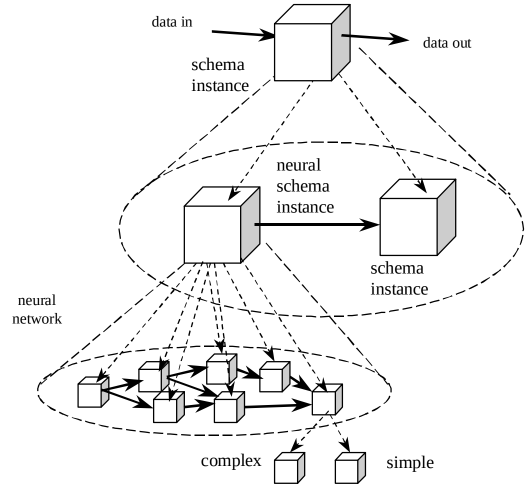

Figure 17. Neural schema hierarchy showing task delegation to neural networks processing.

At this level, neural networks are composed of interconnected neuron, each of which corresponds to a simple processing unit, having many inputs and a single output, as shown in Figure 18.

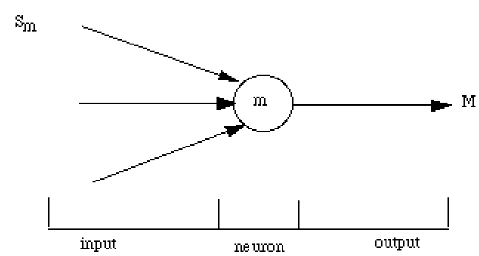

Figure 18. Simple neural element as basic component at the neural network level.

An example of a neural network is shown in Figure 19, corresponding to the maximum selector network [Didday, 1976].

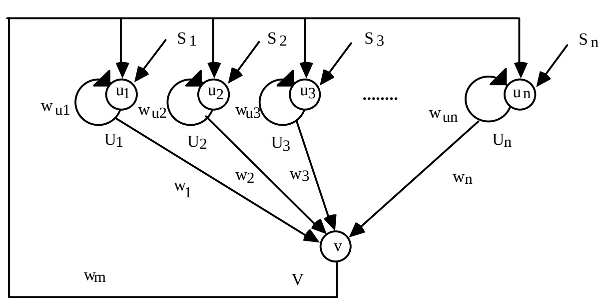

Figure 19 The neural network shown corresponds to the architecture of the Maximum Selector model, where u_i and v represent neural membrane potentials, U_i and V represent neural firing rates, S_i represent inputs to the network, and w_i represent connection weights.

A more complex neural network is shown in Figure 20, corresponding to the cue interaction model [House, 1989].

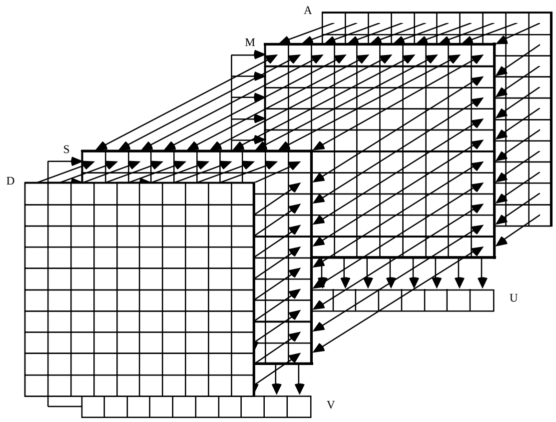

Figure 20. Neural Network for cue interaction model, layers S and M represent the disparity effects from binocular vision, and layers represent the accommodation effects from monocular vision.

The Neural Schema Language (NSL) provides the linkage to ASL by enabling the integration of neural networks processing. Models in NSL are described via a compiled language, where graphics displays and a scripting language provide the interfacing mechanisms between the model and the user. Two implementations of the system currently exist: NSLC in C++ and NSLJ in Java, where a major current thrust is to provide a library of models for direct simulation from the web [Weitzenfeld et al, 1998a].

2.5 Neurons

The neural schema model not only enables the incorporation of neural networks processing, but also provides an extended model where neurons themselves may have their task delegated by neural implementations of different levels of detail, from the very simple neuron models to the very complex ones [Weitzenfeld and Arbib, 1991].
Neuron models vary in their detail, depending of the particular mechanism simulated. At the highest level a neuron is a single cell with a very simple behavior, described , for example by the leaky integrator model [Arbib, 1989], composed of a soma (nucleus of the neuron), an axon (output of the neuron), and dendrites (input to the neuron). Connections between neurons take place through synapses from the axon of one neuron to the dendrites of another neuron. Synapses are the main mechanism for plasticity in neuron, and can be further refined into much more detail, as shown in Figure 21.

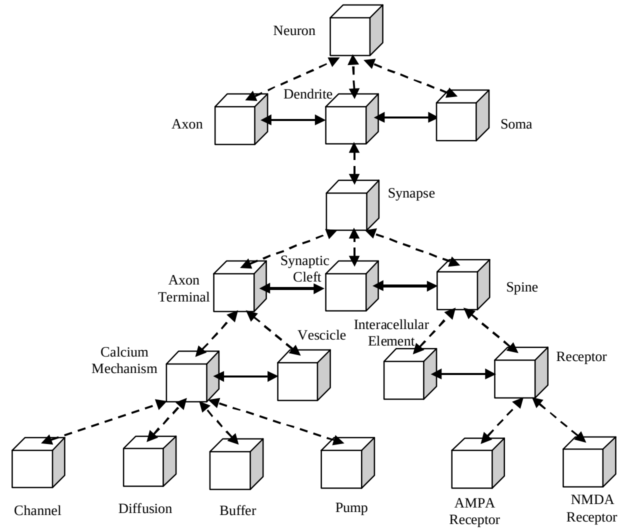

Figure 21. Neural modeling at different levels of details.

A number of models are used depending of the mechanisms simulated, such as the compartmental model, where a single axon is divided in compartments [Rall, 1959], and the ion kinetics model, where chemical concentration responsible for electric current is simulated [Hodgkin and Huxley 1952].

Figure 22 shows a detail neural model for the study of presynaptic inhibition in the selective control of neural pathways [Rodriguez, 1998].

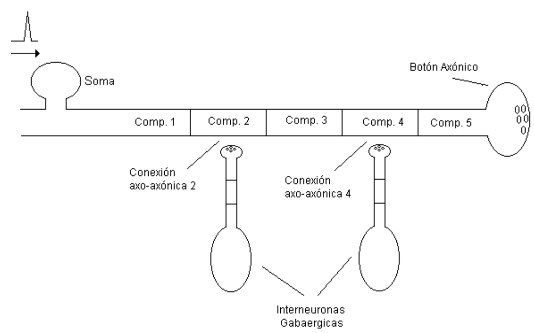

Figure 22. Detailed neural model of presynaptic inhibition for the control of neural pathways.

3 Simulation

Interpreting the output of complex models generating vast amounts of data requires appropriate visualization techniques applied to the different modeling levels and across levels.

3.1 Behavior

An autonomous robotic agent, simulated or embodied, senses its external environment and its internal state in order to produce behavior. While sensor input and motor output can be seen as data detail, appropriate behavior is analyzed by visualizing the animated agent's interaction with its virtual environment or the real robot when in the real world. Two dimension worlds are designed for simpler models, and three dimensions for the more complex ones. Simple output from the prey acquisition with detour model [Corbacho and Arbib, 1995] is shown in Figure 23 as seen from MissionLab's console.

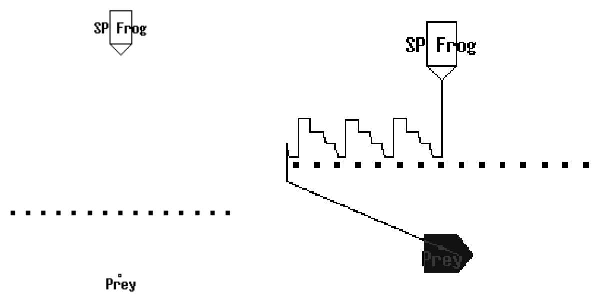

Figure 23. MissionLab console view of agent response to detour behavior.

One particular visualization technique for navigation paths is be means of vector fields describing attraction and repulsion between agents [Arkin, 1989].

3.2 Schemas

At the behavior level, perception and action define external interaction. In trying to understand why an agent behaved in a particular a manner, corresponding schema behavior has to be analyzed. Since schemas are recursive, input and output data analysis is required for each schema in a recursive fashion, together with data passed between connected schemas at the same level or between delegated schemas at different levels. Synchronization plays a key role in concurrent and distributed environments, affecting also visualization of data produced under different time constraints and granularities.

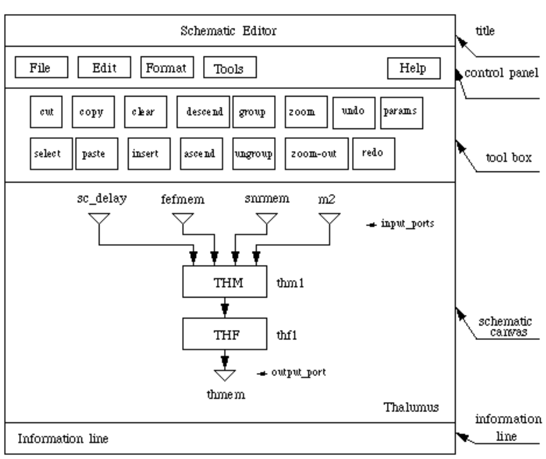

Figure 24. ASL's view of saccade's thalamus schema, with linkage between schema levels.

Figure 24 shows ASL's view of schema internals, taken from the saccade model [Dominey and Arbib, 1992].

Figure 25 shows MissionLab's view of schema network, taken from the prey acquisition with detour model [Corbacho and Arbib, 1995].

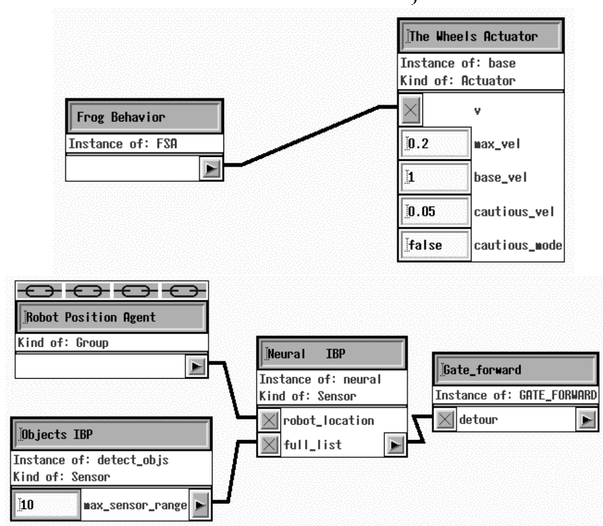

Figure 25. MissionLab's view of schema network for prey acquisition with detour model.

3.3 Neural Networks

At the neural network level, visualization takes the form of temporal and spatial graphs of various dimensions and forms, corresponding to neural input, output (firings) and membrane potentials. Time intervals play a major issue across multiple neural networks.

Figure 26 shows spatial and temporal output from the maximum selector model [Didday, 1976].

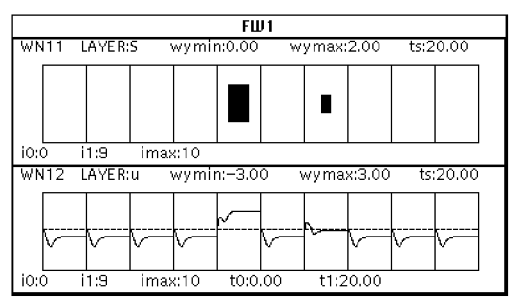

Figure 26. NSL spatial and temporal displays from maximum selector model.

Figure 27 shows two dimensional and three dimensional spatial output from the prey acquisition with detour model [Corbacho and Arbib, 1995].

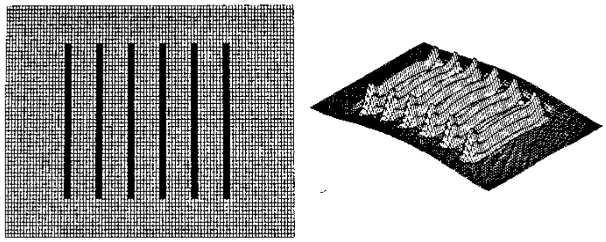

Figure 27. NSL two and three dimensional spatial displays from prey acquisition with detour model.

3.4 Neurons

At the detailed neural level, anatomical and electrical responses are the main visualization concern. Visualization takes the form of temporal graphs displaying electrical parameters, such as voltage and ionic concentrations. Time intervals are even finer.

Figure 28 shows sample temporal outputs taken from the presynaptic inhibition model for the selective control of neural pathways [Rodriguez, 1998].

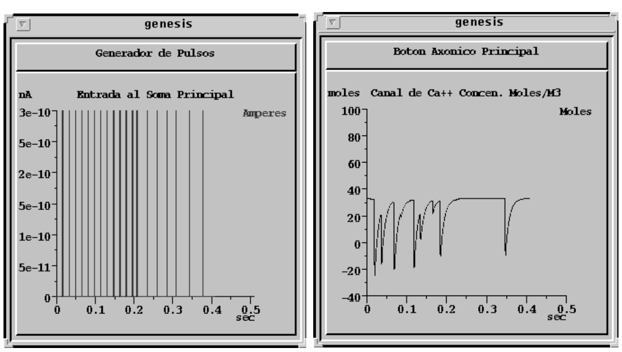

Figure 28. Genesis temporal outputs for action potentials and chemical concentrations.

4 Discussion

The work presented here shows the intrinsic complexity in modeling biological systems. This complexity can be managed by providing powerful tools for modeling and simulation. While existing visualization techniques are appropriate for simpler models at different modeling levels, more complex models incorporating different schema and neural levels, present a great challenge. A critical aspect in developing complex models is being able to analyze their results in a meaningful way. Workstation displays are very limited in the amount of information that can be displayed at once. More sophisticated visualization techniques are required to display greater amounts of data produced by the simulation at each time step. At the moment, model and simulation visualization is rather limited, limiting the scope of models. A future goal is to employ immersion systems, where the modeler can visualize and control more aspects of a simulated environment at once. Furthermore, a virtual reality approach would provide a way for the modeler to take part in the actual experiment by virtually interacting with all other virtual autonomous robotic agents in the simulated world.

References

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

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

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

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

[Arkin, 1989] Arkin, R.C., Motor Schema-based Mobile Robot Navigation, *Int. J. of Robotics Research*, 8,4:92-112, August, 1989.

[Arkin and Balch, 1997] Arkin, R.C. and Balch, T., AuRA: Principles and Practice in Review, *Journal of Experimental and Theoretical Artificial Intelligence*, 9:2-3, pp. 175-189, April-Sept. 1997.

[Arkin *et al*, 1997] Arkin, R.C., Cervantes-Perez, F., and Weitzenfeld, A., Ecological Robotics: A Schema-Theoretic Approach, *Intelligent Robots: Sensing, Modelling and Planning*, Editors R.C.Bolles, H.Bunke, and H.Noltemeier, World Scientific, 1997.

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

[Bower and Beeman, 1994] Bower, J.M., and Beeman, D., *The Book of GENESIS, Exploring Realistic Neural Models with the GEneral NEural SImulation System*, Telos, Springer-Verlag, 1994.

[Calderas and Mármol, 1996] Calderas, C., and Mármol, S., Diseño e Implementación de un Sistema Distribuido Heterogéneo para la Extensión del Lenguaje de Modelos Teórico Esquemáticos ASL, ITAM Undergraduate Thesis, CANNES-TR-96-01, ITAM, México, Octubre 1996.

[Cervantes *et al*, 1985] 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.

[Cervantes *et al*, 1993a] Cervantes-Perez, F., Franco, A., Velazquez, S., Lara, N., 1993, 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.

[Cervantes *et al*, 1993b] 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. Rudoman, M.A. Arbib, F. Cervantes-Perez, and R. Romo, Springer Verlag Research Notes in Neural Computing vol 4, pp. 426-449, 1993.

[Cliff, 1992] 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 Artifical Life (ECAL 91)*, Editors, F.J., Varela and P. Bourgine, MIT Press, pp 78-87, 1992.

[Cobas and Arbib, 1992] 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.

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

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

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

[Eguibar *et al*, 1994] Eguibar, J.R., Quevedo, J., Jiménez, I., Rudomin, P., "Selective cortical control of information flow through different intraspinal collaterals of the same muscle afferent fiber" *Brain Res.* 643: 328-333, 1994.

[Flores, 1997] Flores, L.R., Modulación de Tiempo variable de la Elicitación de la Conducta de Captura de Presas en los Anfibios Anuros, UNAM Master Thesis, CANNES-TR-97-01, ITAM, Mexico, DF, Agosto 1997.

[Hines, 1994] Hines, M., The NEURON Simulation Program, *Neural Network Simulation Environments*, Ed. Josef Skrzypek, Kluwer, 1994.

[Hogkin and Huxley, 1952] 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.

[House, 1989] House, D., Depth perception in frogs and toads: A study of in neural computing, *in Lecture notes in Biomathematics*, 80, Springer-Verlag, 1989.

[Jacobs *et al*, 1991] Jacobs, R.A. Jordan, M., Nowlan, S.J., Hinto, G.E., Adaptive Mixtures of local experts, *Neural Computation*, 3:1, 1991.

[MacKenzie *et al*, 1997] MacKenzie, D., Arkin, R.C., and Cameron, R., Multiagent Mission Specification and Execution, in *Autonomous Robots*, 4:1, pp.29-52, Jan.1997. (Also appears in *Robot Colonies*, ed. R.Arkin and G.Bekey, Kluwer Academic Publishers, 1997.)

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

[Rodriguez, 1998] Rodriguez, G., Modelado y Análisis del Fenómeno de Inhibición Presináptica, ITAM Undergraduate Thesis, CANNES-TR, ITAM, México, 1998 (in preparation).

[Teeters and Arbib, 1991] 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.

[Weitzenfeld, 1993] 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.

[Weitzenfeld and Arbib, 1991] 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.

[Weitzenfeld and Arbib, 1994] Weitzenfeld, A., Arbib, M.A., NSL,Neural Simulation Language, in *Neural Networks Simulation Environments*, Editor J. Skrzypek, Kluwer, 1994.

[Weitzenfeld at al, 1998a] Weitzenfeld, A., Arbib, M., Alexander, A., *NSL - Neural Simulation Language: System and Applications*, MIT Press, 1998 (in preparation).

[Weitzenfeld *et al*, 1998b] Weitzenfeld, A., Arbib, M., Cervantes, F., Rudomin, P., Multi-level Simulation Methodology: A Computational and Experimental Approach to Neural Systems, *Proceedings of NSF Design and Manufacturing Grantees Conference*, Monterrey, Mexico, 651A-652A, 1998.

[Weitzenfeld *et al*, 1998c] Weitzenfeld, A., Arkin, R., Cervantes, F., Olivares, R., and Corbacho, F., A Neural Schema Architecture for Autonomous Robots, Iberamerican Conference on Artificial Intelligence IBERAMIA, Lisbon, Portugal, 1998 (submitted).