



HAL
open science

Neuromimetic ICs with analog cores: an alternative for simulating spiking neural networks

Sylvie Renaud, Jean Tomas, Yannick Bornat, Adel Daouzli, Sylvain Saïghi

► **To cite this version:**

Sylvie Renaud, Jean Tomas, Yannick Bornat, Adel Daouzli, Sylvain Saïghi. Neuromimetic ICs with analog cores: an alternative for simulating spiking neural networks. ISCAS'07, May 2007, New-Orleans, United States. pp.3355-3358. hal-00161313

HAL Id: hal-00161313

<https://hal.science/hal-00161313>

Submitted on 10 Jul 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Neuromimetic ICs with analog cores: an alternative for simulating spiking neural networks

Sylvie Renaud, Jean Tomas, Yannick Bornat, Adel Daouzli and Sylvain Saïghi

IMS Laboratory
University Bordeaux 1 - CNRS / ENSEIRB
Talence, France
sylvie.renaud@ims-bordeaux.fr

Abstract - This paper aims at discussing the implementation of simulation systems for SNN based on analog computation cores (neuromimetic ICs). Such systems are an alternative to completely digital solutions for the simulation of spiking neurons or neural networks. Design principles for the neuromimetic ICs and the hosting systems are presented together with their features and performances. We summarize the existing architectures and neuron models used in such systems, when configured as stand-alone tools for simulating ANN or together with a neurophysiology set-up to study hybrid living artificial neural networks. As a primary illustration, we present results from one of the platforms: hardware simulations of single neurons and adaptive neural networks modeled using the Hodgkin-Huxley formalism for point neurons and spike-timing dependent plasticity algorithms for the network adaptation. Additional examples are detailed in the other papers of the session.

I. INTRODUCTION

Neuromorphic engineering is a growing research domain that merges insights from neurobiology, computer science and IC engineering. A new generation of artificial neural networks is emerging, far from the classical sequential ANN (Artificial Neural Networks) machines such as Perceptrons. These novel ANN are based on computational elements qualified as “neuromimetic”, that reproduce the neural activity of spiking neural networks (SNN) with a high level of precision, as models are based on biophysics properties and parameters provided by biologists [1]. Apart from the classical software computation, some designers of those SNN exploit the analogy between electronics and biology physics; the signals are computed in analog mode, and are as a consequence available as continuous variables, both in time and in value. Analog computation is very competitive when considering the integrated circuit density, but has a complex design flow. The electrical activity of neural elements can be emulated at different degrees of complexity depending on the implemented model. The dynamics of their activity can be either reproduced to ensure real-time computation, or accelerated for faster computation. In the case of real-time computation, it is possible to construct mixed living-artificial networks, where the silicon neurons are interconnected with the biological cells to form “hybrid networks” [2].

These features can be variously combined, and result in systems optimized to address specific SNN issues. We propose to present in this session the design principles and examples of different analog-based ICs and systems that model neural networks. They compute different neuron models at different time scales, and eventually allow connections to real neurons. The systems distribution of computation between analog and digital circuitry are also different. Some systems allow adaptive synaptic connections in the emulated neural networks, by computing plasticity rules such as STDP (Spike-Timing Dependent Plasticity). The five papers in the session will illustrate a range of applications, and show how these ANN can be used in neurophysiology experiments, for computational neuroscience issues, or to study computation paradigms.

II. COMPUTATION OF SPIKING NEURAL NETWORKS

A. Analog or Digital spiking neural networks?

Spiking neural networks (SNN), closely inspired from biology, are now recognized to be an essential simulation tool to study information processing by the brain. They are based on models that explicitly describe the dynamics of a network and take into account the timing of inputs. Information processing in such networks is continuous; the precision of the dynamics description depends on the choice of the neuron model and of the connectivity description. A designer intending to implement a computational support for the neuron (or compartment) model has different material solutions: software, digital hardware, analog hardware.

While the choice between software and digital hardware is often a matter of technical constraints (specially with the generalization of FPGAs), clear discussion arguments emerge when comparing the implementation of neuron models using an analog or a digital solution [3]:

- analog computation intrinsically provides a continuous computation of continuous variables, like in SNN. As a consequence, the simulation time scale can be fixed easily and precisely (real-time or accelerated).
- analog SNN can present a high integration density, as one wire encodes one signal (instead of N wires to represent a N-bits signal), and as one can exploit the primitives of

Research is supported by CNRS, E.U. Grant SenseMaker (FP5-IST-2001-34712) and E.U. Grant FACETS (FP6-IST-FETPI-2004-15879).

computation from the physics of electronic elements as mathematical functions.

- transistors mismatch in analog design can emulate the neurons diversity.
- digital design presents a much lower design cost, including a better time-to-market performance, and an easy reconfigurability.
- design re-use is easier for digital circuits, although it is possible to build libraries of analog IPs (see section IV).

Another important point is the capability of the model to cover the variety of features, which appear in the activity of the addressed biological neurons (see a review for cortical neurons in [4]). As described in [5], a digital implementation of SNN will finally be a compromise between the computational cost and the model richness. The digital solution presents clear limitations for complex biophysical models or for very large networks. We will see in II-B why such models are well-adapted to an analog implementation.

B. Models for analog neuromimetic ICs

Neuromimetic analog ICs exploit the intrinsic current-voltage relationships of active and passive electronic elements to compute the mathematical functions present in the neural element models. Models of spiking neurons, that reproduce the electrical activity of a neural element (dendrite, soma, compartment, point neuron, synapse) are particularly well suited to such implementations, as they rely on the description of the dynamics of ionic or synaptic currents, and of passive circuitry for dendritic trees [1]. These models represent each neural element by its equivalent to an electrical circuit, replicated in the neuromimetic analog circuit. A series of models can be addressed using that approach: between the complex Hodgkin-Huxley (HH) formalism and the simple Integrate-and-Fire (IF) model, many intermediate description levels are possible [1], [5]. Some, like the FitzHugh-Nagumo (FN) model, were specifically designed to optimize an electronic implementation [6]. We can also mention behavioral models, designed to fit the activity waveforms of a spiking neuron [7].

When considering the dynamics of spiking neural networks, and the formation of connectivity patterns, another feature appears to be crucial: the plasticity [1]. Synaptic plasticity appears to be the primary substrate of long-term learning and memory, but non-synaptic plasticity (ie plasticity mechanism on biophysics parameters of ionic conductances) may also play a role. Neuromimetic devices also compute plasticity mechanisms when they are dedicated to networks simulation. The Spike-Timing Dependent Plasticity (STDP) algorithms applied to synaptic connections are the most commonly used models [8], [9]; they induce an important computational cost, and necessitate a dynamic control of the connectivity.

C. Issues in analog SNN

Figure 1 indicates the actual trends in the development of analog-based systems for simulating SNN. As an illustration, we positioned on the graph the systems described in this

session (see IV, [10], [11], [12], [13]). The three axes represent: (x) The model complexity, strongly linked to its biological relevance. We note here the difference between point neuron models, where a neuron is a single computational element, and compartmental models that take into account the cells morphology. (z) The size of the network, evaluated by its number of neurons. (y) The simulation time scale referenced to the biological time scale; 10^0 on this axis is real-time. The white zone in this graph corresponds to analog-based systems existing or under development.

The first point we observe in that performance graph is the limitation of the network size for detailed models: conductance-based models like Hodgkin-Huxley models, or compartmental models present a high silicon cost for a more accurate computation, that limits the number of neurons on a single chip. Another limitation is due to the connectivity issues for large networks of IF models, that generally include dynamic STDP. The connectivity question is generally addressed by using event-based protocols to transmit the information in the network, using a digital interface.

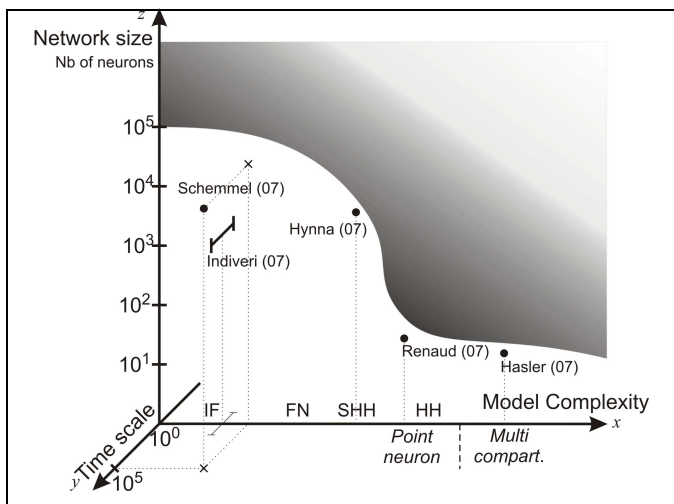


Figure 1. Performances of analog-based systems of SNN. Some systems (as Indiveri(07) [13]) process spikes on an event-based basis at a constant time scale (fixed for a given experiment).

III. IMPLEMENTATION SOLUTIONS

Since the first “Silicon Neuron” by M. Mahowald [14], different groups developed hardware simulations systems based on analog or mixed neuromimetic circuits. The simulated neural element can be either integrated on a single chip or distributed on multiple chips, due to integration constraints. Figure 2 presents a non exhaustive list of neuromimetic systems with analog cores, developed in the last 15 years ([2], [10] to [19]). Those systems compute various models of SNN, and can eventually be used for hybrid networks experiments. An interesting point is the emergence of mixed system, in which computation is shared between analog and digital elements. The digital hardware is in such case in charge of the connectivity computation and/or control. As mentioned earlier, the cost of connectivity increases following a quadratic law with the network size. More and

more neuromimetic systems will present such mixed A/D architectures.

The analog ICs modeling SNN implement the models described in II-B (IF, FN, HH). We labeled SHH the HH-inspired models where some of the conductance functions are simplified or fitted, opposed to the complete Hodgkin-Huxley description of voltage and time-dependent ionic conductances

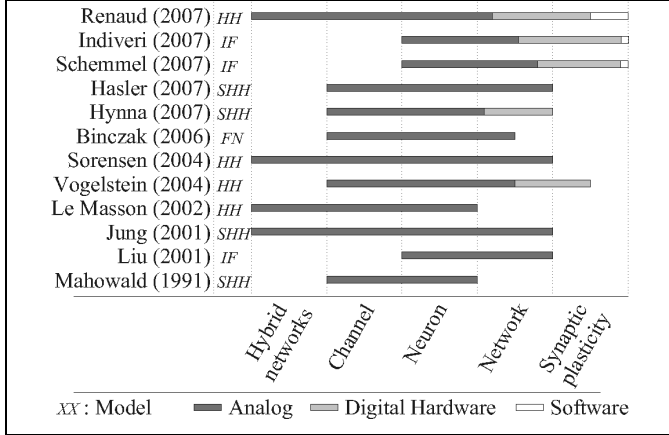


Figure 2. Computation distribution in various SNN analog-based systems.

IV. AN ANALOG-BASED SIMULATION SYSTEM

We present in that section a mixed A/D platform we developed to emulate SNN. In this system, ICs emulate in real-time and analog mode conductance-based models of neurons and synapses; synaptic interactions, subject to short-term and long-term mechanisms, are digitally processed using both hardware and software material. The whole simulation system is organized in 3 layers: software, digital hardware and analog. The analog hardware layer runs the continuous and real-time computation of the neurons and synapses ionic currents. The analog ICs are controlled by the digital hardware layer. This hardware is in charge of computing spike events information from the analog neurons, and of controlling the synaptic connectivity back to the analog hardware. Although the processing mode is globally event-based to optimize computational speed, the spikes are time-stamped to ensure real-time at the event level. The upper layer includes the software driver and interface, in charge of controlling the bi-directional data transfer to the software via a PCI bus. Finally, a PC running a real-time operating system hosts software functions to eventually compute the connectivity dynamics functions in the neural network. The software also includes user interface functions to control the off-line and on-line simulation configuration.

The neuromimetic ICs were designed as application specific ICs (ASICs) to compute in analog mode conductance-based models based on the Hodgkin-Huxley formalism discussed in section II. Individual neurons produce action potentials that express their intrinsic dynamic properties, as well as their response to stimulations (stimulations are currents, flowing from synapses or from other sources). The full custom ASICs are designed using a library of electronic functions we developed previously [20] in order to optimize design re-use. Each mathematical function appearing in the

neuron model corresponds to an analog module in the library, with inputs for tunable parameters, stored on-chip in analog memory cells. The modules are arranged to generate electronic currents that emulate ionic or synaptic currents (figure 3). The parameters are also used to compensate intrinsic diversity of ICs. Finally, the computational cores of the ASICs are configured to fit different types of ionic channels and form the targeted neuron model.

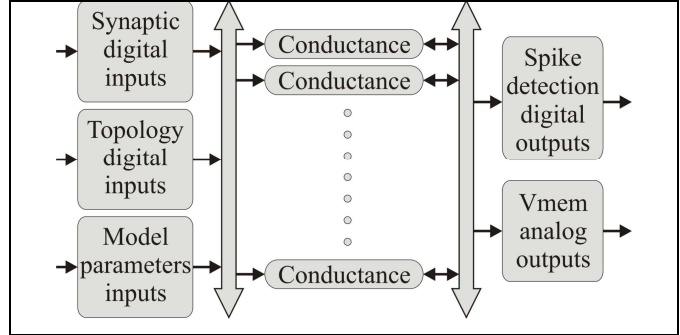


Figure 3. Architecture of the neuromimetic ICs : a set of conductances and functions are configured by digital layers to perform versatile types of neural models following the HH formalism.

While the neuron activity is computed in analog mode (membrane voltage, presenting continuous action potentials), a 1-bit digital representation is also available; conversion is done using a threshold comparator. The digital outputs transmit the spikes ‘events’, computed by the digital layers of the system. They are used for the calculation of the synaptic interactions that take into account the spikes exact timing (according to a 2 MHz timer).

The models for synaptic interactions are pulse-based kinetic models of synaptic conductances [20]. The dynamics of the synaptic channels are captured using a two-state (open-closed) scheme. This model easily handles phenomenon like summation or saturation, and describes precisely the time course of the synaptic interactions. Furthermore, this model is easily integrated on hardware in the form of multi-synaptic elements. Pre-synaptic events are gathered on a single wire to generate the synaptic pulses, applied to the post-synaptic neuron. Each pulse triggers the transition to the opening state of the synaptic channels. The pulse width is modulated to encode the stimulation strength and is dynamically updated by the digital layers, according to the plasticity algorithms. The analog signals (membrane voltage, ionic current) are also exploited in experimental configurations, such as ‘hybrid networks’; in that case, dedicated on-chip conductances are used as artificial synapses that communicate in real-time with in vitro biological neurons [2].

The next figures present experimental results obtained using the presented platform. Figure 4 is a plot of a neuron membrane voltage, available as a system output. The implemented model is a 5-conductance model (sodium, potassium, calcium, calcium-dependent potassium and leak channels) and presents a spiking activity with frequency adaptation due to the calcium-dependent potassium channel. Figure 5 is a simulation of an adaptive network, organized as a reciprocal inhibition network with 6 neurons. This SNN

produces an alternated activity, as in locomotion mechanisms. At the beginning of the simulation, we randomize the weights of the synapses; E1 and E2 are tuned to oscillate around 10 Hz. A STDP algorithm [1] is dynamically processed on all synapses. After a few seconds of real time simulation, the network converges to the expected result: (E1, M1) and (E2, M2) respectively show a biphasic activity pattern. Excitatory synapses have converged to the maximum weight, while inhibitory synapses stabilize to medium values.

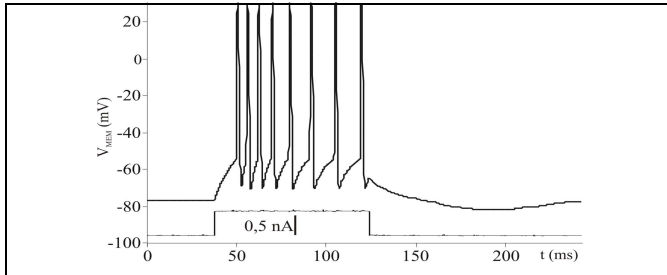


Figure 4. A single neuron real-time simulation with biological equivalent scales. A) Neuron membrane voltage. B) Stimulation current

These 2 examples prove the functionality of this configurable simulation system for neuromimetic SNN. The analog architecture ensures real-time processing whatever the network size. This size is limited by the complexity of the model implemented on the ICs, but this system is powerful enough to be used as a tool for computational neuroscience. Series of experiments addressing biologically-significant questions are currently in progress.

V. CONCLUSION

We have presented the design principles for neuromimetic analog ICs and the associated systems simulating SNN. The aim of such systems is not to compete with classical software simulators, but rather to offer alternative solutions with identified strong points: a constant computational speed due to parallel processing features of analog ICs, the easy connectivity of hardware-based SNN with hardware or living sensors and actuators, and the strong link with the demanding research in embedded biomedical devices. This interdisciplinary research topic will clearly go on increasing its activity, as the applicative field expands together with the hardware capacities.

REFERENCES

- [1] W. Gerstner, W. M. Kistler, *Spiking Neuron Models. Single Neurons, Populations, Plasticity*, Cambridge University Press, 2002.
- [2] G. LeMasson, S. Renaud, D. Debay, T. Bal, "Feedback inhibition controls spike transfer in hybrid thalamic circuits", *Nature*, vol. 4178, pp. 854-858, 2002.
- [3] R. Sarpeshkar, "Analog versus digital: extrapolating from electronics to neurobiology", *Neural Computation*, vol.1, pp. 1601-1638, 1998.
- [4] B.W. Connors, M.J. Gutnick, "Intrinsic firing patterns of diverse neocortical neurons", *Trends Neurosci.*, vol. 13, pp. 99-104, 1990.
- [5] E. Izhikevich, "Which model to use for cortical spiking neurons", *IEEE Trans. on Neural Networks*, vol.15, pp.1063-1070, 2004.
- [6] R. FitzHugh, "Impulses and physiological states in models of nerve membrane", *Biophys. Journal*, vol. 1, pp. 445-466, 1961.

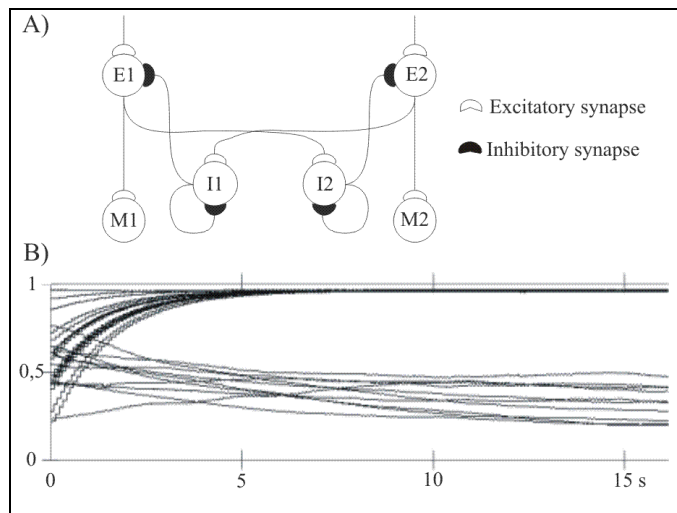


Figure 5. An adaptive network simulation. A) Network description. E1 and E2 are excitatory neurons, I1 and I2 are inhibitory neurons, M1 and M2 are motoneurons. B) Evolution of the synaptic weights due to the STDP algorithm (5 trials), 15 seconds of simulation. The Y axis is the synapse weight (0 = null weight, 1=max. weight). The stronger weights are the ones of excitatory synapses while inhibitory synapses have intermediate weights.

- [7] E. Farquhar, P. Hasler, "A bio-physically inspired silicon neuron", *IEEE Trans. on Circuits and Systems I*, vol. 52, pp. 477-488, 2005.
- [8] S. Fusi, M. Annunziato, D. Badoni, A. Salamon, D.J. Amit, "Spike-driven synaptic plasticity: theory, simulation, VLSI implementation", *Neural Computation*, vol. 12, pp.2227-2258, 2000.
- [9] M. Badoual, Q. Zou, A. Davison, M. Rudolph, T. Bal, Y. Frégnac, A. Destexhe, "Biophysical and phenomenological models of multiple spike interactions in spike-timing dependent plasticity", *International Journal of Neural Systems*, vol. 16, pp.79-97, 2006.
- [10] P. Hasler, S. Koziol, E. Farquhar, "Transistor channel dendrites implement HMM classifiers", *ISCAS'2007*, accepted.
- [11] K.M. Hynna, K. Boahen, "Silicon neurons that burst when primed", *ISCAS'2007*, accepted.
- [12] J. Schemmel, D. Brüderle, A. Grübl, K. Meier, E. Müller, "Modeling synaptic plasticity within networks of highly accelerated I&F neurons", *ISCAS'2007*, accepted.
- [13] G. Indiveri, S. Mitra, S. Fusi, "Spike-based VLSI learning circuits for classifying complex patterns in networks of integrate-and-fire neurons", *ISCAS'2007*, accepted.
- [14] M. Mahowald, R. Douglas, "A silicon neuron", *Nature*, vol.354, pp. 515-518, 1991.
- [15] S.-C. Liu, J. Kramer, G. Indiveri, T. Delbrück, T. Burg, R. Douglas, "Orientation-selective aVLSI spiking neurons", *Neural networks*, vol. 14, pp. 629-643, 2001.
- [16] R. Jung, E. J. Brauer, J. J. Abbas "Real-time interaction between a neuromorphic electronic circuit and the spinal cord", *IEEE Trans. on Neural Systems and Rehab. Eng.*, vol. 9(3), pp.319-326, 2001.
- [17] R.J. Vogelstein, U. Malik, G. Cauwenberghs, "Silicon spike-based synaptic array and address-event transceiver", *Proceedings of ISCAS'04*, vol.5, pp.385-388, 2004.
- [18] M. Sorensen, S. DeWeerth, G. Cymbalyuk, R.L. Calabrese, "Using a hybrid neural system to reveal regulation of neuronal network activity by an intrinsic current", *J. Neurosci.*, vol.24, pp. 5427-5438, 2004.
- [19] S. Binczak, S. Jacquir, J.-M. Bilbault, V. B. Kazantsev, V. I. Nekorkin, "Experimental study of electrical FitzHugh-Nagumo neurons with modified excitability", *Neural Networks*, vol.19, pp.684-693, 2006.
- [20] Y. Bornat, J. Tomas, S. Saighi, S. Renaud, "BiCMOS Analog Integrated Circuits for Embedded Spiking Neural Networks", *DCIS 2005*, Lisbon, Portugal, 2005.