Going Small: Using the Insect Brain as a Model System for Edge Processing Applications

Angel Yanguas-Gil

Energy Systems Division, Argonne National Laboratory
Lemont, Illinois
ayg@anl.gov

ABSTRACT

In this work I explore bio-inspired architectures for adaptive and smart sensing incorporating two key aspects present on the insect brain that are not found in more traditional neural network approaches: modulated, hierarchical processing and modulated learning. Our architecture incorporates two central ideas: 1) a state-dependent processing of inputs that can be triggered internally or externally, and 2) state-dependent online learning capabilities, in this specific case allowing the system to change the valence associated to different types of input. These ideas are explored through a hybrid design in which information is processed through a spiking neural network, while a recurrent non-spiking component provides the modulatory feedback to the system. The proposed approach exemplifies how neuromorphic computing approaches naturally integrate sensing and processing within a single functional unit. The proposed architecture can be implemented using conventional VLSI processing, though the integration of novel materials can help simplify its implementation.

CCS CONCEPTS

• **Theory of computation** → **Reinforcement learning**; • **Computer systems organization** → **Neural networks**; • **Computing methodologies** → *Logical and relational learning*; • **Hardware** → *Sensor devices and platforms; Analog and mixed-signal circuits; VLSI design manufacturing considerations*;

KEYWORDS

neuromorphic computing; spiking neural networks; online learning; smart sensors; neuromodulation; insect brain

ACM Reference Format:

Angel Yanguas-Gil. 2018. Going Small: Using the Insect Brain as a Model System for Edge Processing Applications. In *Proceedings of ACM Great Lakes Symposium on VLSI (GLSVLSI)*. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3194554.3194610>

Publication rights licensed to ACM. ACM acknowledges that this contribution was authored or co-authored by an employee, contractor or affiliate of the United States government. As such, the United States Government retains a nonexclusive, royalty-free right to publish or reproduce this article, or to allow others to do so, for Government purposes only.

GLSVLSI '18, May 23–25, 2018, Chicago, IL, USA

© 2018 Copyright is held by the owner/author(s). Publication rights licensed to ACM.

ACM 978-1-4503-5724-1/18/05...\$15.00
<https://doi.org/10.1145/3194554.3194610>

1 INTRODUCTION

Insects can be viewed as complex, smart sensors able to carry out multiparallel input processing and integration. For instance, the honeybee has 60,000 receptors just in its olfactory system. With an increasingly large body of research consistent with the presence of a complex internal state, [5, 9] they also exhibit remarkable learning capabilities and consume a small amount of power, with a small number of neurons. These properties makes them a fantastic example of how to achieve complex functionality through limited resources. Understanding the design principles of the insect brain and how they can be translated into hardware can therefore be valuable for edge processing and the design of smart sensors that are broadly trained to operate on a changing environment.

An increasing body of research is supporting the idea that the connectome is just one aspect of what provides the functionality to central neural systems: other types of interactions, both through local modulatory interactions, and through volume release of neurotransmitters and neuropeptides, play a central role in modifying the behavior and activity of neurons.[3, 9, 12] A second important role of modulatory signals is regulating online and reinforcement learning, where synaptic weights are reinforced only in presence of a trigger signal.[2]

These ideas can have important implications in how we can design edge processing architectures inspired on neuromorphic systems: from a mathematical point of view, the implication is that rather than a functional transformation between input and output, neural systems could behave as time-dependent *programmable networks*, where an internal state of the system, in the form of local and volume modulatory interactions, controls the the transformation between input and output. This provides an interesting avenue to explore the extension of the concept of programming to neuromorphic architectures, where a sequence of inputs allow us to dynamically control the way information is processed. These principles can also lead to simplified, smaller designs, for instance through the combination of modulation and binary synaptic weights, or by providing a simple way of reusing substructures to perform different tasks. As an example, the network shown in Figure 1 establishes a connectivity pattern for carrying out AND, XOR, and OR operations on the same network based on the modulation of its different neurons.

The olfactory systems of invertebrates is an interesting example of an architecture that takes advantage of modulatory interactions to process information. In Figure 2 we show a scheme of the architecture of the olfactory system of an invertebrate. The antennal lobe acts as the entry level structure of information processing. It is composed of a set of discrete units called glomeruli formed by the synapses between olfactory receptive neurons (ORN) and relay

Figure 1: How modulation can drive functionality: a single neural network (left) shows three distinct functionalities when neurons are externally modulated

Figure 2: Summary of the early stages of the olfactory system of a higher vertebrate. The parallel processing path to the lateral horn has been removed for simplicity

neurons called projection neurons that take the information into two higher level systems, the mushroom body and the lateral horn (not shown in Figure 2). The antennal lobe is the site of *stateful inhibition*: a set of lateral neurons, which are inhibitory in character, modulate and control the way this information is relayed.[4] The activity of these neurons are themselves modulated through a series of modulatory inputs. The lateral horn has been linked to innate responses, whereas the mushroom body is the site of olfactory learning.[1]

In this work I focus on architectures that incorporate this type of modulation to control information processing. We have implemented heterogeneous architectures involving a fast, spiking component and a slower, modulatory component responsible for two types of modulatory interactions: neuron-level modulation, and neuron-synaptic weight interactions. Our ultimate goal is to understand how to build systems that show adaptive behavior and are broadly trained to perform on a changing environment, and the impact that design choices made at a design stage have on the performance and limitation of such devices.

2 MODEL

The focus of our work are heterogeneous architectures formed by two coupled systems, one responding at a fast scale, in this case implemented as spiking neurons, and a second one reacting at a

slower timescale, modulating the performance of the network at the neuron and/or synaptic weight levels, implemented as a non-spiking recurrent system.

2.1 State-based implementation of spiking neurons

Since our goal is to explore architectures that mimic those of biological systems, it is important for us to keep the fundamental aspects of biological systems. However, one of the challenges is how to transform an asynchronous network of analog devices into a physically equivalent model that functions in a synchronous way, respects the analog aspects of the internal state of the neuron, and yet it may be compatible with existing VLSI capabilities. Our approach was to explore a synchronous, recurrent, yet accurate representation of a leaky integrate and fire model.

One of the standard models of spiking neurons is the leaky integrate and fire model, where the neuron dynamics is captured by the following differential equation:[6]

$$\tau \frac{dv_i}{dt} = -v_i + \sum_j w_{ji} \delta(t - t_j) + v_{\text{ext}}^{(i)} \quad (1)$$

Here v_i is the neuron potential, w_{ji} are the different synaptic weights, t_j the firing times of presynaptic neurons, τ controls the leakage of the potential, and $v_{\text{ext}}^{(i)}$ represents other external inputs to the neuron. When the membrane potential reaches the threshold voltage v_0 , the neuron spikes, the voltage resets, and the neuron is non-responsive to inputs during an absolute refractory period τ_r .

$$v_i(t) = 1 \Rightarrow v_i(t + t') = 0 \quad \forall \quad t' < \tau_r \quad (2)$$

Our approach takes advantage of the fact that during any time interval $\Delta t \leq \tau_r$ a neuron can spike only once. We can therefore discretize Eq. 1 by integrating over Δt . This allows us to express the output of each neuron during each interval as a binary variable $s_i = 0, 1$

In order to transform the firing times t_j of the neurons projecting into our neuron into time steps n_i , so that $n_j \Delta t \geq t_j < (n_j + 1) \Delta t$, we assume that t_j is homogeneously distributed within that interval: $P(t_j) = \frac{1}{\Delta t}$. Through this approximation we lose any information on the firing times at time scales smaller than Δt . However, as we'll show in the next section the penalty for this is small. This allows us to formulate Eq. 1 as:

$$\begin{aligned} v_i(n) &= (1 - s_i(n-1)) \left[\xi_i(n) \left(1 - e^{-1/\tau_m} \right) + v_i(n-1) e^{-1/\tau_m} \right] \\ &\quad + s_i(n-1) \xi_i(n) \left(1 - \tau_m (1 - e^{-1/\tau_m}) \right) \\ s_i(n) &= H(v_i(n) - v_{0i}) \\ \xi_i(n) &= v_{\text{ext}}^{(i)} + \sum_j \frac{w_{ji}}{\tau_r} s_j(n - n_{ji}) \end{aligned} \quad (3)$$

where $H(\cdot)$ is the Heaviside or step function.

The state of a neuron is therefore characterized by a set of two parameters, s_i and v_i : $s_i = 0, 1$ depending on whether the neuron is spiking at that time step, and v_i is the (analog) value of membrane potential. In addition there are several parameters controlling the behavior of the neuron: τ_m is the neuron time constant normalized to the absolute refractory period, and v_{0i} is the firing threshold voltage.

The two terms, one proportional to $(1 - s_i)$ and another to s_i , in Eq. 3 are due to the presence of the absolute refractory period on the evolution of the membrane potential v_i : if the neuron spiked in the prior interval, it will be receptive to input only during a fraction of the next time interval.

If we neglect the impact of the refractory period that we just mentioned from Eq. 3, the recurrence relation for v_i simplifies to:

$$\begin{aligned} v_i(n) &= \left(1 - e^{-1/\tau_m}\right) \xi_i(n) \\ &\quad + ((1 - s_i(n-1)) v_i(n-1) e^{-1/\tau_m}) \\ s_i(n) &= H(v_i(n) - v_{0i}) \\ \xi_i(n) &= v_{\text{ext}}^{(i)} + \sum_j \frac{w_{ji}}{\tau_r} s_j(n - n_{ji}) \end{aligned} \quad (4)$$

Eq. 4 represents a recurrent version of the classic McCulloch-Pitts model. [11] If we can neglect the memory effect on the neuron the classic model is recovered.

2.2 Modulatory interactions

2.2.1 Modulatory processing. The focus of modulated processing is the ability to tune the way input data is processed depending on a state, as shown in the trivial example from Figure 1. A central assumption in our model is that the temporal scale of modulatory interactions is slower than that of spiking neurons. We can then model these slower, modulatory interactions as recurrent systems, so that

$$x(n+1) = \frac{a_i(n)\tau_m + x(n)}{1 + \tau_m} \quad (5)$$

where a_i can be both external or through direct integration of spikes.

This modulatory component can interact with the spiking component in our model in four possible ways: 1) through changes on the threshold voltage v_0 2) through and additional input to the neuron 3) through changes in the characteristic response time of the neuron τ , and 4) through the modulation of all postsynaptic weights.

A key advantage of modulatory processing is that it scales linearly with the number of neurons and not with the number of synapses. They also allow us to build systems that may be flexible yet have fixed or binary connectivity, which can greatly simplify their implementation.

2.2.2 Modulated learning. Spike timing dependent plasticity has been by far the most explored learning approach for spiking neural networks. However, this is not the unique mechanism available. In the case of insects, it has been demonstrated that modulated learning is one of the underlying mechanisms for the acquisition of new knowledge: in the insect brain, the activity of clusters of modulatory neurons correlates with the strengthening of synaptic weights.[1, 2] If in spike timing dependent plasticity the synaptic strength W_{AB} between two neurons A and B is a function of their activity: $W_{AB} = f(A, B)$, in the case of modulated learning, the synaptic strength is a function of the pre-synaptic neuron and a *third* neuron C: $W_{AB} = f(A, C)$, and optionally the post-synaptic neuron as well in the case of modulated STDP. This provides a substrate to integrate logic and memory through dynamic plasticity: we can actuate learning or strengthen certain pathways by the action of a third input.

Figure 3: Correlation between the standard LIF model and the state model of a spiking neuron used in this work

Figure 4: Different encodings implemented in this work

3 RESULTS

3.1 The state based network is functionally equivalent to the asynchronous LIF

In order to understand the impact of our synchronous approximation to a leaky integrate and fire model, we have compared the activity pattern of our model and asynchronous model for a series of randomly connected networks subject to a random set of inputs. In Figure 3 we show the correlation between the activity of a leaky integrate and fire modeled following equation 1 and its corresponding state representation given by Eq. 3.

The correlation between the two models is excellent, indicating that the loss of information of the exact firing times does not negatively impact its functionality for a broad set of conditions. In particular, this allows us to extract the lessons from prior exhaustive studies on the leaky integrate and fire neurons to a recurrent, synchronous model.[6]

3.2 The learning ability of spiking networks is equivalent to non-spiking analog

In order to explore the ability of different encodings to learn and discriminate between different inputs we have used the state model introduced above to benchmark the performance of a shallow network against the MNIST dataset. We have considered the following four types of encoding (Figure 4):

- (1) *Single spike delay encoding.* In this case the input is codified as a single spike whose delay with respect to a common epoch decreases with the input intensity.

Table 1: MNIST test accuracy: results for a shallow spiking architecture for different types of encodings and outputs. The state of the art for similar shallow systems is 93%[10]

Encoding	Sigmoidal	Binary non-excl.	Binary excl.
Single spike delay	88%	83%	65%
Spike train delay	90%	90%	87%
Spike rate, prob	91%	91%	87%
Constant	88%	83%	65%

- (2) *Spike train delay encoding.* The single spikes from the time encoding are subsequently followed by a periodic train of spikes spaced by an interval equal to the time delay.
- (3) *spike rate, probabilistic.* The spiking neurons are fed a constant input proportional to the intensity of each pixel in the input data.
- (4) *Constant analog input.* The input is codified as constant input signal.

For each of the proposed encodings, we applied a stochastic gradient descent method to optimize our networks using a cost function that optimized the total number of spikes:

$$C = \frac{1}{N} \sum_i \left(\sum_m a_i(m) - n_i \right)^2 \quad (6)$$

regardless of when those spikes took place.

The results show testing accuracies as high as 90%. This compares well with state of the art results for shallow (no hidden layers) neural networks.[10] In particular, if we optimize a shallow network of non-spiking sigmoidal neurons using the same cost function and stochastic gradient descent method, the resulting accuracy is 91%, as shown in Table 1. If we switch back to a binary output, the differences between the different encodings become more pronounced. The non-exclusive accuracy column includes all cases where the correct digit achieves the maximum number of spikes, regardless of whether other digits also achieves the same number of spikes. In contrast, the exclusive accuracy column considers only those cases where no other digits achieve the maximum number of spikes. The need to distinguish between these two cases comes from the fact that the output is measured in terms of the number of spikes, so two neurons can have identical outputs. According to the results obtained, the single spike encoding's testing (exclusive) accuracy drops below 70%. In contrast, both the spike train time delay and the probabilistic spike rate encodings maintain a high testing accuracy with binary outputs.

3.3 Modulated processing and learning: some simple examples

At a neuron level, the main effect of postsynaptic modulation is to change the transfer function of the neuron from a linear device to a rectified type of transfer function as we reduce postsynaptic modulation. This is shown in Figure 5.

More interesting, we can use modulation to control perception based on expectations. In Figure 6 we show the averaged spike rate of a convolutional layer of a spiking network tracking an array of pixels for 8 consecutive sequences in which an object is moving

Figure 5: Impact of post-synaptic modulation on the transfer function of a neuron

Figure 6: Averaged activity of spiking neurons pooling input from a series of aligned receptive fields.: a reduction on input quality causes a reduction of the activity

from left to right. If we degrade the quality of the input by removing 50% of the spikes after an initial step, the response of the system is, as expected, affected. In contrast, if the activity of this layer is modulated by the expectation that an object is present and moving rightwards, under the same degraded conditions the network is able to successfully track the object, as shown in Fig. 7.

For modulatory learning, we have first explored the case of an agent moving over a 2D space receiving a series of spatially correlated input data. For this particular example we consider a series of point sources randomly distributed over a 2D map (Figure 8). These can be interpreted as either specific set of pulses or frequencies, in the case of radio or visual navigation, or it can be series of molecular signals, for instance different odors. We consider that the intensity of these pulses decay with distance following a gaussian profile.

Our network is a three layer network that mimics the architecture described in Figure 1, with a fan out - fan in architecture that maps input signals to a larger dimensional space through a highly sparse connectivity, which then fans-in to a set of drivers that the system uses to decide when and how to respond. The agent carries out a random walk over the map with a dwell time that is modulated

Figure 7: Averaged activity of spiking neurons pooling input from a series of aligned receptive fields: predictive modulation at a neuron level enhances the system’s ability to track an object even when the input quality has been severely degraded

by the valence associated to the input. A positive valence leads to longer dwell times during the random walk.

As shown in Figure 8, in absence of modulatory input the dwell time is homogeneously distributed on the map. In order to train the system to spend more time at a preferred location, we introduced a modulatory signal that strengthen the positive response with a certain probability p_0 whenever the agent is within a certain region. A feedback path ensures that, once the system has learned the desired response, the modulatory neuron becomes inactive and the synaptic weights are no longer updated. The system remembers the valence associated to the input, and this alters its behavior. This is shown in Figure 8, where through the use of modulatory signals we can make the system spend more time on desired locations in the map.

4 DISCUSSION AND CONCLUSIONS

In the previous section we have introduced two simple examples of systems incorporating modulatory interactions. In the case of neuron-level modulation, it provides a way of controlling functionality without having to introduce drastic changes on the system’s connectivity. In principle, the use of neuron-level modulation provides a pathway toward adaptability and programmability that scales linearly with the number of neurons rather than the number of connections. One of the tantalizing possibilities of such type of neuron-level interactions is that it enhances the flexibility of neural network with binary synaptic weights, which would allow to store the network’s connectivity as a single bit, reducing the complexity of cross-bar architectures or the size of lookup tables. One of our current areas of research is the design of optimization algorithms for such hybrid architectures, with digital/binary connectivity and continuously varying parameters modulating system performance at a neuron level.

Modulated learning provides a simple yet powerful physical implementation of reinforcement learning. A key advantage of this approach is that it can be readily implemented using existing

Figure 8: Aggregated dwell times of an agent carrying out a random walk over a 2D map. Top left: spatial distribution of source points providing the input of the system. Top right: fraction of action Kenyon cells as a function of the position. Bottom left: Dwell time distribution in absence of reinforcement. Bottom right: Dwell time distribution after on-line reinforcement of the central position of the map, showing an increase of dwell time close of the excitation area in the top left quadrant.

floating-gate type of technology, as it was already advanced by Mead and collaborators.[7] The use of neurons with binary outputs also simplifies the design of such elements, since they would always operate at a similar DC point, and it is based on local information. Reinforcement learning presents significant advantages for implementations based on memristive type of devices. In the case of our random walker mentioned in the previous section, reinforcement is shut off once a desired level of response is achieved. This means that variabilities at a device performance can be eliminated by the incorporation of such internal feedback loops. In this context we have also explored how to implement real online learning using memristors in a way that it does not interfere with the constant processing of input data: we have found that the combination of voltage-controlled devices with a clear threshold and the use of selector devices preventing current inversion is critical to minimize the cross-talk between sensing and modulated learning.

Our results show that spiking neurons can be as efficient as non-spiking approaches in terms of learning. In particular, in as few as 8 cycles it is possible to achieve accuracies that are as good as the equivalent non-spiking architectures, at least for the case of shallow

systems. Consequently, while there is always going to be a trade-off between binary spiking and analog non-spiking systems from a design standpoint, the use of binary spikes do not necessarily lead to a loss in performance. An additional aspect of spiking systems is that they are ideally suited for the processing of pulsed and sparse input streams. Our results also show that performance in fairly insensitive to the structure of the spike trains. This opens the possibility of considering both periodic spike generators, for instance through conventional numerically controlled oscillators, or stochastic generators that take advantage of novel devices, such as spin-torque stochastic oscillators, using concepts such as the spin dice approach to encode inputs as stochastic streams of random bits.[8]

Finally, a key design aspect that makes edge processing different from conventional computing is that the relevant temporal range, and therefore the characteristic time of our *neuromorphic clock* Δt is driven by the input data. In our model, all the relevant parameters of the neuron are normalized to Δt . Consequently, the ability to modulate these parameters at a neuron level enable us to design systems with an adaptive dynamic range.

ACKNOWLEDGMENTS

This material is based upon work supported by Laboratory Directed Research and Development (LDRD) funding from Argonne National Laboratory, provided by the Director, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-06CH11357

REFERENCES

- [1] Yoshinori Aso, Daisuke Hattori, Yang Yu, Rebecca M. Johnston, Nirmala A. Iyer, Teri-TB Ngo, Heather Dionne, L. F. Abbott, Richard Axel, Hiromu Tanimoto, and Gerald M. Rubin. 2014. The neuronal architecture of the mushroom body provides a logic for associative learning. *eLife* 3 (Dec. 2014), e04577. <https://doi.org/10.7554/eLife.04577>
- [2] Yoshinori Aso, Divya Sitaraman, Toshiharu Ichinose, Karla R. Kaun, Katrin Vogt, Ghislain Belliard-Guérin, Pierre-Yves Plaçais, Alice A. Robie, Nobuhiro Yamagata, Christopher Schnaitmann, William J. Rowell, Rebecca M. Johnston, Teri-T. B. Ngo, Nan Chen, Wyatt Korff, Michael N. Nitabach, Ulrike Heberlein, Thomas Preat, Kristin M. Branson, Hiromu Tanimoto, and Gerald M. Rubin. 2014. Mushroom body output neurons encode valence and guide memory-based action selection in *Drosophila*. *eLife* 3 (Dec. 2014), e04580. <https://doi.org/10.7554/eLife.04580>
- [3] Cornelia I. Bargmann. 2012. Beyond the connectome: How neuromodulators shape neural circuits. *Bioessays* 34, 6 (June 2012), 458–465. <https://doi.org/10.1002/bies.201100185>
- [4] Matthew E. Berck, Avinash Khandelwal, Lindsey Claus, Luis Hernandez-Nunez, Guangwei Si, Christopher J. Tabone, Feng Li, James W. Truman, Rick D. Fetter, Matthieu Louis, Aravinthan DT Samuel, and Albert Cardona. 2016. The wiring diagram of a glomerular olfactory system. *eLife* 5 (May 2016), e14859. <https://doi.org/10.7554/eLife.14859>
- [5] Gordon J. Berman, William Bialek, and Joshua W. Shaevitz. 2016. Predictability and hierarchy in *Drosophila* behavior. *PNAS* 113, 42 (Oct. 2016), 11943–11948. <https://doi.org/10.1073/pnas.1607601113>
- [6] Nicolas Brunel. 2000. Dynamics of Sparsely Connected Networks of Excitatory and Inhibitory Spiking Neurons. *Journal of Computational Neuroscience* 8, 3 (01 May 2000), 183–208. <https://doi.org/10.1023/A:1008925309027>
- [7] C. Diorio, P. Hasler, A. Minch, and C. A. Mead. 1996. A single-transistor silicon synapse. *IEEE Transactions on Electron Devices* 43, 11 (Nov 1996), 1972–1980. <https://doi.org/10.1109/16.543035>
- [8] Akio Fukushima, Takayuki Seki, Kay Yakushiji, Hitoshi Kubota, Hiroshi Imamura, Shinji Yuasa, and Koji Ando. 2014. Spin dice: A scalable truly random number generator based on spintronics. *Applied Physics Express* 7, 8 (2014), 083001. <http://stacks.iop.org/1882-0786/7/i=8/a=083001>
- [9] Kang I Ko, Cory M Root, Scott A Lindsay, Orel A Zaninovich, Andrew K Shepherd, Steven A Wasserman, Susy M Kim, and Jing W Wang. 2015. Starvation promotes concerted modulation of appetitive olfactory behavior via parallel neuromodulatory circuits. *eLife* 4 (jul 2015), e08298. <https://doi.org/10.7554/eLife.08298>
- [10] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (Nov 1998), 2278–2324. <https://doi.org/10.1109/5.726791>
- [11] Warren S. McCulloch and Walter Pitts. 1943. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics* 5, 4 (01 Dec 1943), 115–133. <https://doi.org/10.1007/BF02478259>
- [12] Dick R. Nässel and Åsa M. E. Winther. 2010. *Drosophila* neuropeptides in regulation of physiology and behavior. *Progress in Neurobiology* 92, 1 (Sept. 2010), 42–104. <https://doi.org/10.1016/j.pneurobio.2010.04.010>