Open access • Proceedings Article • DOI:10.1109/IJCNN.2012.6252779

Emergence of competitive control in a memristor-based neuromorphic circuit

Saeed Afshar, Omid Kavehei, André van Schaik, Jonathan Tapson ...+2 more authors

Institutions: University of Western Sydney, University of Melbourne, University of New South Wales

Published on: 10 Jun 2012 - International Joint Conference on Neural Network

Topics: Neuromorphic engineering, Memistor and Memristor

WCCI 2012 IEEE World Congress on Computational Intelligence
June, 10-15, 2012 - Brisbane, Australia IJCNN

Emergence of Competitive Control in a Memristor-Based Neuromorphic Circuit

Saeed Afshar*‡, Omid Kavehei†, André van Schaik*, Jonathan Tapson*, Stan Skafidas†, Tara Julia Hamilton‡
*Bioelectronics and Neuroscience, University of Western Sydney, Australia
†Centre for Neural Engineering, University of Melbourne, Australia
‡School of Electrical Engineering and Telecommunications, University of New South Wales, Australia
Email: s.afshar@student.unsw.edu.au

***Abstract*—Recent work in neuroscience is revealing how the blowfly rapidly detects orientation using neural circuits distributed directly behind its photo receptors. These circuits like all biological systems rely on timing, competition, feedback, and energy optimization. The recent realization of the passive *memristor* device, the so-called fourth fundamental passive element of circuit theory, assists with making low power biologically inspired parallel analog computation achievable. Building on these developments, we present a memristor-based neuromorphic competitive control (mNCC) circuit, which utilizes a single sensor and can control the output of N actuators delivering optimal scalable performance, and immunity from device variation and environmental noise.**

***Index Terms*—Neuromorphic, Memristor, Distributive Control, Divisive Gain, Cybernetics, Neural Networks, Emergence**

I. Introduction

Flies are nature's supreme aerobatic pilots. The control system that sits behind the eyes of a male House Fly is capable of chasing an equally maneuverable female at angular velocities exceeding 3000 deg/s in less than two hundredths of a second and in any direction [1]. Amazingly, the few hundred neurons responsible for this extraordinary level of control, constitute less computational hardware than is present in most toasters [2]. Furthermore, in contrast to signals in a toaster, which move at relativistic speeds, the speed of the signals in the fly's neuronal controller propagate at well below 100 m/s [3], [4]. Not surprisingly, the optic flow processing controlling such performance has been the focus of much recent research [5].

Of the many profound developments to date, has been the discovery that the neural computation involved in fly motion vision and the compensatory motor response, begins both physically and temporally immediately after input enters the eyes [6]. Moreover, this computation is fully parallel and with no central processing element acting as a bottleneck, information moves at close to the conduction velocity of the signal from the sensors (eyes) to the actuators (wings) and is processed as it travels (Fig. 1) [7]. In this paper, we present a control circuit that draws its inspiration from the elegance and efficiency of the fly eye. We call this biologically-inspired control system: Neuromorphic Competitive Control (NCC). Like the fly eye, it utilizes fundamental elements such as timing, competition, and feedback in order to optimize energy and ensure robustness to noise and other nonidealities.

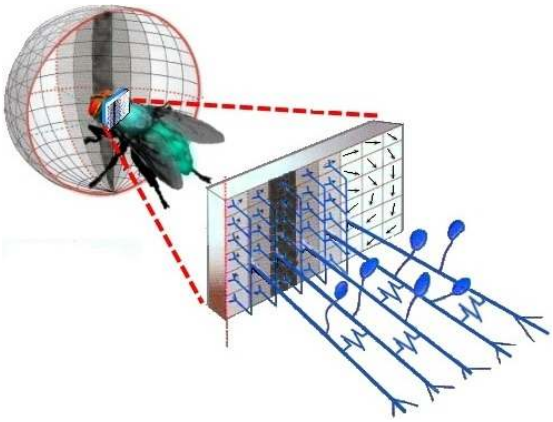

Fig. 1. Optic flow processing circuits in the blowfly *Calliophora vicina*, modified from [7].

Along with the challenge of modeling the emergent properties of such a control system, we also examine how such a circuit can be physically implemented. We show that the recently *rediscovered* fourth fundamental circuit element, the memristor, can simplify the circuit design and illustrate how it can be used in several modes in order to facilitate the operation of NCC or memristor-based NCC (mNCC).

II. Control in Artificial Systems and in Nature

Control systems come in many forms. A simplified diagram of a control system in a conventional configuration is given in Fig. 2. Here the *plant* is required to give a particular output. A *sensor* is then employed to measure the output of the system. This measured output is then compared with a reference signal and the error between the two signals is used to drive a *controller*. The output of the controller is a signal that changes the state of the system in such a way that its output should move closer to the desired output (i.e. the input or reference to the control system).

Conventional control systems, as described above, are used to control today's most advanced technology. Since such systems conventionally employ digital microcontrollers, the

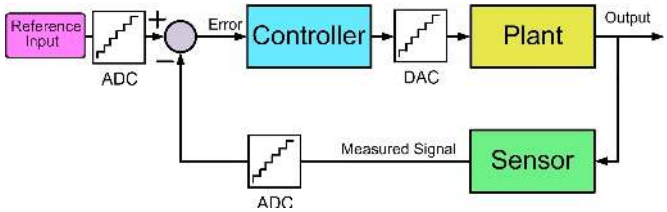

Fig. 2. Conventional digital control system.

signal(s) measured by the sensor, the reference signal, and control signal used to drive the actuator(s) need to be discretized both in amplitude and time. The accuracy of the system is dependent on the number of digital bits utilized and this can significantly affect the power consumption and complexity of the entire system.

The control systems utilized in biology (such as the fly eye) and the modern digital control system described above are two drastically different and highly optimized solutions to very similar control problems. The differences between these control solutions reflect the fundamentally different processes that shaped their design and the constraints that led to their creation.

The evolution of the fly was in response to an unpredictable environment with energy, and its limited supply the major constraint. The development of the fly eye control system via evolution relies on emergent behavior rather than rational design. In emergent systems, simple components interact together through multiple pathways resulting in complexity, both in design and in performance [8]. In stark contrast, artificial control systems, in particular those that must perform complex tasks, are often broken into subsystems where the complexity is found *within* each subsystem but where the interactions between subsystems are kept as simple as possible. Thus, while a conventional control system is easy to understand, an emergent system can be easily scaled with no data conversion bottlenecks or processing power limitations due to its use of simple building blocks. Moreover, emergent systems are inherently power efficient, while artificial control systems incur a trade-off between power consumption and complexity.

Control systems in nature are analog, however, attempting to replicate these systems using an analog computer comes with a significant set of complications [9]. Specifically, analog computers are power hungry. They also often suffer from instability and have only a narrow operating range, which must often be manually ascertained [10]. The precision of the analog computer is also relatively low due to its poor signal-to-noise ratio and reliance on matched circuit elements. These factors together with the design complexity of analog and mixed signal systems are the main reason for the dominance of digital computing [11]. Thus, in order to develop a control system that is inspired by nature, the limitations of the analog computer must be addressed. This challenging task has been taken on by the field of neuromorphic engineering [12], [13].

In the next section, we discuss the memristor, a circuit element that facilitates low-power analog computation. Following this we present our control algorithm, memristor-based Neuromorphic Competitive Control (mNCC), which is based on emergent control systems that use stochastic rather than deterministic processing and thus, subverts the narrow operating range of the analog computer.

III. The Memristor

As the transistor's feature size approaches 16 nm and beyond, a number of fundamental issues emerge that create a set of practical difficulties in fabrication, voltage scaling, and energy efficiency. As a result, technology scaling is not having the same impact on computing performance as it did in the past. In this situation, the memristor is one of the promising disruptive technologies to enable further energy efficiency per operation through ultra-high density integration and parallelism.

Unlike three-terminal transistors, it is a two-terminal device and does not require power to retain its states. Memristor's resistance state, *memristance*, can be controlled by either charge or fluxlinkage [14], [15].

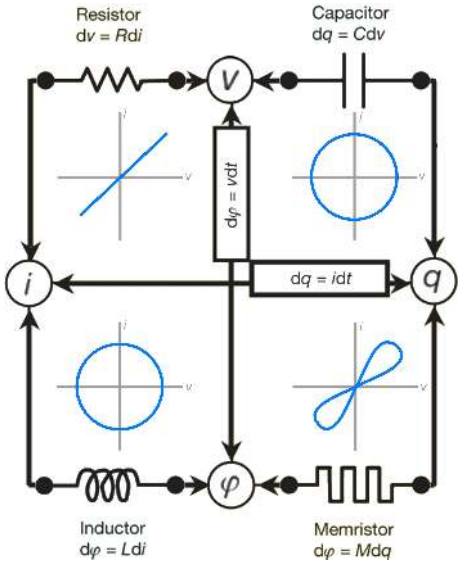

Fig. 3. The four fundamental two-terminal circuit elements and their current voltage response to a sinusoidal input, modified from [16].

As shown in Fig. 3, the pinch hysteresis loop in the current-voltage characteristic of the memristor make it distinguishable from the rest of the fundamental passive elements [15]. Change in the resistance state can be used to store digital or analog information in a crossbar array of memristive devices. Memristive device concept covers a wide range of devices with inherent memory. In fact, any hysteretic resistive device that is able to retain its state without the need of a power supply, is a memristive device.

Memristors implemented in compatibility with CMOS have a Metal-Insulator-Metal (MIM) device structure [14]. The switching effect occurs by the formation of a conductive

filament inside the insulator layer due to electrostatic, electrochemical metalization, valency change, phase change, or thermochemical effects [17]. A current controlled memristor device can be described with the following relationship [14],

$$\begin{cases} V(t) &= R(x,I,t)I(t) \\ \dot{x} &= f(x,I,t)\ , \end{cases} \tag{1}$$

where $R(\cdot)$ is the device resistance (memristance), $f(\cdot)$ is a continuous function describing the state variables, x. The simplest model of an ideal memristor can be found in [14], [16]. The analog memristive effect is the result of gradual, and incremental changes in resistance as the result of continuous motion of metal ions that are incorporated into the insulator material during the fabrication process [18], [19]. This property is used as a key part of our mNCC system.

In addition to the memory application of memristors, these devices have shown a number of promising capabilities, such as performing logic operations [18], [20], the appearance of a memory element under stochastic conditions [21], and mimicking synapse functionality through Spike Timing-Dependent Plasticity (STDP) [18], given that this STDP functionality is crucial for use in neuromorphic computation [22], these findings have garnered significant attention.

The memristor also promises a revolutionary change in the computing architecture that plays an important role in the future improvement of our proposed control system. As mentioned earlier, a radical change in computer architecture is required to be able to physically mimic natural system's behavior. Today's artificial control systems, however, are based on the centralized von Neumann computing machine that suffers from a large sequential (fetch-execute-store cycles) processing overload due to the existence of the data bus between memory and logic. In contrast, neuromorphic computing hardware introduced a more efficient implementation but not necessarily low-power implementation for complex systems. Therefore, tighter coupling between memory and logic seems to be necessary to significantly increase energy efficiency and reduce power consumption. Experimental results [18] show that the memristor is able to provide a solution at the very bottom level of design hierarchy (device level). In fact, each memristor combines (non-volatile) memory characteristics and in-situ computing capability, which promises an entirely new computing architecture.

Considering the nanometer scale feature size of a memristive device and the tremendously large neuron-synapse connectivity in addition to the extremely high synaptic density in biological systems, memristive devices may provide a significant change in the traditional approach of CMOS-based implementation of neuromorphic engineering. Furthermore, owing to the highly nonlinear (exponential) relationship between the voltage bias and programming pulse width, highly energy efficient memristor-based systems can be achieved [14].

In this paper, we use the memristor in two different modes of operation. These two modes are described graphically in the current-voltage characteristic plots in Fig. 4 and the transient response plots of Fig. 5.

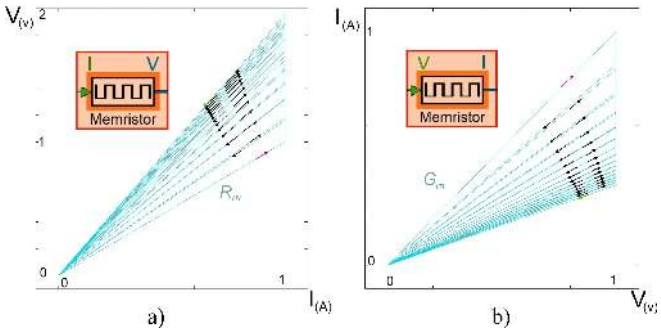

Fig. 4. Current-Voltage characteristics of the memristor in response to a train of spikes. The memristor response, (a) in the IV (current in, voltage out) synaptic operating mode, and (b) in the VI (voltage in, current out) choke operating mode.

In Fig. 4, the plot on the left shows the synaptic operating mode for the memristor. Here, the input to the memristor is a current-controlled train of pulses which progressively increases the resistance (slope), the measured output is the voltage across the memristor. Thus, since the input current is kept constant, the output voltage progressively increases. This is the same mode of operation as that used for the emulation of synaptic weight adaptation.

The plot on the right, in Fig. 4, shows the *choke* operating mode for the memristor. Here, the input to the memristor is a voltage-controlled train of pulses which progressively decreases the conductance (slope), the measured output is the current drawn by the memristor. Thus since the input voltage across the memristor is kept constant, the output current rapidly decreases. This mode of operation proves to be of critical significance in the design of mNCC.

IV. Memristor-based Neuromorphic Competitive Control

Memristor-based Neuromorphic Competitive Control (mNCC) is a control scheme that emerges when simple building blocks, namely neurons, are connected together in such a way that competition is created between the individual units. Here, competition is generated through inhibitory connections between neurons.

Fig. 6 shows the top-level block diagram of the mNCC system embedded, in an example application: a *smart* torch. Here the *N* LEDs that make up the light emitted from the torch are the actuators while a sensor resides in the centre of the LED array. For each actuator there is a mNCC circuit regulating its output while there is only one sensor for the entire system. In mNCC the output of each of the actuators is the result of complex non linear interactions between three recurrent loops that operate simultaneously.

In order to explain the overall behavior of the entire system, we will examine each recurrent path in isolation describing: how it performs as a single neruo-computational unit, how

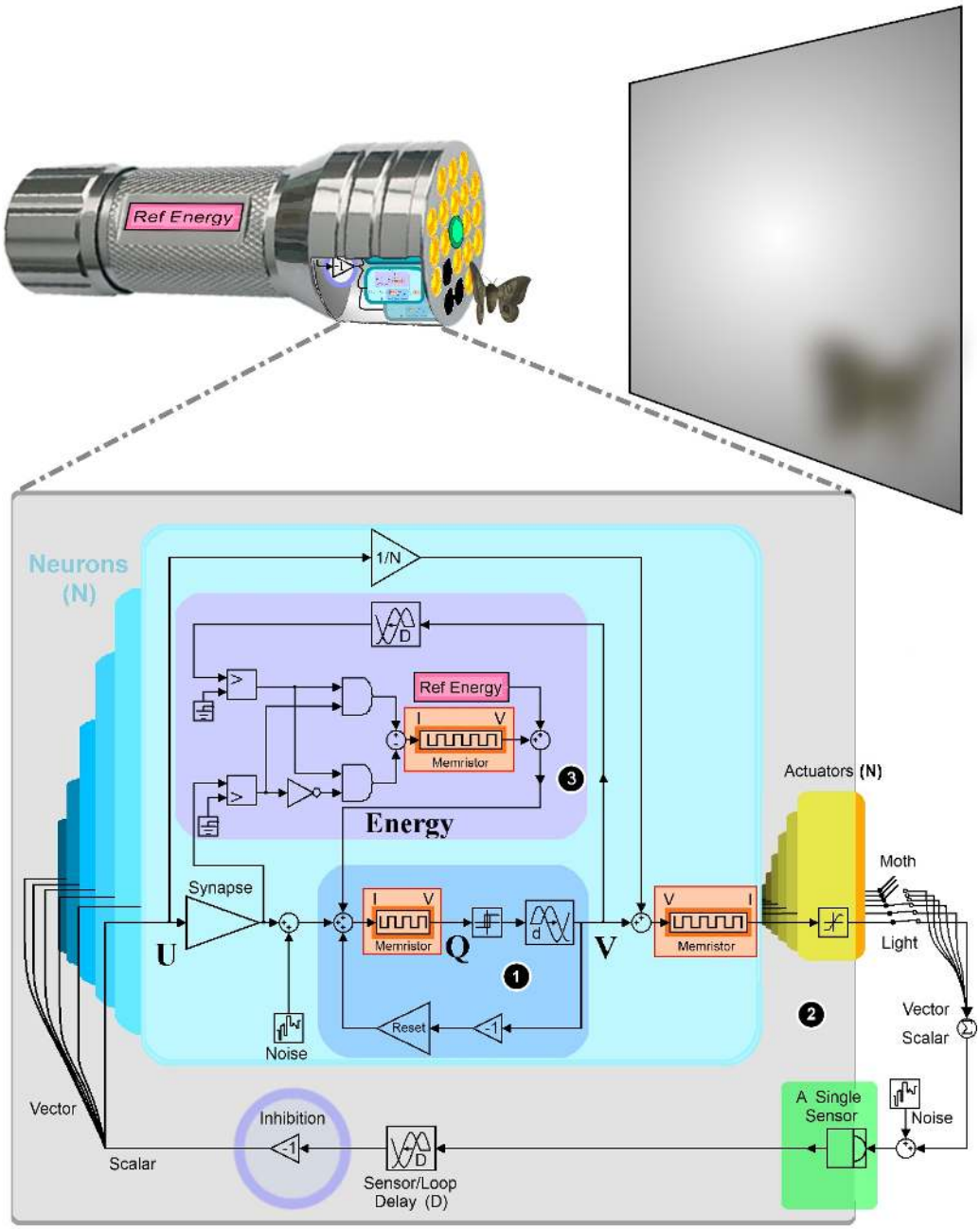

Fig. 6. Top-level functional block diagram of memristor-based Neuromorphic Competitive Control (mNCC) system.

it brings the system closer to its performance goals, and the additional features it requires in order to meet the performance specification. Invariably, these additional features lead to the creation of another loop.

As a preliminary consideration we begin with the reference signal to the system, which we will call *reference energy*. This signal determines the stochastic level of actuation at the output and is analogous to the battery in the smart torch. It is distributed between each actuator's mNCC circuit based on the actuator's *effectiveness* as measured by the sensor. In the following subsections each recurrent loop is described along with its interactions with the energy signal.

A. Loop 1: The modified leaky-integrate-and-fire neuron based on the memristor

Fig. 7 gives a more detailed diagram of the modified leaky-integrate-and-fire neuron (LIF) based on the memristor. Unlike the description of the ideal memristor in Section III, the memristor here is *leaky*. The principle of the leaky memristor is illustrated in Fig. 5. Here, panels b) and c) show the difference in the voltage output of a memristor with, c), and without, b), a constant *leak*. While d) and e) show the difference in the current output of a memristor. The leak is

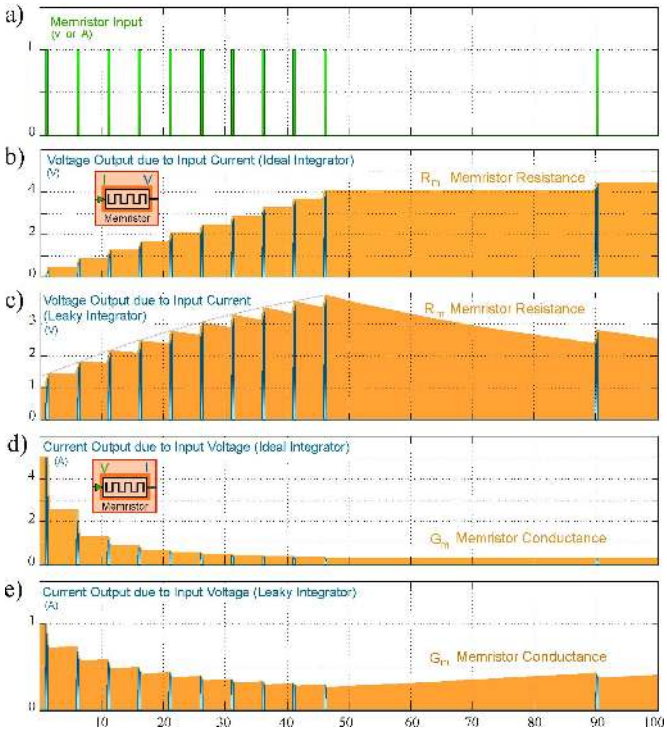

Fig. 5. Time domain simulation of the memristor in its two modes. The top graph shows input signal (voltage or current) and the others illustrate memristor responses.

introduced in fabrication and, just as in biology, ensures that memory is not perpetual but rather adapts over time giving the strongest response to the most recent or common events. In this case, the leak also ensures that the operating range of the circuit does not become too large.

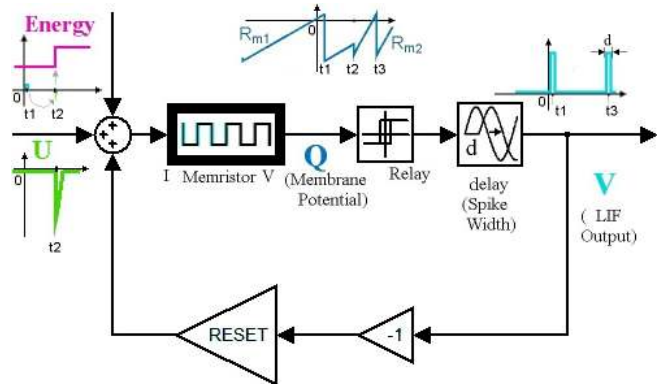

Fig. 7. Modified Leaky-Integrate-and-Fire (LIF) neuron based on the memristor.

In Fig. 7, **U** is an inhibitory input, **Q** is the membrane potential and **V** is the output of the LIF. A description of the LIF's operating principles follows. In the forward path of the LIF, the energy signal, which is in the form of a constant current, is integrated by the memristor until the memristor output reaches a threshold (here the threshold, **Q**, is zero) at which point the relay outputs a signal (here 1 Volt). The signal is then immediately inverted and fed back to the memristor's input via a feedback loop. This in turn switches the relay, resetting the LIF. The net result is a sharp 1 Volt spike periodically produced at the output.

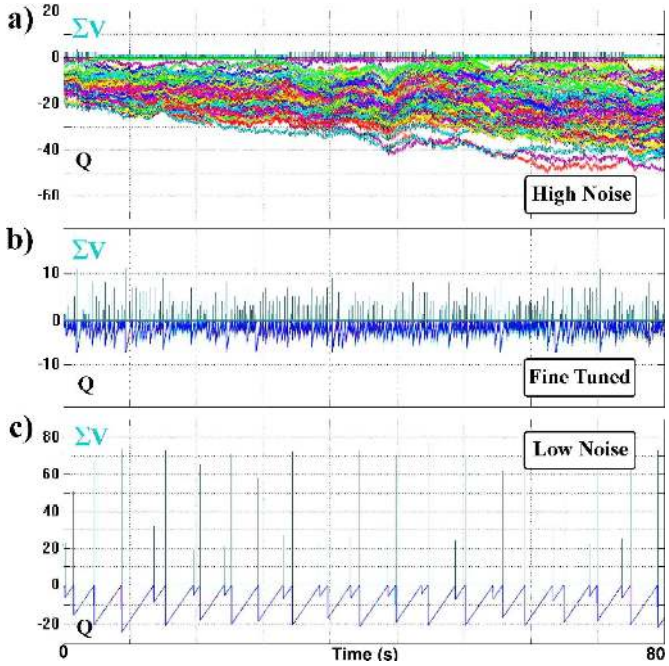

Fig. 8. The effect of noise (or equivalently component mismatch) on the activation pattern of $N = 100$ highly inhibitively coupled Leaky Integrate and Fire neurons. In the panel (a), the high level of noise causes a minority of neurons to completely dominate their neighbors. In (b), the noise is very finely tuned to *minimize* the domination effect, while also avoiding the phase locking effect. The panel (c) shows the characteristic phase locking of seen in low noise networks.

When large networks of LIFs are globally (all-to-all) connected such that the positive output of each is inverted and fed back to the input of the rest (mutual inhibition), a surprising number of organised behaviors can occur. These behaviors include: random spreading, decoherence, and coordinated spiking.

From a neuro-computational point-of-view, the *decoherence phenomenon* is of particular interest. This describes regular out-of-phase spiking as illustrated in Fig. 8 c). The ability of LIF networks to decohere facilitates decentralized yet highly effective sampling of a neural network's external environment. In this paper we present an actuator/sensor system, however, this phenomenon may underpin more complex and interesting problems in neuro-computation.

There are, however, a number of major problems with a global inhibition network of LIF neurons. Specifically, for our application where we assume real-world conditions, i.e. mismatched variables and noise, there is only a narrow operating range where the organised behaviors listed above can occur, as shown in Fig. 8. Outside this range the network of neurons will phase lock or synchronize in a low noise environment

or permit some LIFs to dominate the output in a high noise environment. For a detailed analysis of inhibitive LIF networks see [23]. Either situation is undesirable and stops the free flow of information through the network.

Within the narrow central operating range (Fig. 8 b)), the decoherence of inhibitory LIFs is delivered by a simple chance scattering of spikes. Thus due to noise, this neuro-computationally desirable behavior is granted for free [24].

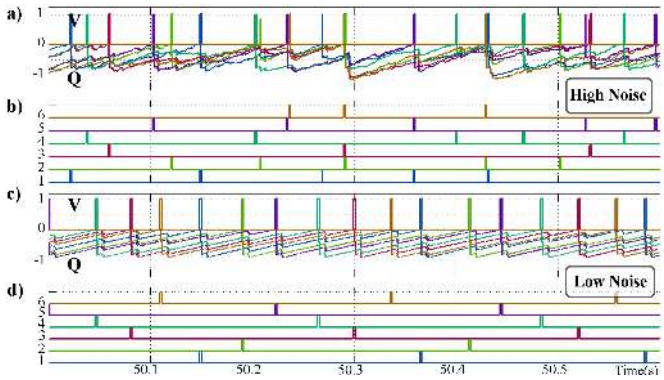

Fig. 9. The mNCC enforced decoherence across 3 orders of magnitude difference in noise. A simple LIF network with similarly high inhibitive coupling and high/low noise would immediately enter a domination/phase lock state.

The major drawback of globally inhibited networks of LIFs is their lack of robustness. Any introduced positive feedback (for example to allow the more effective neurons to spike more often) can result in the system becoming permanently stuck in a sub-optimal state where neurons that had been reinforced previously refuse to relinquish their dominance. Such a situation can lead to degraded performance and even instability.

B. Loop 2: The LIF output and the feedback from the sensor

To address the shortcomings of the simple LIF network, a second control loop is introduced. This loop ensures that the activation of dominating neurons is self limiting and that all neurons are given an equal opportunity to fire.

Loop 2 is labelled in Fig. 6. Starting at the LIF output, a memristor in the VI choke mode is placed immediately behind the system's actuators. As a LIF fires more frequently, the memristor acts as a choke constricting the output current. Concurrently, a weak unchoking path is introduced, where the feedback from the sensor is inverted, divided by the number of actuators in the network, N, and channelled into the choked memristor. Since the input coming in from the sensor corresponds to the summed output spikes of all the neurons, the strength of an effective neuron is sustained by the firing of all other effective neurons. A yet more subtle consequence of this actively enforced order, is that formerly ineffective neurons are always offered an opening to probe their environment with test spikes, allowing them to reactivate quickly in response to a changed environment.

This interplay of competition and cooperation delivers an efficient and stable foundation on which more complex functionality can be designed.

Fig. 9 illustrates the consequence of Loop 2 on the spiking behavior of 6 competing neurons in both low noise and high noise situations. In Fig. 9 a) and c) we see the LIF output, $\mathbf{V}$, and the membrane voltage, $\mathbf{Q}$, for the high and low noise conditions respectively. While in Fig. 9 b) and d) we see the spiking outputs of 6 LIFs. These plots show that each neuron has an opportunity to spike, with no phase-locking or dominance by a particular neuron. If Loop 2 were disabled then the system would revert to the domination or phase-locked states of the simple LIF network as shown in Fig. 8 a) and c).

C. Loop 3: The output spike, the detected echo and the adjustment of energy

The requirements of our control system demand we weaken ineffective neurons and strengthen effective neurons thus compensating for external as well as internal imbalances. The two variables from which effectiveness can be measured are the LIF output, $\mathbf{V}$, and the echo received by the sensor, $\mathbf{U}$. An effective neuron is one whose output spike is followed shortly by an echo at its input, and an ineffective neuron is one whose output spikes are not echoed back. We change the energy level into a neuron based on this measure of effectiveness.

In the energy adjustment block (labelled "3" in Fig. 6) a co-incidence detection circuit detects the neuron's effectiveness, any subsequent change in energy is then stored on a memristor operating in the synaptic mode.

In the following section we provide simulation results that verify that the resultant network performs efficiently and effectively under a number of operating conditions. This robustness is crucial for successful silicon implementation.

V. Results

The mNCC system was simulated across a wide range of parameters using Matlab and Simulink software. The mNCC's performance was verified to be invariant to internal and external noise, network size, neuron non-uniformity, and variation, thus, the results in Fig. 10 from a six neuron network accurately captures the salient behaviors of mNCC networks for $N > 1$.

Fig. 10 (a) shows the raw combined output of the mNCC. The positve half of the graph $[0; 1]$ is the super-imposed output of all neurons N1–N6. This corresponds to the LIFs' output, $\mathbf{V}$. The bottom half of the graph $[0; -1]$ is the super-imposed plot of the neuron's membrane potentia,l $\mathbf{Q}$. The mid-line at zero is the spike threshold. In Fig. 10 (b), the positive half of the graph $[0; N]$, here $N = 6$, the output $\mathbf{V}$ is summed (Cyan), here the black plot $\mathbf{V_{avg}}$ is a scaled moving average of $\mathbf{V}$ showing the overall activation level. The negative half of the graph $[0; -2]$ shows the input $\mathbf{U}$ to the mNCC (green), this corresponds to the inhibitory feedback from the sensor. The black plot $\mathbf{U_{avg}}$ is the scaled moving average of $\mathbf{U}$ showing the overall activation detected by the sensor, note that $\mathbf{V_{avg}} \geq \mathbf{U_{avg}}$.

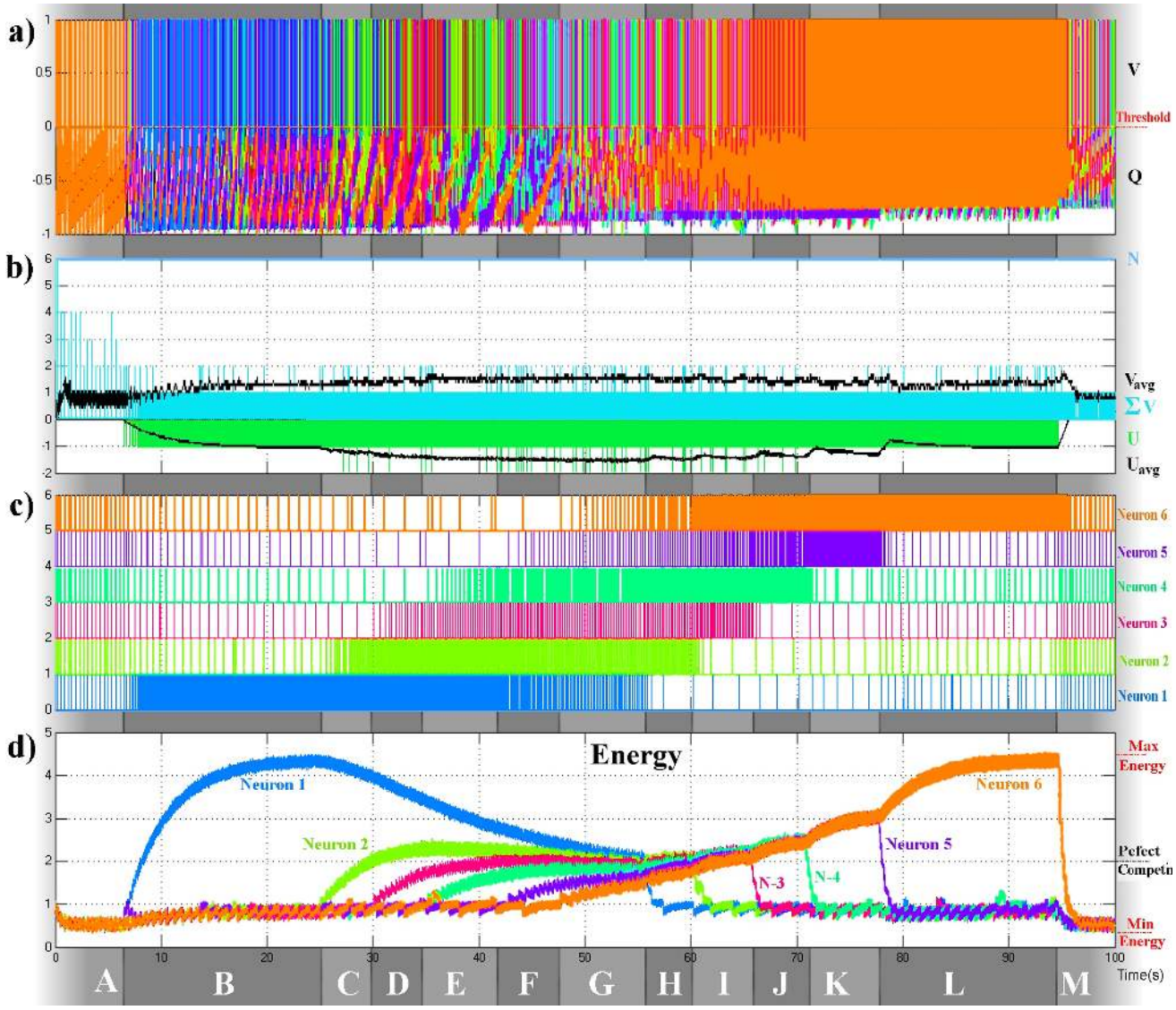

Fig. 10. The dynamic response of a 6 neurons mNCC to the distuptive actions of 6 virtual moths.

Fig. 10 (c) shows the raster plot of each of the neurons N1–N6 and Fig. 10 (d) shows the super-imposed plot of the neruons' dynamic energy level.

The capital letters, A – M below Fig. 10 correspond to external interruptions to the network. For the purposes of illustration we discuss these interruptions in the context of the smart torch example, where we have 6 light sources (actuators) and a single sensor. The operating principle of the smart torch is that a constant illumination level is set. If one of the lights becomes obstructed or ineffective for any reason, the remaining lights must compensate to ensure the desired illumination. An external interruption in this context refers to a light(s) being blocked or unblocked. Each of these external interruptions can be explained as follows:

A — The simulation begins with section **A**. Initially, due to all the lights being blocked (say by six virtual moths), all neurons are fully ineffective and their output spikes are not followed by a delayed echo at the sensor input $\mathbf{U}= 0$.

The energy levels of all neurons are equal and at a minimum. Furthermore, the net output of the LIFs, $\Sigma\mathbf{V}$, is also at a minimum. Note that, due to identical initial states, the system is initially synchronized, but the noise soon causes the neurons to decohere.

B — Light-1 is unblocked, thus neuron 1 is now effective as spikes at its output $\mathbf{V}$ are followed by a delayed echo spike at the sensor input. This coincidence is detected in neuron 1 causing its energy to rise rapidly, increasing its output with a corresponding increase in the inhibitive feedback from the sensor. This in turn causes the other still ineffective neurons to emit test spikes less frequently. This is the mechanism through which an equilibrium based on relative effectiveness is rapidly reached (**B**). This is the optimal energy and actuation solution for the smart torch. Note that for $t \rightarrow \infty$, the actuation detected by the sensor from Neuron-1's actuator$\rightarrow \%75$ of full

actuation.

C–F — During stages **C** through to **F**, Light-2 through to Light-5 are sequentially unblocked. Thus, with each new unblocked actuator a new optimal solution must be reached. Through the same mechanisms described above the system adapts optimally to its new environment.

G — At the beginning of stage **G**, Neuron-6 has its light unblocked, the system now moves toward a perfect competition equilibrium. The absolute effective actuation of the system ($\mathbf{U_{avg}}$) as well as the efficiency $\mathbf{U_{avg}/V_{avg}}$ is at its maxiumum when all energy levels are exactly equal. A close up of Fig. 10 in such a state would produce the plot shown in Fig. 9.

H–M — From stage **H** onwards, the lights are again blocked in the same order as they were unblocked. At the beginning of stage **H** Light-1 is blocked, Neuron-1 rapidly becomes dormant (after only three failed spikes), with its energy at minimum. The same process is repeated for Neuron-2 through to Neuron-5, at each stage an increase in activity can be observed in the other neurons. This corresponds to the compensation seen in $\mathbf{U_{avg}}$ in Fig. 10 (b). By the beginning of stage **L** only Neuron-6 remains effective. Its output and energy response mirrors that of Neuron-1 in stage **B**. Finally during stage **M** Neuron-6 is blocked and the system's response is again reduced to probing its environment for a response.

VI. Conclusion

In this paper we have presented a novel, biologically-inspired control network, mNCC, that is robust to noise, element mismatch, network size, and actuator performance. The control system does not require finely tuned operating points, but rather utilizes both competition and cooperation between the actuators in order to obtain a stable operating state. In order to design an efficient neuromorphic circuit that would allow the implementation of mNCC we utilized the memristor, operating in two distinct states. The memristor facilitates low-power analog computing while the emergent properties of the control network itself subverts the other problems associated with analog computing. We have demonstrated the functionality of mNCC using Matlab simulations and verified its robustness to a number of nonidealities. The robustness of this architecture is critical if it is to be useful in real-world applications and implemented in silicon with nonideal components.

Acknowledgment

This work was supported by Australian Research Council grants DP0881219, DP0986025 and the University of Western Sydney.

References

[1] H. Wagner, "Flight performance and visual control of flight of the free-flying housefly (Musca domestica L.) II. pursuit of targets," *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, vol. 312, no. 1158, pp. 553–579, 1986.

[2] R. Zbikowski, "Fly like a fly [micro-air vehicle]," *Spectrum, IEEE*, vol. 42, no. 11, pp. 46–51, 2005.

[3] J. Perge, K. Koch, R. Miller, P. Sterling, and V. Balasubramanian, "How the optic nerve allocates space, energy capacity, and information," *The Journal of Neuroscience*, vol. 29, no. 24, pp. 7917–7928, 2009.

[4] Q. Wen and D. Chklovskii, "To myelinate or not to myelinate?" *New Aspects of Axonal Structure and Function*, pp. 103–113, 2010.

[5] A. Borst, J. Haag, and D. Reiff, "Fly motion vision," *Annual review of neuroscience*, vol. 33, pp. 49–70, 2010.

[6] K. Farrow, A. Borst, and J. Haag, "Sharing receptive fields with your neighbors: tuning the vertical system cells to wide field motion," *The Journal of Neuroscience*, vol. 25, no. 15, pp. 3985–3993, 2005.

[7] A. Borst and J. Haag, "Optic flow processing in the cockpit of the fly," *Cold Spring Harbor Monograph Archive*, vol. 49, no. 0, pp. 101–122, 2007.

[8] S. Stepney, F. Polack, and H. Turner, "Engineering Emergence," in *11*th *IEEE International Conference on Engineering of Complex Computer Systems, ICECCS*, 2006, pp. 89–97, stanford, CA, USA.

[9] T. Lehmann and R. Woodburn, "Biologically-inspired on-chip learning in pulsed neural networks," *Analog Integrated Circuits and Signal Processing*, vol. 18, no. 2, pp. 117–131, 1999.

[10] A. Russell, G. Orchard, Y. Dong, S. Mihalas and, E. Niebur, J. Tapson, and R. Etienne-Cummings, "Optimization methods for spiking neurons and networks," *IEEE Transactions on Neural Networks*, vol. 21, no. 12, pp. 1950–1962, 2010.

[11] A. van Schaik, T. Hamilton, and C. Jin, "Silicon models of the auditory pathway," *Computational Models of the Auditory System, Springer Handbook of Auditory Research*, vol. 35, pp. 261–276, 2010.

[12] G. Indiveri, B. Linares-Barranco, T. J. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen, "Neuromorphic silicon neuron circuits," *Frontiers in Neuroscience*, vol. 5, 2011, *http://tinyurl.com/neurosi*.

[13] A. van Schaik, "Building blocks for electronic spiking neural networks," *Neural Networks*, vol. 14, no. 6-7, pp. 617–628, 2001.

[14] O. Kavehei, A. Iqbal, Y. S. Kim, K. Eshraghian, S. F. Al-Sarawi, and D. Abbott, "The fourth element: characteristics, modelling and electromagnetic theory of the memristor," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, vol. 466, no. 2120, pp. 2175–2202, 2010.

[15] L. O. Chua and S. M. Kang, "Memristive devices and systems," *Proceedings of the IEEE*, vol. 64, no. 2, pp. 209–223, 1976.

[16] D. Strukov, G. Snider, D. Stewart, and R. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, pp. 80–83, 2008.

[17] R. Waser and M. Aono, "Nanoionics-based resistive switching memories," *Nature Materials*, vol. 6, no. 11, pp. 833–840, 2007.

[18] O. Kavehei, S. Al-Sarawi, K. Cho, N. Iannella, S. Kim, K. Eshraghian, and D. Abbott, "Memristor-based synaptic networks and logical operations using in-situ computing," in *International Conference Series on Intelligent Sensors, Sensor Networks and Information Processing, ISSNIP*, 2011, arxiv preprint arXiv:1108.4182.

[19] S. Jo, K. Kim, T. Chang, S. Gaba, and W. Lu, "Si memristive devices applied to memory and neuromorphic circuits," in *Proceedings of IEEE International Symposium on Circuits and Systems, ISCAS*, 2010, pp. 13–16.

[20] J. Borghetti, G. Snider, P. Kuekes, J. Yang, D. Stewart, and R. Williams, "'Memristive' switches enable 'stateful' logic operations via material implication," *Nature*, vol. 464, no. 7290, pp. 873–876, 2010.

[21] A. Stotland and M. Di Ventra, "Stochastic memory: Memory enhancement due to noise," *Physical Review E*, 2011, (accepted), arxiv preprint arXiv:1104.4485.

[22] R. Wang, C. Jin, A. McEwan, and A. van Schaik, "A programmable axonal propagation delay circuit for time-delay spiking neural networks," in *IEEE International Symposium on Circuits and Systems, ISCAS*, 2011, pp. 869–872.

[23] N. Brunel and D. Hansel, "How noise affects the synchronization properties of recurrent networks of inhibitory neurons," *Neural Computation*, vol. 18, no. 5, pp. 1066–1110, 2006.

[24] J. Tapson and A. van Schaik, "An asynchronous parallel neuromorphic ADC architecture," in *IEEE International Symposium on Circuits and Systems, ISCAS*, 2012 (accepted).