



On spike-timing-dependent-plasticity, memristive devices, and building a self-learning visual cortex

Carlos Zamarreño-Ramos¹, Luis A. Camuñas-Mesa¹, Jose A. Pérez-Carrasco¹, Timothée Masquelier², Teresa Serrano-Gotarredona¹ and Bernabé Linares-Barranco^{1*}

¹ Mixed Signal Design, Instituto de Microelectrónica de Sevilla (IMSE-CNM-CSIC), Sevilla, Spain

² Computational Neuroscience Group, Universitat Pompeu Fabra, Barcelona, Spain

Edited by:

Gert Cauwenberghs, University of California San Diego, USA

Reviewed by:

Venkat Rangan, Qualcomm

Incorporated, USA

Siddharth Joshi, University of California San Diego, USA

Dan Hammerstrom, Portland State University, USA

*Correspondence:

Bernabé Linares-Barranco, IMSE-CNM-CSIC, Av. Américo Vespucio s/n, 41092 Sevilla.

e-mail: bernabe@imse-cnm.csic.es

In this paper we present a very exciting overlap between emergent nanotechnology and neuroscience, which has been discovered by neuromorphic engineers. Specifically, we are linking one type of memristor nanotechnology devices to the biological synaptic update rule known as spike-time-dependent-plasticity (STDP) found in real biological synapses. Understanding this link allows neuromorphic engineers to develop circuit architectures that use this type of memristors to artificially emulate parts of the visual cortex. We focus on the type of memristors referred to as voltage or flux driven memristors and focus our discussions on a behavioral macro-model for such devices. The implementations result in fully asynchronous architectures with neurons sending their action potentials not only forward but also backward. One critical aspect is to use neurons that generate spikes of specific shapes. We will see how by changing the shapes of the neuron action potential spikes we can tune and manipulate the STDP learning rules for both excitatory and inhibitory synapses. We will see how neurons and memristors can be interconnected to achieve large scale spiking learning systems, that follow a type of multiplicative STDP learning rule. We will briefly extend the architectures to use three-terminal transistors with similar memristive behavior. We will illustrate how a V1 visual cortex layer can be assembled and how it is capable of learning to extract orientations from visual data coming from a real artificial CMOS spiking retina observing real life scenes. Finally, we will discuss limitations of currently available memristors. The results presented are based on behavioral simulations and do not take into account non-idealities of devices and interconnects. The aim of this paper is to present, in a tutorial manner, an initial framework for the possible development of fully asynchronous STDP learning neuromorphic architectures exploiting two or three-terminal memristive type devices. All files used for the simulations are made available through the journal web site¹.

Keywords: STDP, memristor, synapses, spikes, learning, nanotechnology, visual cortex, neural network

1 INTRODUCTION

Neuromorphic engineering² is a new interdisciplinary discipline that takes inspiration from biology, physics, mathematics, computer science, and engineering to design artificial neural systems, such as vision systems, head-eye systems, auditory processors, and autonomous robots, the physical architecture, and design principles of which are based on those of biological nervous systems. The term neuromorphic was coined by Carver Mead, in the late 1980s (Mead, 1989) to describe very-large-scale integration (VLSI) systems containing electronic analog circuits that mimic neurobiological architectures present in the nervous system. In recent times the term neuromorphic has been used to describe both analog, digital, or mixed-mode analog/digital VLSI systems that implement models of neural systems (for perception, motor control, or sensory processing) and also software algorithms. A key aspect of neuromorphic design is understanding how the morphology

of individual neurons, circuits, and overall architectures creates desirable computations, affects how information is represented, influences robustness to damage, incorporates learning and development, and facilitates evolutionary change.

It is obvious that interdisciplinary research broadens our view of particular problems yielding fresh and possibly unexpected insights. This is the case of neuromorphic engineering, where technology and neuroscience cross-fertilize each other. One example of this is the recent impact of fabricated memristor devices (Strukov et al., 2008; Borghetti et al., 2009; Jo et al., 2009, 2010), postulated since 1971 (Chua, 1971; Chua and Kang, 1976; Chua et al., 1987), thanks to research in nanotechnology electronics. Another is the mechanism known as spike-time-dependent-plasticity (STDP; Gerstner et al., 1993, 1996; Delorme et al., 2001; Rao and Sejnowski, 2001; Guyonneau et al., 2004; Porr and Wörgötter, 2004; Masquelier and Thorpe, 2007, 2010; Young, 2007; Finelli et al., 2008; Masquelier et al., 2008, 2009a,b; Weidenbacher and Neumann, 2008; Sjöström and Gerstner, 2010) which describes a neuronal synaptic learning mechanism that refines the traditional Hebbian synaptic plasticity proposed in 1949 (Hebb, 1949). These are very different subjects from

¹A compressed zip file *AdditionalMaterial.zip* can be downloaded from the journal web site at http://www.frontiersin.org/Neuromorphic_Engineering/10.3389/fnins.2011.00026/abstract

²<http://en.wikipedia.org/wiki/Neuromorphic>

relatively unrelated disciplines (nanotechnology, biology, and computer science), which have nevertheless recently been drawn together by researchers in neuromorphic engineering (Snider, 2007, 2008; Linares-Barranco and Serrano-Gotarredona, 2009). STDP was originally postulated as a family of computer learning algorithms (Gerstner et al., 1993), and is being used by the machine intelligence and computational neuroscience community (Delorme et al., 2001; Guyonneau et al., 2004; Masquelier and Thorpe, 2007, 2010; Young, 2007; Finelli et al., 2008; Masquelier et al., 2008, 2009a,b; Weidenbacher and Neumann, 2008). At the same time its biological and physiological foundations have been reasonably well established during the past decade (Markram et al., 1997; Bi and Poo, 1998, 2001; Zhang et al., 1998; Feldman, 2000; Mu and Poo, 2006; Cassenaer and Laurent, 2007; Jacob et al., 2007). If memristance and STDP can be related, then (a) recent discoveries in nanophysics and nanoelectronic principles may shed new light on the intricate molecular and physiological mechanisms behind STDP in neuroscience, and (b) new neuromorphic-like computers built out of nanotechnology memristive devices could incorporate biological STDP mechanisms, yielding a new generation of self-adaptive ultra-high-dense intelligent machines. Here we explain how by combining memristance models with the electrical wave signals of neural impulses (spikes) converging from pre- and post-synaptic neurons into a synaptic junction, STDP behavior emerges naturally (Linares-Barranco and Serrano-Gotarredona, 2009). This helps us to understand how neural and memristance parameters modulate STDP, and may offer new insights to neurophysiologists searching for the ultimate physiological mechanisms responsible for STDP in biological synapses. At the same time, it also provides a direct means of incorporating STDP learning mechanisms into a new generation of nanotechnology computers employing memristors. Here we focus on this second aspect.

In this paper we first quickly review STDP and memristor concepts. Then we explain how the memristance mechanism, and one particular formulation of it, can explain the experimental characterization of the STDP phenomena in biological synapses. We will see how the shape of action potentials is a crucial component which influences and defines the mathematical learning of STDP, and how by changing action potential shapes the STDP learning rule can be modulated and changed. The paper then concentrates on proposing circuit techniques and architectures for achieving STDP learning neural systems using memristors as synapses. We will see that ideal voltage/flux driven memristors implement a particular type of multiplicative STDP (mSTDP) with quadratic law. We will show how excitatory and inhibitory learning synapses can be implemented and efficiently simulated using a memristor macro-model with neurons described behaviorally in electric circuit simulators. We will also briefly extend the discussion from two-terminal passive memristor devices to three-terminal active field effect transistor (FET) devices with memristive-like adaptation mechanisms. We will then show how to implement a prototype V1 layer mimicking the early processing part of the visual cortex. We will use real data collected from a fabricated Spiking retina chip observing life scenes and use it to train this artificial simulated V1 layer. As a result, this artificial V1 layer learns to become orientation sensitive, like its biological counterpart.

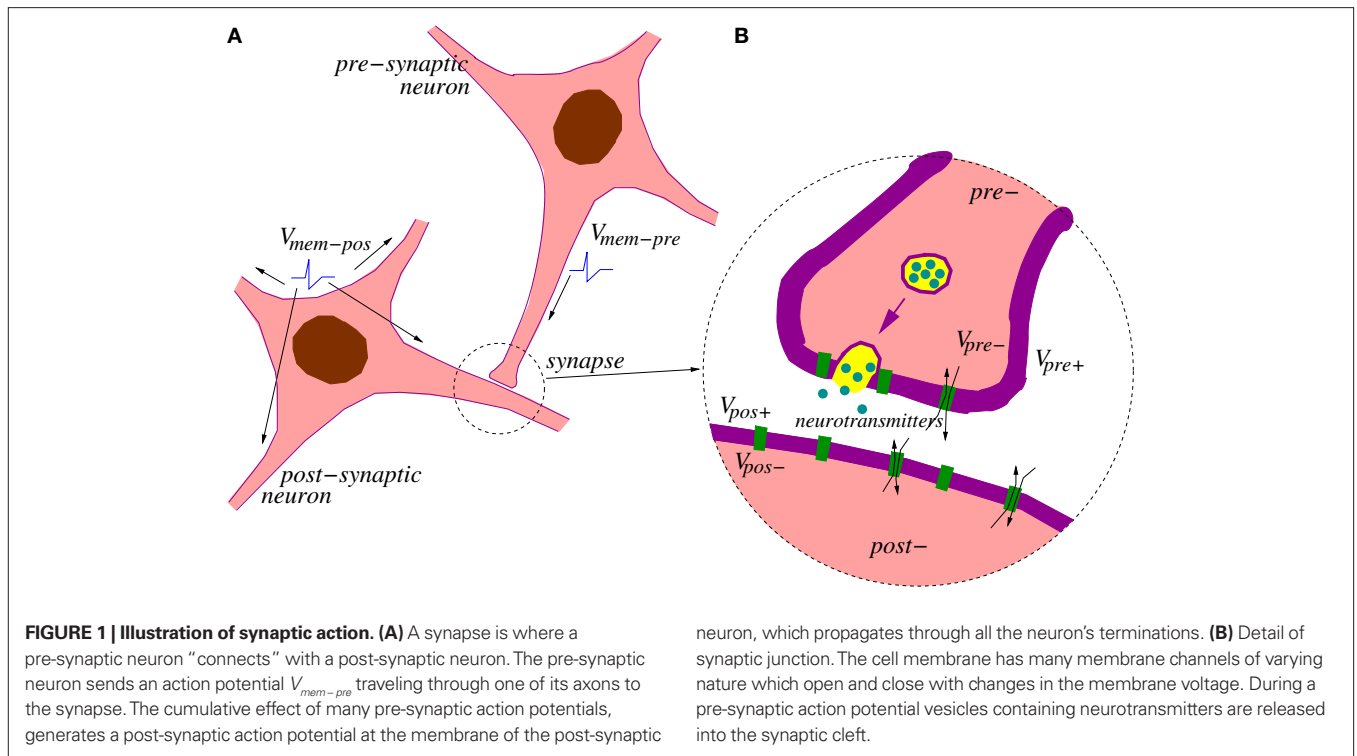
2 STDP

Spike-time-dependent-plasticity is a family of learning mechanisms originally postulated in the context of artificial machine learning algorithms (or computational neuroscience), exploiting spike-based computations (as in brains) with great emphasis on the relative timings of spikes. Gerstner started to report the first-spike-timing-dependent learning algorithms (Gerstner et al., 1993, 1996) in 1993. STDP has been shown to be better than Hebbian correlation-based plasticity at explaining cortical phenomena (Young, 2007; Finelli et al., 2008), and has been proven successful in learning hidden spiking patterns (Masquelier et al., 2008) or performing competitive spike pattern learning (Masquelier et al., 2009a). Astonishingly, experimental evidence of STDP has been reported by neuroscience groups during the past decade (Markram et al., 1997; Bi and Poo, 1998, 2001; Zhang et al., 1998; Feldman, 2000; Mu and Poo, 2006; Cassenaer and Laurent, 2007; Jacob et al., 2007), so today we can state that the physiological existence of STDP has been reasonably well established³.

However, the full implications of the molecular and electrochemical principles behind STDP are still under debate (Rubin et al., 2005). Before describing STDP mathematically, let us first explain how neurons interchange information and what the synaptic connections are.

Figure 1 illustrates two neurons connected by a synapse. The pre-synaptic neuron is sending a pre-synaptic spike $V_{mem-pre}(t)$ through one of its axons to the synaptic junction. Neural spikes are membrane voltages from the outside of the cellular membrane V_{pre+} with respect to the inside V_{pre-} . Thus $V_{mem-pre} = V_{pre+} - V_{pre-}$ and $V_{mem-pos} = V_{pos+} - V_{pos-}$. The “large” membrane voltages during a spike (in the order of a hundred mV) cause a variety of selective molecular membrane channels to open and close allowing many ionic and molecular substances to flow, or preventing them from flowing through the membrane. At the same time, synaptic vesicles inside the pre-synaptic cell containing “packages” of neurotransmitters fuse with the membrane in such a way that these “packages” are released into the synaptic cleft (the inter cellular space between both neurons at the synaptic junction). Neurotransmitters are collected in part by the post-synaptic membrane, contributing to a change in its membrane conductivity. The cumulative effect of pre-synaptic spikes (coming from this or other pre-synaptic neurons) will eventually trigger the generation of a new spike at the post-synaptic neuron. Each synapse is characterized by a “synaptic strength” (or weight) w which determines the efficacy of a pre-synaptic spike in contributing to this cumulative action at the post-synaptic neuron. This weight w could well be interpreted as the size and/or number of neurotransmitter packages released during a pre-synaptic spike. However, for our analyses, we will interpret w more generally as some kind of structural parameter of the synapse (like the amount of one or more metabolic substances) that directly controls the efficacy of this synapse per spike. The synaptic weight w is considered to be non-volatile and analog in nature, but it changes in time as a function of the spiking activity of pre- and post-synaptic neurons. This phenomenon was originally

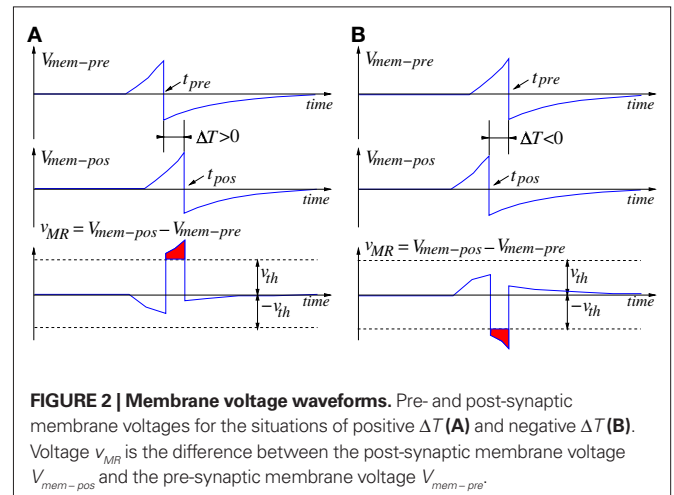
³For a historical overview on how STDP research evolved independently among computational and experimentalist groups, please refer to the last paragraph in Sjöström and Gerstner (2010).



observed and reported in 1949 by Hebb (1949), who introduced his Hebbian learning postulate: “When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A’s efficiency, as one of the cells firing B, is increased.” Traditionally, this has been described by computational neuroscientists and machine learning computer engineers as producing an increment in synaptic weight Δw proportional to the product of the mean firing rates of pre- and post-synaptic neurons. STDP is a refinement of this 1949 rule which takes into account the precise relative timing of individual pre- and post-synaptic spikes, and not their average rates over time. In STDP the change in synaptic weight Δw is expressed as a function of the time difference between the post-synaptic spike at t_{pos} and the pre-synaptic spike at t_{pre} (see Figure 2). Specifically, as is shown in Figure 3, $\Delta w = \xi(\Delta T)$, with $\Delta T = t_{pos} - t_{pre}$. The shape of the STDP function ξ can be interpolated from experimental data from Bi and Poo (2001) as shown in Figure 3A. For positive ΔT (that is to say, the pre-synaptic spike has a highly relevant role in producing the post-synaptic spike) there will be a potentiation of synaptic weight $\Delta w > 0$, which will be stronger as $|\Delta T|$ reduces. For negative ΔT (that is to say, the pre-synaptic spike is highly irrelevant for the generation of the post-synaptic spike), there will be a depression of synaptic weight $\Delta w < 0$, which will be stronger as $|\Delta T|$ reduces. Bi and Poo concluded that they had observed an asymmetric critical window for ΔT of about $\pm 40-80$ ms for synaptic modification to take place. Mathematically, this $\xi(\Delta T)$ STDP learning function is described by computational neuroscientists as

$$\xi(\Delta T) = \begin{cases} a^+ e^{-\Delta T/\tau^+} & \text{if } \Delta T > 0 \\ -a^- e^{\Delta T/\tau^-} & \text{if } \Delta T < 0 \end{cases} \quad (1)$$

neuron, which propagates through all the neuron’s terminations. (B) Detail of synaptic junction. The cell membrane has many membrane channels of varying nature which open and close with changes in the membrane voltage. During a pre-synaptic action potential vesicles containing neurotransmitters are released into the synaptic cleft.



2.1 STDP VERSUS ANTI-STDP

The STDP learning function $\xi(\Delta T)$ as defined in Figures 3A,B is useful for synapses with positive weights. In these cases, weight w is strengthened if it is increased ($\Delta w > 0$) when $\Delta T > 0$, and vice versa. However, if the weight is negative ($w < 0$), as in some inhibitory synapse implementations, the STDP learning function in Figure 3B is not appropriate because an increase in weight ($\Delta w > 0$) would weaken the strength of the synapse, and vice versa. For negative weight synapses an STDP learning function with a shape similar to that shown in Figure 3C (Lubenov and Siapas, 2008) is required. In this case, the synapse is strengthened by decreasing its weight ($\Delta w < 0$), which should happen for $\Delta T > 0$. Let us call this an Anti-STDP synaptic update or learning function. Other more exotic shapes for $\xi(\Delta T)$ are also possible, as we will discuss later in Sections 4.1 and 5.1.

2.2 ADDITIVE VERSUS MULTIPLICATIVE STDP

Most of the present day literature on STDP presents a learning function ξ which depends on ΔT but not on the actual weight value w . This type of weight-independent STDP learning rule is usually known as “additive STDP.” Additive STDP requires the weight values to be bounded to an interval because weights will stabilize at one of their boundary values (van Rossum et al., 2000; Rubin et al., 2001).

On the other hand, in “multiplicative STDP” or mSTDP (van Rossum et al., 2000; Rubin et al., 2001; Güttig et al., 2003) the learning function is also a function of the actual weight value $\xi_m(w, \Delta T)$. Furthermore, there usually appears a weight dependent factor which multiplies the original additive STDP learning function ξ_a , and which may generally be different for the positive ($\Delta T > 0$) and negative ($\Delta T < 0$) sides

$$\xi_m(w, \Delta T) = F(w, \text{sign}(\Delta T))\xi_a(\Delta T) \tag{2}$$

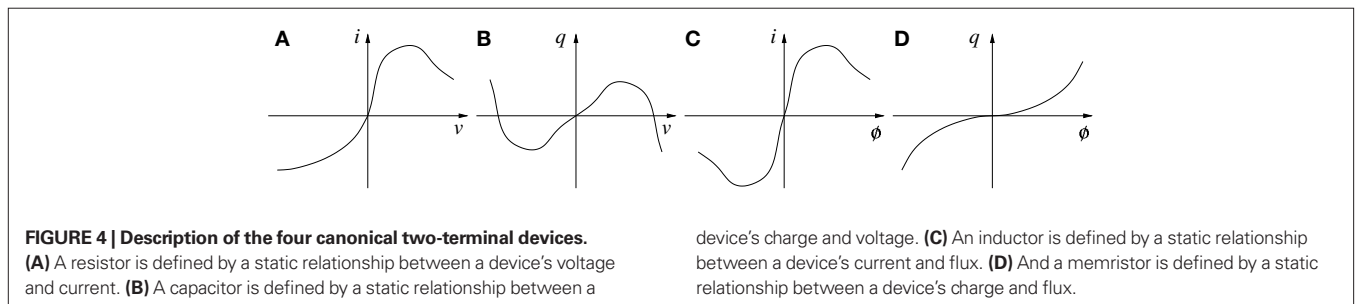
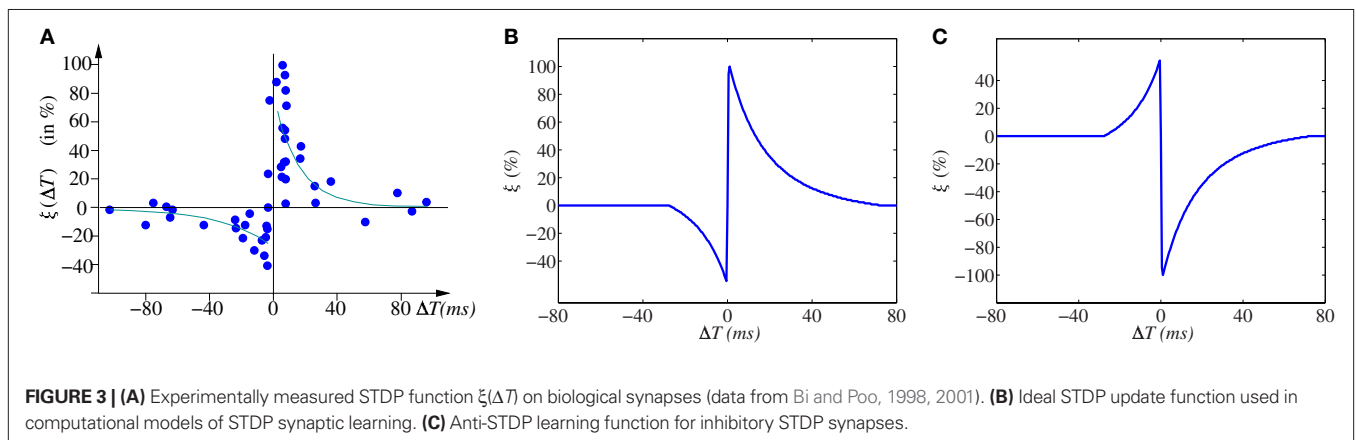
In mSTDP weights can stabilize to intermediate values inside the boundary definitions. Thus, it is often not even necessary to enforce boundary conditions for the weight values (van Rossum et al., 2000). As we will see later in Section 4.2, for the particular type of memristive devices we are analyzing (voltage/flux driven) the resulting STDP learning rule will be of a multiplicative type with function F having a quadratic dependence with w , while the values of w are kept bounded.

3 MEMRISTANCE

Memristance was postulated by Chua (1971) based on circuit theoretical reasonings. According to circuit theoretical fundamentals, there are four basic electrical quantities (Chua et al., 1987):

(1) voltage difference between two-terminals “ v ,” (2) current flowing through into a device terminal “ i ,” (3) charge flowing through a device terminal or integral of current $q = \int i(\tau)d\tau$, and (4) flux or integral of voltage $\phi = \int v(\tau)d\tau$. A two-terminal device is said to be canonical (Chua et al., 1987) if either two of the four basic electrical quantities are related by a static⁴ relationship, as shown in **Figure 4**. A resistor has a static relationship between terminal voltage v and device current i , as shown in **Figure 4A**. A capacitor shows a static relationship between charge q and voltage v , as shown in **Figure 4B**. An inductor has a static relationship between its current i and flux ϕ , as shown in **Figure 4C**. These three devices have been very well known since the origins of Electronics and Electricity. However, there are other possibilities for combining the four basic electrical quantities: (q, i) , (v, ϕ) , and (q, ϕ) . Ignoring the combinations of a quantity with its time derivative leaves us with one single additional possibility: (q, ϕ) . This reasoning led Chua to postulate the existence of a fourth basic two-terminal element, which he called the *Memristor*. The memristor would show a static relationship between charge q and flux ϕ , as shown in **Figure 4D**. If the q versus ϕ relationship is linear, the *memristor* degenerates into a linear resistor. *Memristors* behave as resistances in which the resistance changes through some of the basic electrical quantities, and is somehow *memorized*. The simple concept of *memristance* as defined in **Figure 4D** can be extended to refer to any device exhibiting resistive behavior whose resistance can change through

⁴By “static” we mean it is not altered by changes of the above electrical quantities, or by their history, integrals, derivatives, etc. These “static” curves can, however, be time varying if the change is caused by an external agent. For example, a motor driven potentiometer would have a “static” i/v curve that is time varying.



some of the four basic electrical quantities, but at the same time exhibiting *memory* for that resistance. In that case, more elaborate mathematical descriptions are required (Chua and Kang, 1976).

Memristance has recently been demonstrated (with extraordinary impact among the research community) in nanoscale two-terminal devices, such as certain titanium-dioxide (Strukov et al., 2008; Borghetti et al., 2009) and amorphous Silicon (Jo et al., 2009) cross-point switches. However, memristive devices were reported earlier by other groups (Argall, 1968; Swaroop et al., 1998). Memristance arises naturally in nanoscale devices because small voltages can yield enormous electric fields that produce the motion of charged atomic or molecular species, changing structural properties of a device (such as its doping profile) while it operates. Memristors are asymmetric two-terminal passive devices. Consequently, their circuit symbol must indicate somehow their polarity. **Figure 5A** shows two possible symbols. Here we will consider one particular subset of memristors described by (Snider, 2007, 2008)

$$i_{MR} = g(w, v_{MR})v_{MR} \quad (3)$$

$$\dot{w} = f(w, v_{MR}) \quad (4)$$

where w is some physical (structural) parameter, i_{MR} is the current through the device, v_{MR} the voltage drop across it, and g is its (non-linear) conductance. Since the change of structural parameter w is driven by voltage v_{MR} , we say this memristor is voltage (or flux) driven. The group at the Michigan University claims to have fabricated a memristor of this kind (Jo et al., 2009, 2010). If function $f()$ in eq. (4) is driven by memristor current i_{MR} , then we say the memristor is current (or charge) driven. The HP group tends to model their memristor as one of this type (Strukov et al., 2008; Borghetti et al., 2009).

In memristive nanoscale devices, function f may describe ionic drift under electric fields. Although this may conceivably be modeled by a linear dependence of f with voltage v_{MR} (Strukov et al., 2008), it is clear that in reality such dependence is more likely to grow exponentially and/or include a threshold barrier v_{th} (Jo et al., 2010). For our discussions, let us assume the following dependence

$$f(v_{MR}) = I_o \text{sign}(v_{MR}) \left[e^{v_{MR}/v_o} - e^{-v_{MR}/v_o} \right] \text{ if } |v_{MR}| > v_{th} \quad (5)$$

$$f(v_{MR}) = 0 \quad \text{otherwise} \quad (6)$$

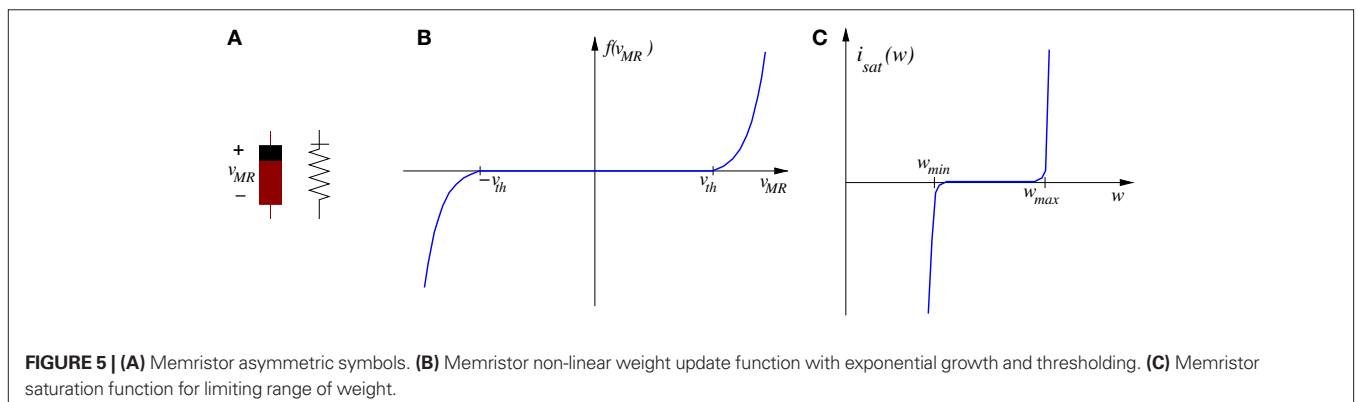
where I_o and v_o are parameters which may or may not depend on w . This shape of f is shown in **Figure 5B**. Many other mathematical formulations can be used (Chua and Kang, 1976). In order to relate memristance to biological STDP, as will be done in Section 4, we need a voltage/flux controlled memristor with thresholding behavior, exponential behavior beyond threshold, and bidirectional behavior (i.e., to be able to increment and decrement w). Since a memristor has polarity, as indicated in **Figure 5A**, we need to establish a criterion to assign one of the terminals as the positive terminal. The criterion adopted to assign polarity is that if a sufficiently large *positive* voltage v_{MR} is applied to the memristor (i.e., larger than its positive threshold), it will increase its conductance. Otherwise, if a sufficiently large *negative* voltage v_{MR} is applied (i.e., increasing beyond its negative threshold), it will decrease its conductance.

3.1 MEMRISTOR MACRO-MODEL FOR TWO-TERMINAL DEVICES

A macro-model of a device is a behavioral model made of circuit elements (ideal or not) that describes its behavior. Some circuit simulators allow a device to be defined mathematically using for example AHDL or Verilog-A circuit description languages. However, if the device can be described with a macro-model circuit, this will have some advantages⁵. (1) First, it uses already built-in components providing faster simulations; (2) second, as it is made of circuit elements it gives (analog) circuit designers a richer intuitive insight into how it works and performs, and how to improve it for specific goals; (3) it is very intuitive when adding parasitic components (resistors and capacitors) to aid in the convergence of the simulator's internal algorithms; (4) and if care is taken to keep the operating voltages and currents of internal nodes to the levels the simulator expects from conventional circuits, simulations converge easier and faster.

Reported memristors (as in Strukov et al., 2008; Jo et al., 2010) adhere very well to the "moving wall model" (see **Figure 6A**), where the wall position w separates two different resistive regions in

⁵Note that our aim in providing a macro-model circuit is to have a means of simulating large number of memristors efficiently in a circuit simulator, and hence take advantage of its computational efficiency and ease of use. Our aim is not to provide a means of building physical circuits out of such macro-models (as Chua, 1971, did in the past using mutator circuits). A direct physical circuit realization of **Figure 6B** would result, for example, in leaks of the memory value w due to unavoidable current leak paths in parallel with the capacitor.



series, and moves depending on device current or voltage. A circuit macro-model that implements eqs. (3–6) is shown in **Figure 6B**. It comprises a controlled resistor in which resistance R is controlled linearly by internal state voltage w , $R(w) = k_R \times (w + w_o)$. Voltage w represents the “structural” parameter of the wall position, which is bounded to $[0, L]^6$. Component **NOTA** is a non-linear differential-input voltage-controlled current source (transconductor), also known as non-linear operational transconductance amplifier (OTA) which provides an output current $i_g(v_{MR}) = f(v_{MR})$ controlled by input differential voltage v_{MR} , as given in eqs. (5–6) and **Figure 5B**. Non-linear element g_{sat} is a non-linear resistor with a shape as shown in **Figure 5C**, which limits the range of the resistance $R(w)$ to $[R_{min}, R_{max}]$, thus keeping w inside its natural boundary $[0, L]$. Consequently, the macro-model circuit in **Figure 6** is mathematically described by^{7,8}

$$v_{MR} = R(w)i_{MR} \tag{7}$$

$$R(w) = k_R (w + w_o) \tag{8}$$

$$C_{MR}\dot{w} = i_g(v_{MR}) - i_{sat}(w) \tag{9}$$

⁶For current/charge driven memristors, the boundary constraint was modeled by adding a multiplicative windowing function to allow for non-linear drift (Biolek et al., 2009; Joglekar and Wolf, 2009).

⁷These equations are similar to Di Ventra’s macro-model (Pershin and Di Ventra, 2010; Pershin and Di Ventra, in press), except that we use an additive rather than a multiplicative term g_{sat} to constrain the interval for w . A multiplicative term may result in the stacking of w at one of its limits, as mentioned by Biolek et al. (2009).

⁸We originally reported the macro-model of **Figure 6B** in May 2010 (Pérez-Carrasco et al., 2010a). Simultaneously, a mathematically equivalent macro-model was reported by Shin et al. (2010), based on Charge-Flux Constitutive Relationships.

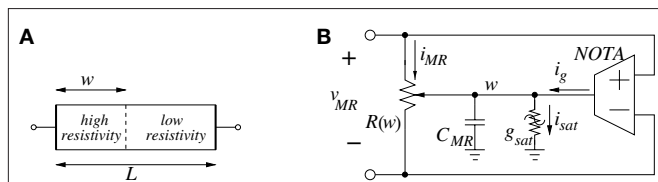


FIGURE 6 | (A) Memristor “moving wall” model. **(B)** Simple macro-model circuit for electrical simulations.

Parameter k_R scales between the voltage domain range of w (usually within a few volts, for proper simulator convergence) and the resistance domain range of R which can be as high as hundreds of Megaohms (Jo et al., 2009, 2010). **Figure 7** shows the simulation results⁹ of a memristor connected in series with a 5-M Ω resistance stimulated with a 2-V sinusoid of decreasing frequency from 5 KHz to 0 Hz in 26 cycles. Maximum and minimum memristor resistance limits were $R_{max} = 100 \text{ M}\Omega$ and $R_{min} = 10 \text{ M}\Omega$, symmetric threshold voltages were $|v_{th}| = 1 \text{ V}$, the exponential $f(v_{MR})$ was characterized by $I_o = 10 \mu\text{A}$, $v_o = 0.1 \text{ V}$, $v_{th} = 1 \text{ V}$, and the other macro-model parameters were $w_{max} = 10 \text{ V}$, $w_{min} = -10 \text{ V}$, $k_R^{-1} = 222 \mu\text{A}$, $w_o = 12.2 \text{ V}$, and $C_{MR} = 10 \text{ mF}$.

4 RELATION BETWEEN STDP AND MEMRISTANCE

How can STDP be related to memristance? The key is to consider carefully the shape of the electric neural spikes (Linares-Barranco and Serrano-Gotarredona, 2009). The exact shape of neural spikes, usually called “action potentials” among neuroscientists, is difficult to measure precisely since the experimental setup influences strongly. Furthermore, different action potential shapes have been recorded for different types of neurons, although in general they all display a certain resemblance. For our discussion it suffices to assume a generic action potential shape with the following properties (see **Figure 8A**). During spike on-set, which happens during a time t_{ail}^+ , membrane voltage increases exponentially to a positive peak amplitude A_{mp}^+ . After this, it changes quickly to a peak negative amplitude $-A_{mp}^-$ and returns smoothly to its resting potential during a time t_{ail}^- . A shape of the type shown in **Figure 8A** can be expressed mathematically, for example, as

$$spk(t) = \begin{cases} A_{mp}^+ \frac{e^{t/\tau_{ail}^+} - e^{-t_{ail}^+/\tau_{ail}^+}}{1 - e^{-t_{ail}^+/\tau_{ail}^+}} & \text{if } -t_{ail}^+ < t < 0 \\ -A_{mp}^- \frac{e^{-t/\tau_{ail}^-} - e^{-t_{ail}^-/\tau_{ail}^-}}{1 - e^{-t_{ail}^-/\tau_{ail}^-}} & \text{if } 0 < t < t_{ail}^- \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

Parameters τ_{ail}^+ and τ_{ail}^- control the curvature of the on-set and off-set sides of the action potential.

⁹Spectre netlist files for these simulations are available in folder *Spectre_for_Fig7* in the *AdditionalMaterial* file, downloadable from the Journal web site.

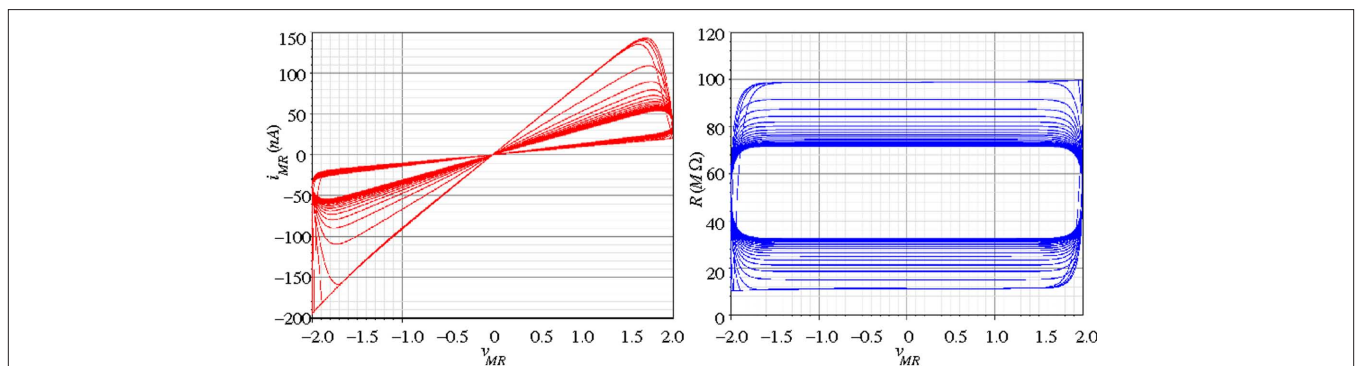
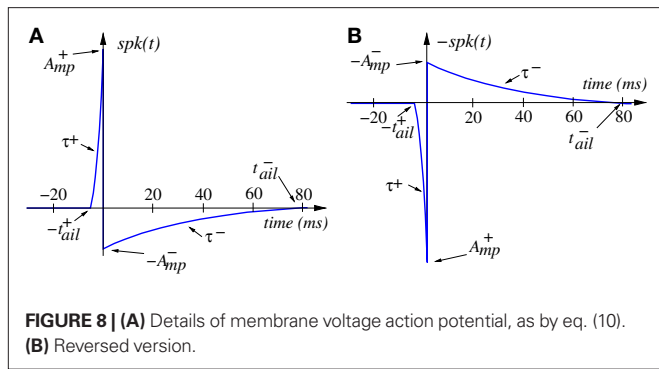


FIGURE 7 | Memristor simulations using the macro-model. Left, i/v curves obtained. Right, dependence of memristor time varying resistance with memristor voltage.



Consider the case of pre- and post-synaptic neurons in **Figure 1** being of the same type, and thus generating the same action potential shape, $spk(t)$ of eq. (10), when they fire. Axons and dendrites operate as transmission lines, so it is reasonable to expect some attenuation when the spikes arrive at the respective synapses. Let α_{pre} be the attenuation for the pre-synaptic spike $V_{mem-pos}(t) = \alpha_{pre} spk(t - t_{pre})$, and α_{pos} for the post-synaptic spike $V_{mem-pos}(t) = \alpha_{pos} spk(t - t_{pos})$. When both spikes are more or less simultaneously present at the two cell membranes of the synapse, then channels on both membranes are open. Consequently, in principle, it makes sense to assume that during such time there could be a path for substances in the inside of one cell to move directly to the inside of the other cell and vice versa. Furthermore, let us assume now that such motion of substances obeys a memristive law similar to those described by eqs. (3–6). This means, that we would have a two-terminal memristive device between the inner sides of the two cells; more specifically, between V_{pos-} and V_{pre-} in **Figure 1B**. Consequently, the memristor voltage would be $v_{MR} = V_{pre-} - V_{pos-}$. On the other hand, since the outside nodes of both membranes V_{pos+} and V_{pre+} are very close together, both voltages will be approximately equal, yielding

$$v_{MR}(t') \approx V_{mem-pos}(t') - V_{mem-pre}(t') \\ = \alpha_{pos} spk(t' - t_{pos}) - \alpha_{pre} spk(t' - t_{pre}) \quad (11)$$

A simple change of variables $t = t' - t_{pos}$ and recalling that $\Delta T = t_{pos} - t_{pre}$, results in

$$v_{MR}(t, \Delta T) = \alpha_{pos} spk(t) - \alpha_{pre} spk(t + \Delta T) \quad (12)$$

This memristor voltage v_{MR} is shown in **Figure 2** for the cases of ΔT being positive or negative. According to eqs. (5–6), memristive update will take place only if v_{MR} exceeds threshold v_{th} , as indicated by the red shaded area in **Figure 2**. As we postulated earlier, during this memristive update some amount of synaptic structural substance(s) Δw would be interchanged between the two sides of the synapse. The amount of substance Δw will ultimately affect the synaptic strength of this synapse. If this amount of synaptic structural substance interchanged between the two synaptic terminations obeys a memristive law as in eqs. (3–6), then from eq. (4)

$$\Delta w(\Delta T) = \int f(v_{MR}(t, \Delta T)) dt = \xi(\Delta T) \quad (13)$$

which is the red area of the shaded regions in **Figure 2**, previously amplified exponentially through function $f()$ of eqs. (5–6). Positive areas (above v_{th} , when $\Delta T > 0$) yield increments for w ($\Delta w > 0$), while

negative areas (below $-v_{th}$, when $\Delta T < 0$) result in decrements for w ($\Delta w < 0$). As $|\Delta T|$ approaches zero, the peak of the red area in v_{MR} is higher. Since this peak is amplified exponentially, the contribution for incrementing/decrementing w will be more pronounced as $|\Delta T|$ is reduced. The resulting function $\Delta w(\Delta T)$, computed using the memristor model through eq. (13) is shown in **Figure 3B**. Actually, it imitates the behavior of the STDP function ξ obtained by Bi and Poo from physiological experiments, which was shown in **Figure 3A**. For this numerical computation we used the following parameters: $\alpha_{pos} = 1$, $\alpha_{pre} = 0.9$, $v_{th} = A_{mp+} = 1$, $A_{mp-} = 0.25$, $v_o = 1/7$, $t_{ail+} = 5$ ms, $t_{ail-} = 75$ ms, $\tau^+ = 40$ ms, $\tau^- = 3$ ms. Making $\alpha_{pos} \neq \alpha_{pre}$ breaks the symmetry of function $\xi(\Delta T)$, and making them very different removes one of the branches in $\xi(\Delta T)$. It is possible to have more freedom to achieve a desired shape for $\xi(\Delta T)$ by setting $\alpha_{pos} = \alpha_{pre} = 1$, but instead shaping the spikes traveling forward and backward independently. Also, note that for ΔT values very close to zero $\Delta w(\Delta T)$ is approximately linear and crosses the origin. This is because as ΔT approaches zero, v_{MR} approaches zero for any t [see eq. (12)].

This result shows that a memristive type of mechanism could be behind the biological STDP phenomenon¹⁰.

4.1 INFLUENCE OF ACTION POTENTIAL SHAPE

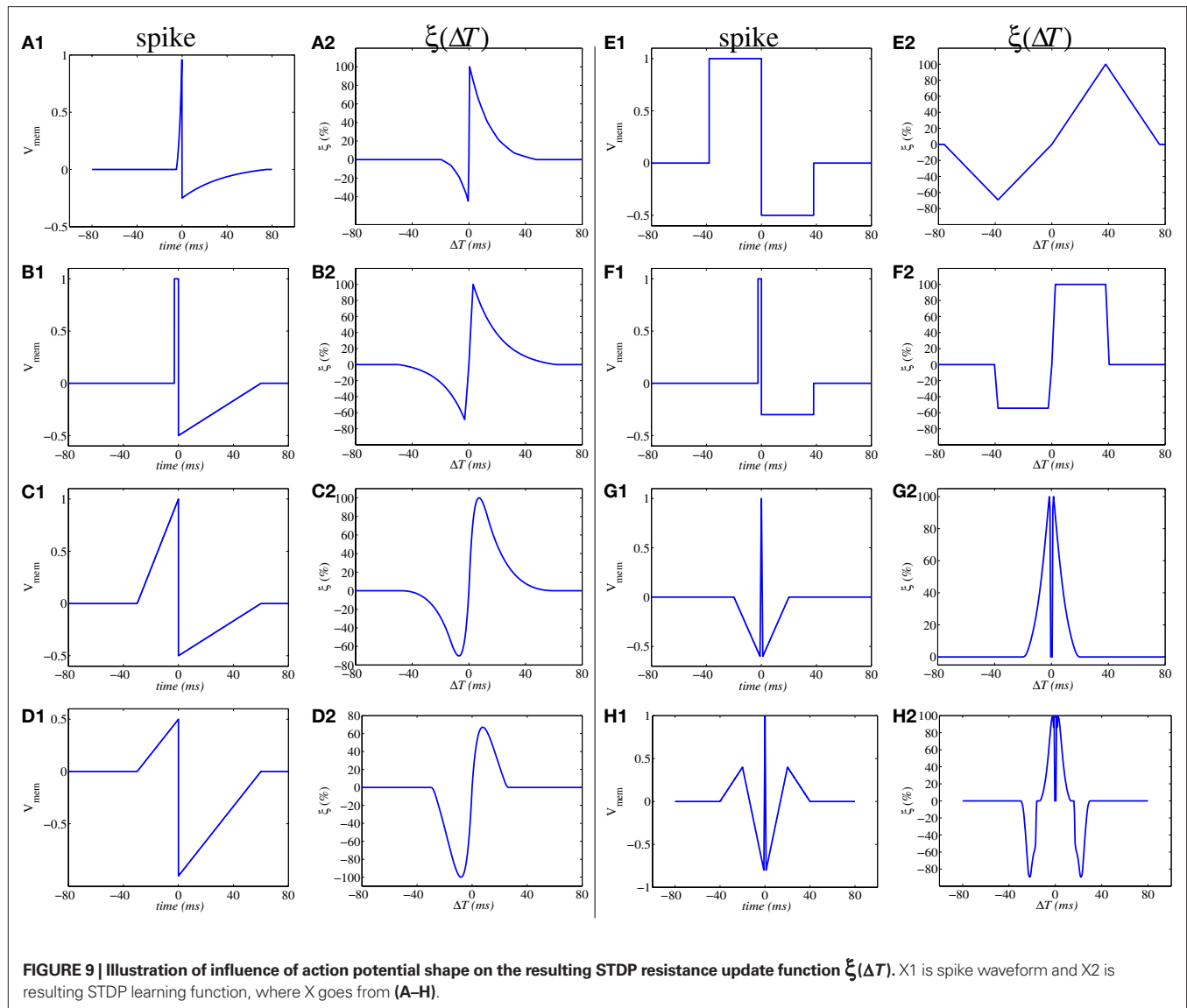
The shape of the action potential function $spk(t)$ strongly influences the shape of the resulting STDP function $\xi(\Delta T)$. As an illustration, **Figure 9** shows how for several shapes of action potentials (“spikes”) different STDP learning functions $\xi(\Delta T)$ are obtained¹¹. For example, if the exponential shape degenerates into a triangular type of shape, then the central region of $\xi(\Delta T)$ will display a smoother transition from the negative peak to the positive peak. Note that this would weaken learning for cases with small $|\Delta T|$. Making the positive peak of the spike smaller than the negative peak, makes the negative branch for $\xi(\Delta T)$ stronger than the positive branch. If the action potential is substituted by a rectangular shape signal, then the central region becomes linear and a saturation effect might occur. If the rectangular spike is made more symmetric, then $\xi(\Delta T)$ degenerates into a triangular type of shape, which is very different from the original biological STDP learning function. In general, to obtain a biological-like STDP learning function a narrow short positive pulse of large amplitude and a longer relaxing slowly decreasing negative tail are required. However, from a computational point of view, it might be more interesting to massage the shape of the “spikes” and tune the STDP learning function as desired.

4.2 MEMRISTORS IMPLEMENT A MULTIPLICATIVE TYPE OF STDP

It is worth mentioning here that memristors actually implement a multiplicative type of STDP. The reason is because, according to eq. (13), the structural parameter updates $\Delta w(\Delta T)$ follow an additive type of STDP rule, independent of w . Parameter w is the memristor “wall” position separating the low and high resistivity

¹⁰A few months after announcing our finding (Linares-Barranco and Serrano-Gotarredona, 2009), we became aware of a similar result (Afifi et al., 2009), which is a particular case of eqs. (10–12) for $\alpha_{pre} = \alpha_{pos} = 1$ and a different action potential shape (the one shown in **Figure 9B1**).

¹¹Matlab files for generating these figures are available in folder *Matlab_for_Fig9* in the *AdditionalMaterial* file, downloadable from the Journal web site.



regions (see **Figure 6A**). According to eq. (8), the memristor instantaneous resistance $R(w)$ is linear with w . Consequently, the memristive STDP resistance update $\Delta R(\Delta T) = k_R \times \Delta w(\Delta T) = k_R \xi(\Delta T)$ will follow an additive STDP update rule as well, independent of the actual value of R . However, as we will see in the next Section, when memristors (or resistors) are used as synapses in a neural circuit, the synaptic strength of such synapses is proportional to their conductance $G = 1/R$, because as conductance increases more current will be delivered to the post-synaptic neuron. Consequently, synaptic strength update is given by

$$\Delta G(\Delta T) = -\frac{\Delta R(\Delta T)}{R^2} = -G^2 k_R \xi(\Delta T) \quad (14)$$

which is quadratically proportional to the actual conductance. Memristors will therefore yield larger update steps for high conductances, but smaller steps for low conductances. This suggests that, before training, weights (conductances) should be initialized to rather high values, so that as learning progresses the updates tend to become smaller.

Note that the fact that memristors implement a multiplicative type of STDP derives from the fact that the wall position w is linear with resistance, and thus inversely proportional to synaptic strength. Consequently, we should expect mSTDP also from synchronous STDP realizations, using either current or voltage-driven memristors, as the update would also be weight dependent. On the other hand, we have assumed that function $f()$ in eqs. (5, 6, and 13) does not depend on w . In practice, there might exist such dependence, and if true, the resulting mSTDP learning function might deviate from the one discussed here.

5 CONNECTING MEMRISTORS WITH SPIKING NEURONS FOR ASYNCHRONOUS STDP LEARNING

Synchronous memristive STDP learning architectures were proposed by Snider (2007, 2008), assuming voltage/flux driven memristors, and recently demonstrated by the group at Michigan University (Jo et al., 2010). In that proposal each neural spike is mapped into a sequence of precisely spaced fixed amplitude digital pulses which must maintain global synchronization to separate

the integration phase of neural activity from the synaptic weight update phase. This global synchronization requirement imposes severe difficulties when the system scales up to very large sizes.

On the other hand, from the discussions in previous Sections, we present an alternative approach which is fully asynchronous (Linares-Barranco and Serrano-Gotarredona, 2009). Consequently, no global synchronizations will be required nor global separations into neuron-integration phases and synapse-learning phases, making this approach attractive for scaling up to very large numbers of neurons and synapses.

We first need a neural circuit that integrates spikes until a threshold is reached. At that moment, it should provide a spike of the desired shape. A possible schematic diagram for a leaky integrate-and-fire (LIF) neuron block is shown in **Figure 10**. The neurons need to include a current summing and sinking input terminal so that in the absence of spike output the integral of input current spike signals can be computed, while maintaining the input node tied to a fixed voltage. This can be done by using a lossy integrator with a clamped voltage input. The output of this accumulated integral V_{int} is compared against a reference V_{REF} . If this reference is reached, the comparator output will trigger a spike generation circuit, which provides the output spike of the neuron. During spike generation, the integration capacitor is charged to refractory voltage V_{refr} , while the input opamp is configured as a voltage buffer, thus copying the spike waveform to the neuron input node. An attenuated version of the post-synaptic neural voltage $\alpha_{pos} V_{pos}(t)$ is thus made available to the synaptic memristors connected to this neuron input. Another attenuated version of the spike is fed forward to the output of the neuron $V_{pre}(t) = \alpha_{pre} spk(t)$. During the whole time of the spike (typically in the order of 20–100 ms) the neuron is not integrating (computationally inactive). This time is also called “refractory time.” During the absence of spike output, the spike generation circuit provides a constant voltage V_{rest} .

For the *Spike Circuit* in **Figure 10** an analog circuit can be devised that generates a specific action potential shape with some tunable parameters (Indiveri et al., under review). However, for STDP experiments it is more desirable to allow for full programmability of arbitrarily shaped action potentials. Since all neurons should have the same spike shape (at least all neurons of a part of a whole system), one interesting option is to have a circuit at the chip periphery broadcasting digital samples of the action potentials at different phases to all neurons. The spike generation circuits would then capture the closest phase, delay it properly,

and through a local, compact digital-to-analog converter provide the programmed action potential. Implementation details of such spike generation circuit are beyond the scope of this paper.

For the synapses, it is possible to fabricate very high density memristor crossbar structures (Strukov et al., 2008; Borghetti et al., 2009) which connect to the neural layers, as shown in **Figure 11A**. Neurons generate action potentials with a shape similar to those given in **Figure 8**. Note that the positive terminals of the memristors connect to the pre-synaptic neurons. This way, when $\Delta T > 0$ the memristors see a negative voltage beyond threshold and their resistance (inverse of synaptic strength) will decrease. Alternatively, the same result can be obtained by having the neurons generate an inverted spike action potential (as in **Figure 8B**) but connecting the memristors with opposite polarity, as shown in **Figure 11B**.

In an excitatory synapse a pre-synaptic action potential spike should produce an increment in the neuron integral, i.e., make the integrator output voltage V_{int} approach V_{REF} (see **Figure 10**). During neuronal spike integration a neuron simply accumulates the contributions of incoming spikes on its integrator. All synapses connected to its input node do not experiment any weight update and operate as resistances of constant value. This is guaranteed by making the action potential peaks lower than the threshold value v_{th} in **Figure 5**. In order to have a constant positive resistance contribute a net positive charge packet during each incoming pre-synaptic spike, the net area under the spike waveform (see **Figure 8**) should be positive. For the particular case of parameter selection that results in the action potential shapes in **Figure 8**, it turns out that the spike in **Figure 8A** presents a net negative area while the spike in **Figure 8B** presents a net positive area. Consequently, using spikes with the shape and parameters as in **Figure 8A** results in synapses delivering net negative charges, while using spikes with the shape and parameters as in **Figure 8B** results in synapses delivering net positive charges. If neurons are set such that $V_{REF} > V_{rest}$, the incoming net negative charge packets make the integrator output V_{int} approach V_{REF} . In this case synapses delivering net negative charge packets operate as excitatory synapses. On the contrary, if neurons are set such that $V_{REF} < V_{rest}$ then synapses delivering net positive charge packets operate as excitatory synapses. The arrangement shown in **Figure 11A**, which uses spikes as in **Figure 8A**, therefore results in excitatory synapses delivering net negative charge packets if $V_{REF} > V_{rest}$. On the other hand, for **Figure 11B** which uses spikes as in **Figure 8B**, synapses are excitatory, delivering net positive charge packets if $V_{REF} < V_{rest}$.

Interestingly, the strength of STDP learning can be modulated by changing the amplitudes (or shapes) of the electric spikes in time. This would allow the implementation of faster learning at the beginning of a learning process, and progressively slow learning down as the system stores acquired knowledge, or even turn it off completely after some time. This is a very desired feature for STDP machine learning systems (W. Maass, Private communication).

This way of interconnecting memristors with neurons as in **Figures 10 and 11** avoids cross-coupling of spikes between rows and columns, because all lines are driven by (ideal) voltage sources. Using this arrangement with the memristor macro-model of **Figure 6** we performed intensive behavioral simulations in Cadence-Spectre to test the concept on the 4×4 feed forward array shown in **Figure 12**. Note that the neuron used (shown in the inset) is a particular case of

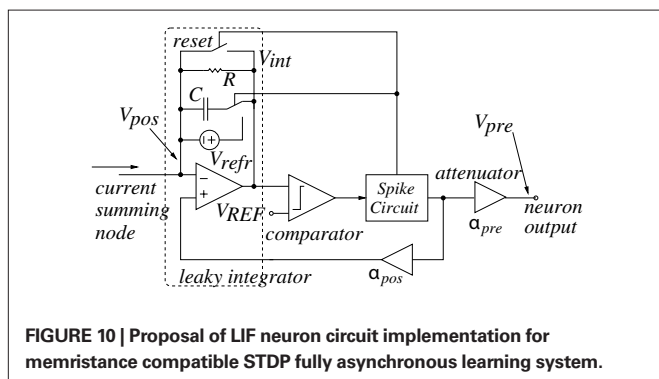


FIGURE 10 | Proposal of LIF neuron circuit implementation for memristance compatible STDP fully asynchronous learning system.

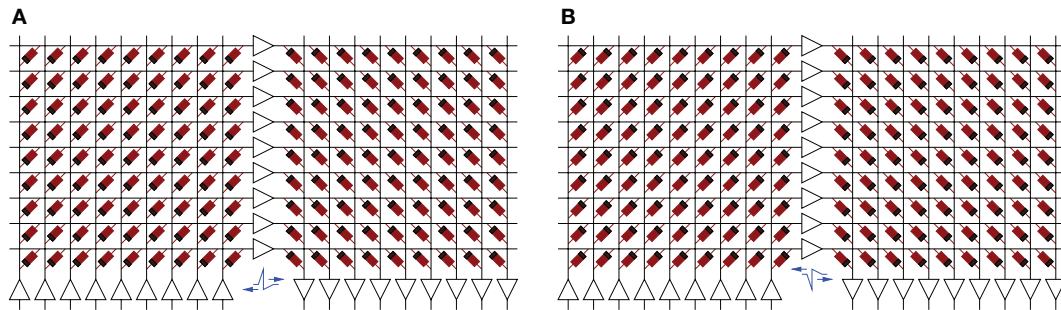


FIGURE 11 | Two possible interconnection schemes between memristors and neurons for STDP learning. (A) Memristor polarities for spikes as in **Figure 8A**. **(B)** Memristor polarities for inverted spikes as in **Figure 8B**.

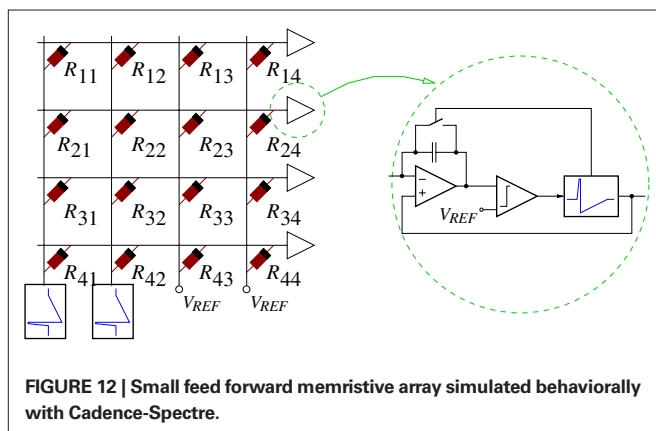


FIGURE 12 | Small feed forward memristive array simulated behaviorally with Cadence-Spectre.

the one in **Figure 10** with $V_{ref} = V_{rest} = 0$ (spike resting potential) and $R = \infty$. The results are shown in **Figure 13**¹². Only the first two column synapses are stimulated, with 200 ms period spikes (of 45 ms duration) with a 25-ms relative delay between the two columns. As can be seen, only the synapses at the first two columns change their resistance, while those on the other two columns do not, confirming the correct operation of STDP without any crosstalk between columns or rows. This demonstrates that this architecture can be scaled up to arbitrary size, at least conceptually. Practical considerations that could limit maximum size are mainly fan-out of neurons, interconnect delays, and parasitic crosstalk. Note that in **Figure 13** memristor resistances do not always converge to their extreme values R_{min} or R_{max} (as in additive STDP) but that some of them (R_{12} , R_{21} , R_{31} , R_{34}) have converged to intermediate values (as is characteristic for mSTDP).

5.1 STDP VARIATIONS

Standard STDP aims to implement the synaptic learning functions of the shape shown in **Figure 3B**. In the case of synapses with negative weights anti-STDP learning functions similar to the one shown in **Figure 3C** need to be implemented. This is achieved with memristors by simply changing their polarity. **Figure 14** shows how neurons and

memristors can be interconnected to achieve anti-STDP learning. Memristors are reversed with respect to the cases in **Figure 11**. Note that now, for anti-STDP, when $\Delta T > 0$ the memristors see a positive v_{MR} voltage beyond threshold (which will produce an increase in resistance and a decrease in synaptic strength), while for $\Delta T < 0$ they see a negative voltage beyond threshold (which will produce a synaptic strength increment). Memristors are physically positive resistances (of time varying values). Whether memristors act as excitatory or inhibitory synapses is determined by the combination of (1) net area under the action potential waveform (i.e., sign of net charge sent to the post-synaptic neuron) and (2) whether V_{REF} is above or below V_{rest} , as mentioned in the discussion around **Figure 11**.

Another twist in STDP variations is obtained by having the neurons send back an inverted version of the spike sent forward, as shown in **Figure 15A**¹³. In this case, it is possible to have the resulting STDP learning function show a positive learning window around $\Delta T = 0$ with positive increments for both positive and negative values of ΔT close to zero. Beyond a specific value of $|\Delta T|$ there are decrements in the weights ($\Delta w < 0$), for both positive and negative sides of ΔT . This is shown in **Figure 15C**. This type of learning is useful under some circumstances (W. Maass, Private communication).

6 ACHIEVING STDP WITH MEMRISTIVE FET-LIKE DEVICES

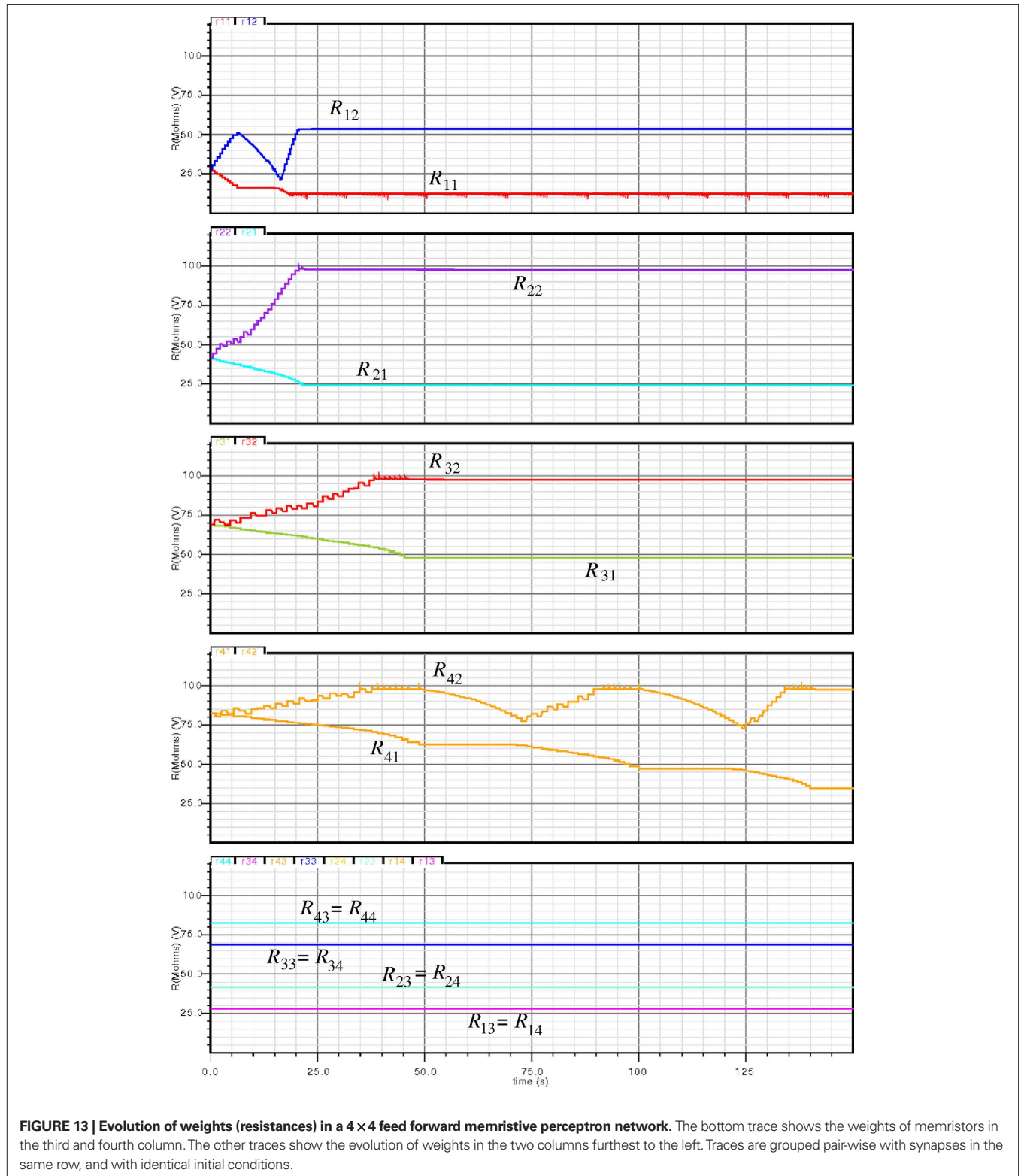
Most present day literature on memristors refers to two-terminal (nano)devices. In essence, memristors operate as resistors the resistance of which can be changed and maintained in a non-volatile manner. This feature (and their compact size) is basically what makes them highly attractive not only for building neuromorphic systems, but also for memories and computing systems in general. However, memristance is not necessarily restricted to two-terminal devices. It is well known that it is possible to use three (or four) terminal FETs as resistors, current sources, or (volatile) memory elements. If the same nanoscale principles that give rise to memristance in two-terminal devices could be extrapolated to three or four terminal FETs, then the adaptive memristive circuits presented so far could be extrapolated to more generic FET-based circuits as well. FETs have more terminals and consequently will

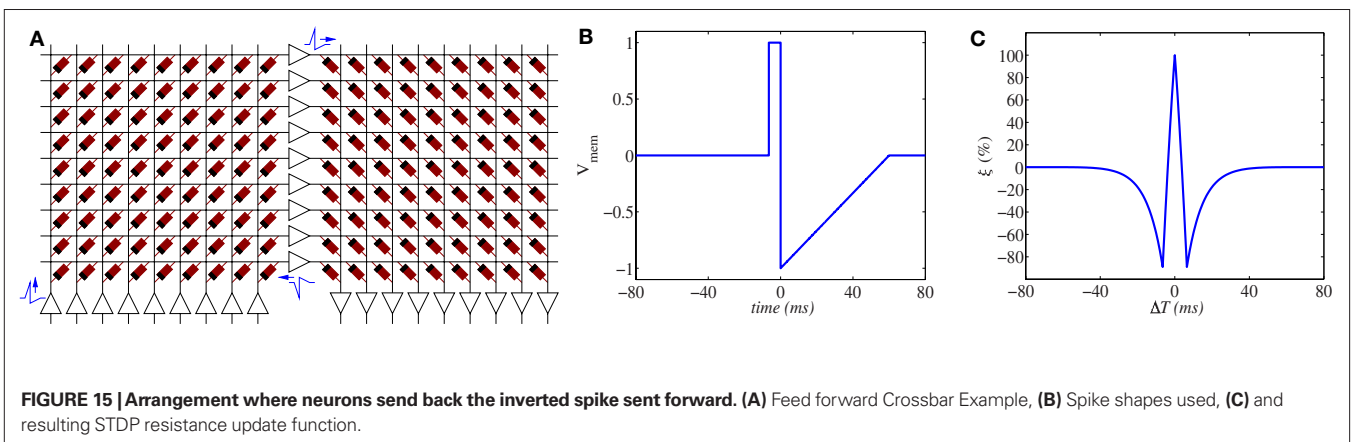
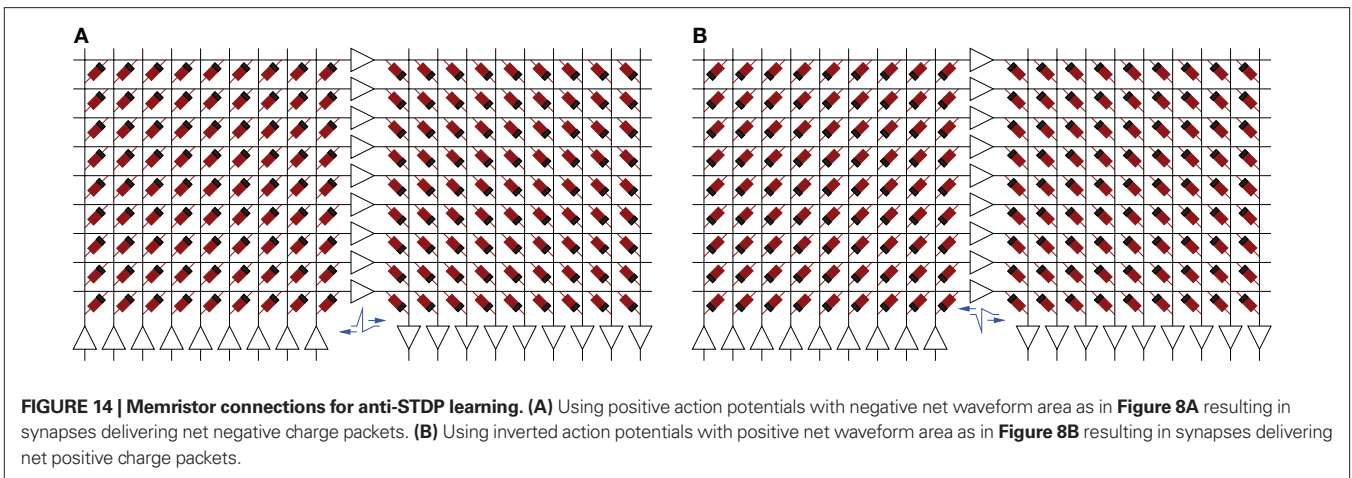
¹²Spectre netlist files for these simulations are available in folder *Spectre_for_Fig13* in the *AdditionalMaterial* file, downloadable from the Journal web site.

¹³Matlab files for these simulations are available in folder *Matlab_for_Fig15* in the *AdditionalMaterial* file, downloadable from the Journal web site.

result in less dense structures than their two-terminal counterparts. However, FETs can present very wide tuning ranges. For example, imagine a (nano)FET transistor in which the threshold voltage could be tuned through some memristive-like mechanism.

Figure 16A shows a 3-terminal FET symbol, and **Figure 16B** illustrates how such a FET can be used as a very-wide-range tunable current source by maintaining its V_{GS} voltage constant but changing its threshold voltage.





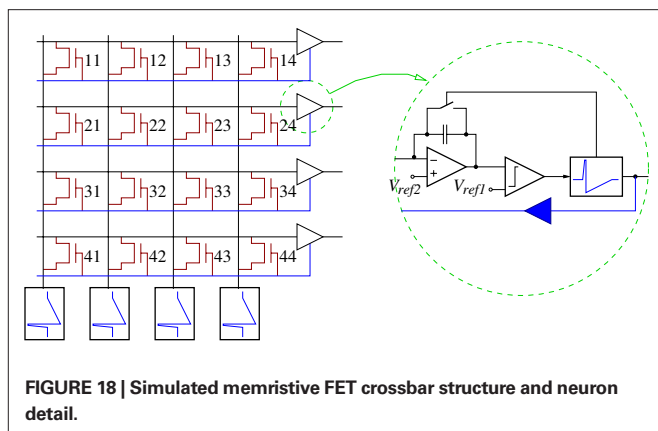
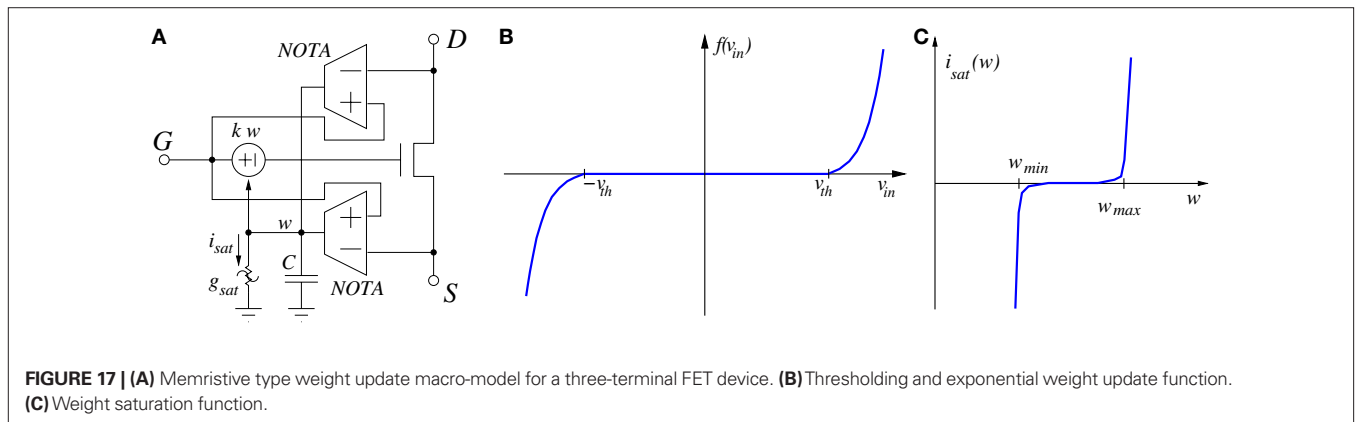
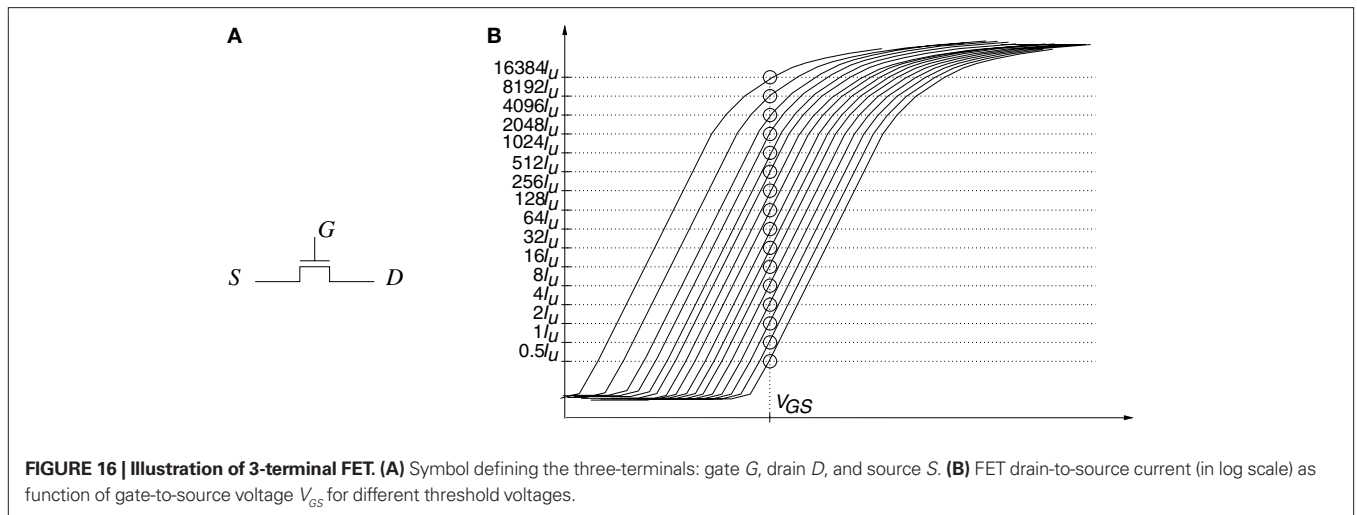
Some researchers are developing nanoscale FET devices in which the threshold voltage can be adjusted by manipulating their terminal voltages above certain thresholds¹⁴ (Lai et al., 2008; Agnus et al., 2010; Bichler et al., 2010; Jeong et al., 2010). Such behavior suggests a memristive (nanoscale) weight update mechanism similar to the one we have assumed for two-terminal memristors in the previous Sections. Let us assume we can change the FET threshold voltage up and down whenever either the gate-to-drain V_{GD} and/or gate-to-source V_{GS} voltages exceed certain positive and negative thresholds. A FET macro-model for describing such behavior, inspired by the one in **Figure 6** for memristors, is shown in **Figure 17A**. There, a fixed-threshold (memory-less) FET is used, in which the internal gate terminal is connected through a variable voltage source to the external device gate G . This voltage source is controlled by an internal weight w , which changes by means of a mechanism similar to that of the memristor weight in **Figure 6**. The NOTA non-linear function, shown in **Figure 17B** is identical to that in **Figure 5B**, and the weight saturation function shown in **Figure 17C** is identical to that in **Figure 5C**.

Using such memristive FETs we can assemble a crossbar-like feed forward perceptron network, as shown in **Figure 18**. The synapses are now implemented using FETs of this type. Input spikes coming

from previous layer neurons activate the FET drain terminals. The FET source terminals are connected to fixed voltages V_{ref2} clamped by the neurons' input. The neurons' inputs are now always clamped to a fixed voltage, i.e., the neuron output spike is not sent back through the neuron input terminal. Now, the neuron spike is sent back through an independent extra terminal which connects to the gates of the FETs in the same row. If a pre-synaptic spike (on a synapse FET drain terminal) is prior to a post-synaptic spike (on the same synapse FET gate terminal) then the FET's gate-to-drain voltage would exceed positive threshold v_{th} in **Figure 17B**, activating the corresponding NOTA, increasing w . This would decrease the FET's threshold voltage and shift its current versus V_{GS} curve in **Figure 16B** to the left, thus increasing its driving current for the same V_{GS} voltage. Consequently, this results precisely in an STDP positive update. If the pre-synaptic spike occurs later than the post-synaptic spike, the opposite behavior is obtained: a negative STDP update. The circuit in **Figure 18** was simulated in Cadence-Spectre using the macro-model in **Figure 17** for the FETs. The simulation results are shown in **Figure 19**¹⁵. For the neuron we used a simplified version of the circuit in **Figure 10** which is not lossy and which has a refractory voltage equal to the resting potential. For the simulations in **Figure 19** we set the range

¹⁴Some other researchers have demonstrated STDP with 4-terminal FETs that include a floating gate (Ramakrishnan et al., 2010).

¹⁵Spectre netlist files for these simulations are available in folder *Spectre_for_Fig19* in the *AdditionalMaterial* file, downloadable from the Journal web site.



of w as $w_{max} = 10$ V and $w_{min} = -10$ V. We can see again that some weights converge to the boundary values, while others stabilize at intermediate values, as is characteristic for mSTDP.

7 ADDRESS-EVENT-REPRESENTATION

Address-event-representation (AER) is a well established technology among neuromorphic engineers. AER was originally proposed 20 years ago in Mead’s Caltech research lab (Sivilotti, 1991;

Mahowald, 1992). For over 10 years AER sensory systems were reported by only a handful of research groups, examples being Lazzaro et al. (1993) and Johns Hopkins University (Cauwenberghs et al., 1998) pioneering work on audition, or Boahen (1999, 2002) early developments on retinas. However, during these years some basic progress was made. A better understanding of asynchronous design (Sparsø and Furber, 2001; Martin and Nyström, 2006) leading to robust unarbitrated (Mortara et al., 1995) and arbitrated (Boahen, 1996, 2000) asynchronous event readout, combined with the availability of user-friendly FPGA external support for interfacing and new submicron technologies allowing complex pixels in reduced areas, heralded a new trend in AER bio-inspired Spiking Sensor developments. Since 2003 many researchers have embraced this trend. AER has been used fundamentally in vision (retina) sensors, for purposes such as simple light intensity to frequency transformations (Culurciello et al., 2003; Posch et al., 2010), time-to-first-spike coding (Ruedi et al., 2003; Chen and Bermak, 2007), foveated sensors (Azadmehr et al., 2005), spatial contrast (Costas-Santos et al., 2007; Massari et al., 2008; Ruedi et al., 2009; Leñero-Bardallo et al., 2010), temporal contrast (Mallik et al., 2005; Posch et al., 2007, 2010; Lichtsteiner et al., 2008; Leñero-Bardallo et al., 2011), motion sensing and computation (Boahen, 1999), and combined spatial and temporal contrast sensing (Zaghloul and

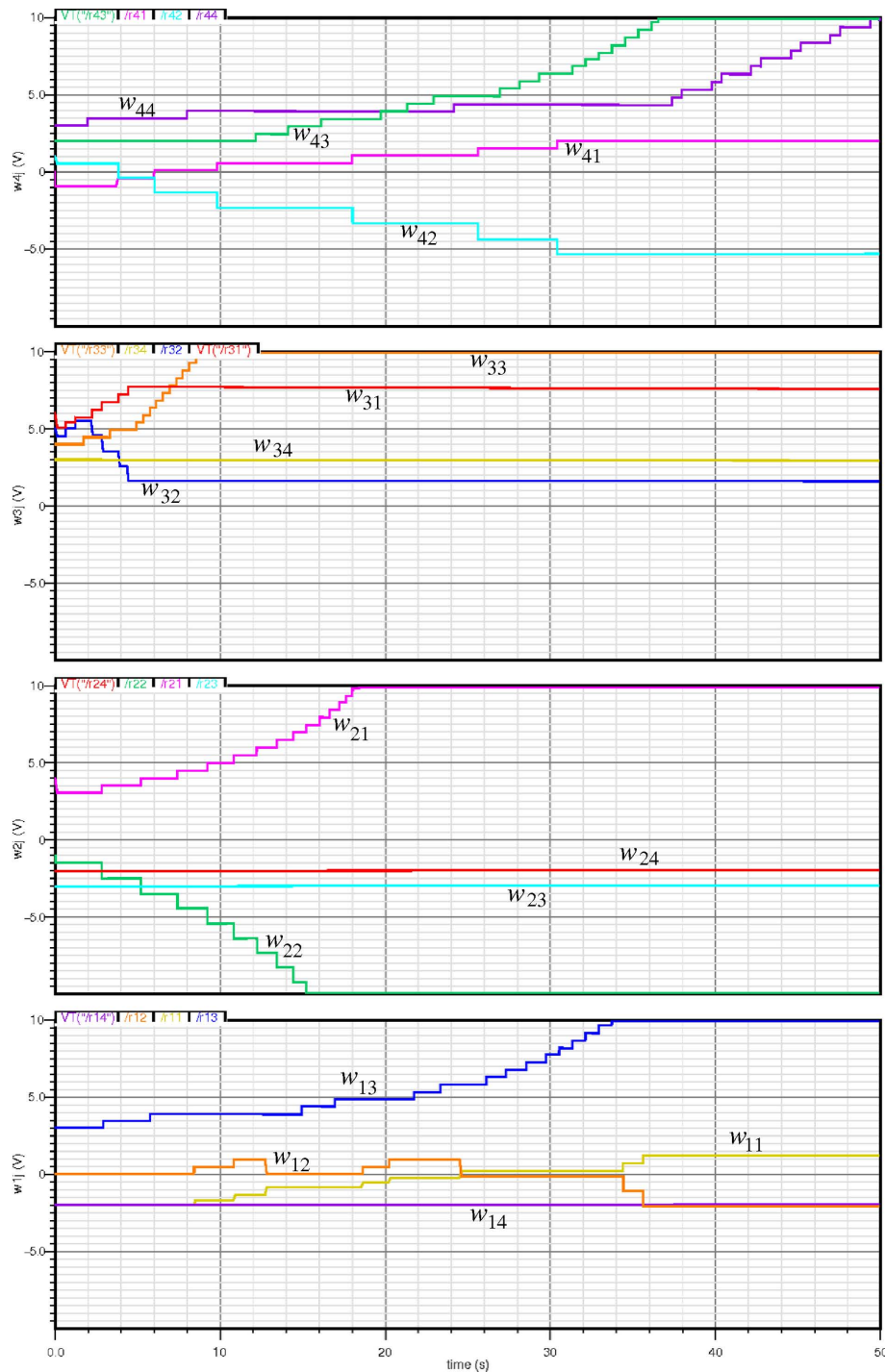


FIGURE 19 | Simulated evolution of weights (w in Figure 17A) for the FET network in Figure 18.

Boahen, 2004a,b). AER has also been used for auditory systems (Lazzaro et al., 1993; Cauwenberghs et al., 1998; Sarpeshkar et al., 2005; Wen and Boahen, 2006; Chan et al., 2007), competition, and winner-takes-all networks (Chicca et al., 2007; Oster et al., 2008), and even for systems distributed over wireless networks (Teixeira et al., 2005).

But AER has also been employed for post-sensing event-driven processing, emulating biological cortical structures. Vernier et al. (1997) developed AER convolutional filters with elliptic-like kernels while Choi et al. (2005) reported more sophisticated Gabor-like filters. Serrano-Gotarredona et al. (1999) reported an AER architecture that would allow more generic kernels, although with some

geometric symmetry restrictions. In 2006 the same group started to report working AER Convolution chips with arbitrary shape programmable kernels of size up to 32×32 (Serrano-Gotarredona et al., 2006, 2008; Camuñas-Mesa et al., in press).

Figure 20 explains the basic idea behind a point-to-point AER link. An emitter chip (or module) includes an array of neurons generating spikes. Each neuron is assigned an address, such as its (x, y) coordinate within the array. Neurons generate spikes at low frequency (10–1000 Hz), and these are arbitrated and put on an inter-chip (or inter-module) high-speed asynchronous AER bus. The AER bus is a multi-bit (either parallel, serial, or mixed) bus which transmits the addresses of the emitting neurons. Typical delays for transmitting Address Events between AER chips range from about 30 ns (Costas-Santos et al., 2007) to 1 μ s (Lichtsteiner et al., 2008) per event for parallel AER, and have been reported down to 24 ns per event for serial AER with potential to go as low as 8 ns per event (Berge and Hafliker, 2005). These addresses are received, read, and decoded by the receiver chip (or module) and sent to the corresponding destination neuron or neurons. **Figure 20** illustrates a point-to-point AER link with a single emitter and a single receiver. The use of AER splitters and mergers (Serrano-Gotarredona et al., 2009) allows extension to one-to-many, many-to-one, or many-to-many AER links. Inserting AER mappers (Serrano-Gotarredona et al., 2009) allows coordinate transformations (rotations, translations, etc.) to be performed while address events travel between modules. Current research is looking at how large numbers of AER convolutional modules can be combined through independent and multiple AER links to build high-speed object and texture recognition systems (Camuñas-Mesa et al., 2010; Pérez-Carrasco et al., 2010b).

8 BUILDING A SELF-LEARNING VISUAL CORTIX WITH MEMRISTORS AND STDP-READY AER HARDWARE

In previous Sections we have shown how to interconnect memristors with spiking neurons to achieve STDP learning systems. We have illustrated this with a very specific topology, a feed forward crossbar structure (**Figures 11, 14, and 15**), where all neurons in one layer connect to all neurons in the next layer. However, the methodology is not restricted to this specific spatial topology, and can be extended to any generic neural network topology. In this Section we will apply those same concepts to a topology representing the first processing layer of the visual cortex, namely layer V1. We will first explain the V1 layer topology we will use, show how to build it physically, then we will describe the training data we will use from

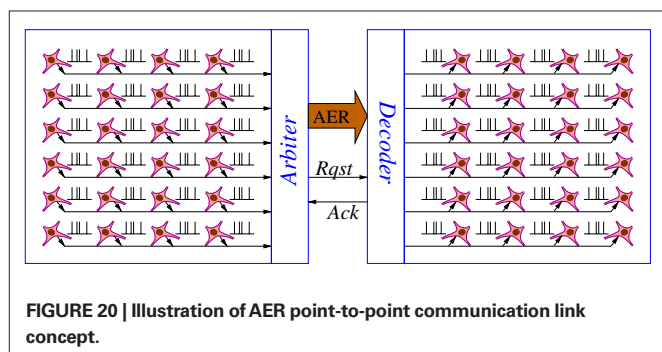


FIGURE 20 | Illustration of AER point-to-point communication link concept.

a real artificial AER retina, and finally we will show the receptive fields formed through STDP learning in the artificial memristive V1 layer with this training data. The biological V1 visual cortex layer is known to be sensitive to specific orientations (Hubel and Wiesel, 1959). We will show how such orientation sensitive receptive fields arise naturally when building an artificial memristive V1 layer with STDP learning and stimulated with real spiking data obtained with an artificial AER motion sensitive retina.

The spontaneous formation of orientation sensitive receptive fields through STDP learning has already been reported by other researchers¹⁶ (Delorme et al., 2001; Snider, 2007). In those works static luminance images were used for training. Pixel intensities were coded into spikes through some kind of computational transformation: either a stochastic rate coding scheme (Snider, 2007), or a rank-order coding scheme (Delorme et al., 2001). Here we directly use the continuous AER output stream of events produced by a real motion sensitive retina CMOS sensor.

8.1 TOPOLOGY OF V1 VISUAL CORTIX LAYER AND PHYSICAL REALIZATION

The simplified V1 topology we want to emulate can be explained with the help of **Figure 21A**. The retina sends spikes to the V1 cortex layer through synaptic connections¹⁷. The V1 layer is structured in a number of “Feature Maps.” We can think of the retina as an array of “pixels,” each with coordinate (x, y) . Let us assume each “Feature Map” in V1 replicates the same coordinates (x, y) , so that each pixel in the retina has a corresponding pixel in each “Feature Map” with the same coordinate. Each pixel (x_c, y_c) in a “Feature Map” receives inputs not only from pixel (x_c, y_c) in the retina, but also from all neighbors within a spatial neighborhood $(x_c + x_r, y_c + y_r)$. Alternatively, we may say that each pixel in the retina (x_c, y_c) connects to a *Projection Field* of pixels $(x_c + x_r, y_c + y_r)$ in each of the *Feature Maps*. Thus, *projection fields* include a number of synaptic connections, so that the spikes produced by one pixel in the retina are sent to the pixels of the *projection field* in each *feature*

¹⁶It should be clarified, however, that the present body of experimental neuroscience literature confirms that preliminary orientation tuned receptive fields build up during development before eye opening. Afterward, visual experience contributes to the maturation and sharpening of orientation selectivity (White et al., 2001).

¹⁷Here we are ignoring the lateral geniculate nucleus (LGN) between the retina and V1, as in many visual system models. LGN seems to introduce a relay only, without performing any computation.

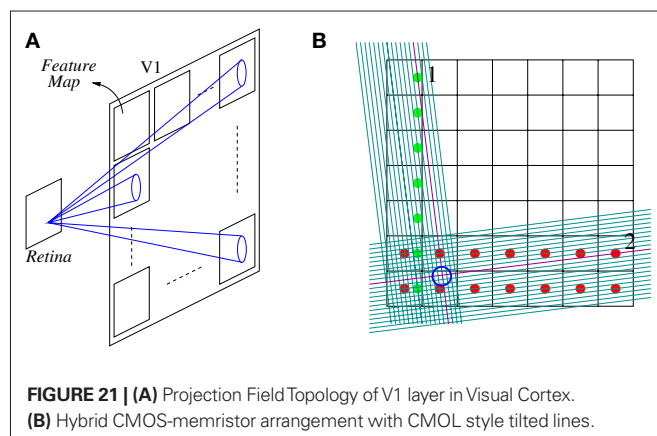


FIGURE 21 | (A) Projection Field Topology of V1 layer in Visual Cortex. (B) Hybrid CMOS-memristor arrangement with CMOL style tilted lines.

map. Feature maps operate as feature extractors. Specifically, the feature maps in V1 detect the presence of oriented edges at specific orientations and scales (Hubel and Wiesel, 1959).

The physical implementation of one such *Feature Map* with AER CMOS neurons and a layer of memristor crossbar structure on top is shown in **Figure 21B** (Snider, 2008). The lower CMOS contains the array of neurons (or pixels) of one V1 *Feature Map*. Each neuron has coordinate (x, y) , as its corresponding retina pixel. Address Event spikes of coordinate (x, y) coming from the retina are sent to pixel (x, y) in the *Feature Map*. This neuron then sends out a spike of the desired shape (for example, as in **Figure 8**) through its output node. In **Figure 21B** each neuron has an output node (green) and an input node (red). The output node connects to a thin line tilted slightly with respect to the CMOS tile (as in CMOL; Strukov and Likharev, 2005), so that it does not intersect with any other neuron output node in the CMOS tile. This thin line has many others in parallel, each connecting to one neuron output node. Perpendicular to all these lines there are other thin lines (at a different altitude), each connecting to the input node of one neuron. The two sets of perpendicular lines form a “sandwich” with a separation layer formed by memristive material. This way, at the intersection of each perpendicular thin line there is a memristor. For example, in **Figure 21B** neuron 1 output node connects to the vertical pink line, while neuron 2 input node connects to the horizontal pink line. The synaptic memristor connecting neuron 1 output to neuron 2 input is at the intersection of the two lines (blue circle). The vertical pink line (neuron 1 output) has memristive intersections with all horizontal thin lines. Consequently, neuron 1 output connects to all other neuron inputs. In the same manner, all neuron output nodes connect to all neuron input nodes. For *projection field* based topologies, each neuron output does not connect to all other neuron inputs. Instead, connectivity is limited to a given spatial neighborhood. This is achieved by having lines of limited length (instead of one reaching over the full CMOS array). For square projection fields of size $10k = 100^2$, for example, each line has to be extended to 50 cells on each side.

Below we present some simulation results from training a set of such AER *Feature Maps* with real stimuli coming from a temporal derivative AER retina watching life scenes. First, we briefly explain

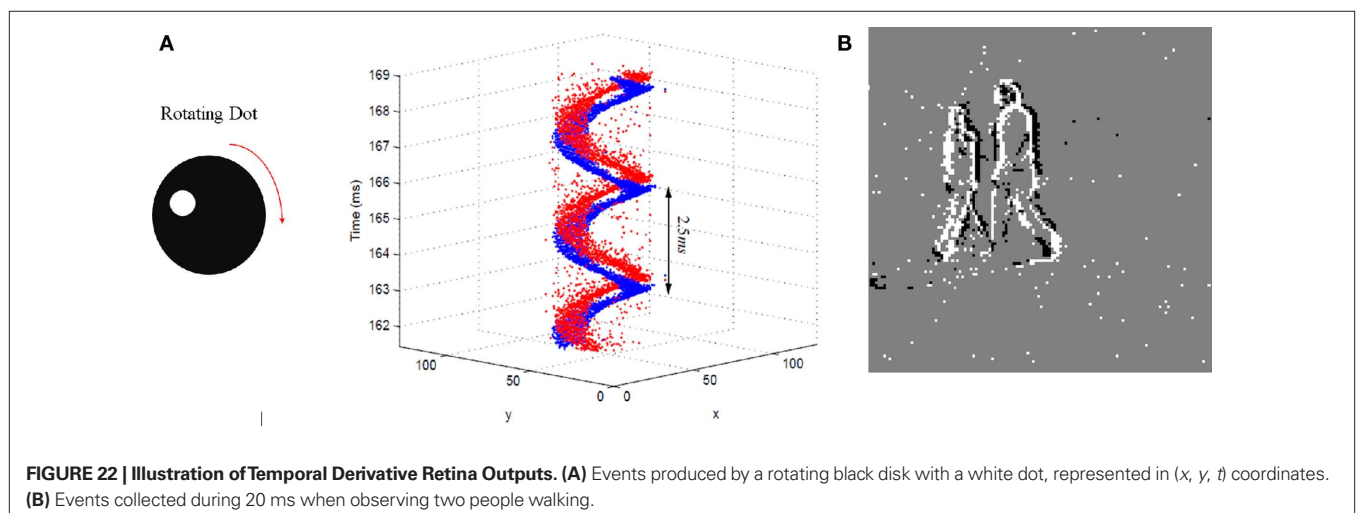
the AER temporal derivative retina and what kind of spikes it produces. We then describe how we used this data to stimulate a set of AER hybrid CMOS-memristor *Feature Maps* and what kind of selectivity these *Feature Maps* developed.

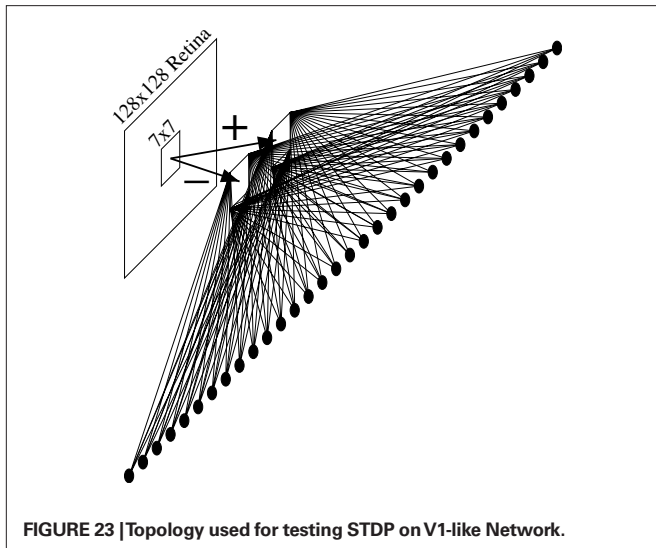
8.2 AER TEMPORAL DIFFERENCE RETINA

We will use the spiking data obtained from an AER Temporal Difference Retina chip (Posch et al., 2007, 2010; Lichtsteiner et al., 2008; Leñero-Bardallo et al., 2011) to train an artificial V1 STDP visual cortex layer. The retina has an array of 128×128 pixels. Each pixel (x, y) has a photo sensor that provides a continuous photo current $I_{ph}(x, y)$ plus a circuitry that generates a signed spike every time its photo current changes by a given relative amount $|\Delta I_{ph}/I_{ph}| > C_{th}$. **Figure 22A** shows the output events produced by the retina when observing a dot rotating at 400 Hz. Blue dots represent positive events (going from dark to bright) and red dots represent negative events (going from bright to dark). The address events collected during 7 ms are plotted in (x, y, t) coordinates. **Figure 22B** shows the events collected during 20 ms when observing two people walking. White pixels correspond to positive events ($\Delta I_{ph}/I_{ph} > 0$), while black pixels to negative events ($\Delta I_{ph}/I_{ph} < 0$).

8.3 STDP TRAINING RESULTS OF V1 LAYER

In this Section we will analyze the learning behavior a hybrid CMOS-memristive V1-like system when it is trained through STDP using the architectural and circuitual principles outlined throughout the paper and using real stimuli obtained from a 128×128 pixel AER temporal derivative retina. Specifically, we used a 521-s recording with 20.5 million events showing scenes observed when driving in a car (Delbruck). We used a simplified network structure to simulate and see what kind of receptive fields would naturally arise. The network structure is shown in **Figure 23**. From the retina visual field of 128×128 pixels we cropped 324 non-overlapping patches of 7×7 pixels each, and concatenated all these events sequentially making a recording of $324 \times 521 = 168804$ s (47 h) with 19.6 million events. This concatenation was used for one training epoch, and we required a total of five epochs to observe convergence in the learned weights. The events from each patch are separated into two additional 7×7 fields depending on the event sign. The activity of





these two subfields is projected onto 32 neurons¹⁸. Consequently, there are $32 \times 2 \times 7 \times 7$ trainable weights. Weights are always positive. Each of the 32 neurons inhibits the other neurons through lateral inhibitory connections, as in reference (Masquelier et al., 2009a). Each neuron is as shown in **Figure 10**, with a leak and a refractory voltage. Inhibitory lateral connections have fixed weights, while the weights of the feed forward connections follow STDP learning. Weights were initialized either to random values, or to maximum values. The STDP learning functions were such that the ratio of the negative side area over the positive side area was [see eq. (1)] $a^- \tau^- / a^+ \tau^+ = 1.25$, meaning that STDP was biased toward depression. The time constant for the positive side was $\tau^+ = 13.6$ ms, while that of the negative side was $\tau^- = 15.2$ ms, and there is a central linear region for $|\Delta\tau| < 0.5$ ms. Memristor resistances were bounded to the interval $R_{min} = 10 \text{ M}\Omega$, $R_{max} = 100 \text{ M}\Omega$. We simulated this system theoretically in several ways: (1) by solving the differential equations of biological integrate-and-fire neurons via an Euler method with a time step update of 0.1 ms, using the Brian simulator (Goodman and Brette, 2008) and a conventional additive STDP learning rule; (2) by using a dedicated event based simulator adapted from reference (Masquelier et al., 2009a) implementing the quadratic STDP learning rule of the memristors and the neuron dynamics corresponding to the circuit in **Figure 10** with spikes as in **Figure 8**¹⁹; and (3) by simulating a simplified event-driven matlab code with instantaneous neuron dynamics (but including a non-instantaneous leak) and with quadratic mSTDP²⁰. In all cases, receptive fields became clearly orientation selective. The resulting receptive fields are biased to vertical edges, similar to the visual input stimuli we have used for this experiment.

¹⁸Compared to the arrangement in **Figure 21A**, each of the 32 neurons in **Figure 23** represents one of the *Feature Maps* in **Figure 21A**. Consequently, to implement the full V1 structure physically, each neuron with all its 7×7 input synapses in **Figure 23** has to be “cloned” in a 128×128 array.

¹⁹The corresponding Matlab and C code files for these simulations are available in folder *STDP_V1_C_code_Fig24* in the *AdditionalMaterial* file, downloadable from the Journal web site.

²⁰The Matlab file is available in folder *STDP_V1_matlab* in the *AdditionalMaterial* file, downloadable from the Journal web site.

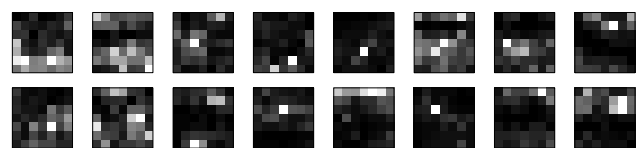
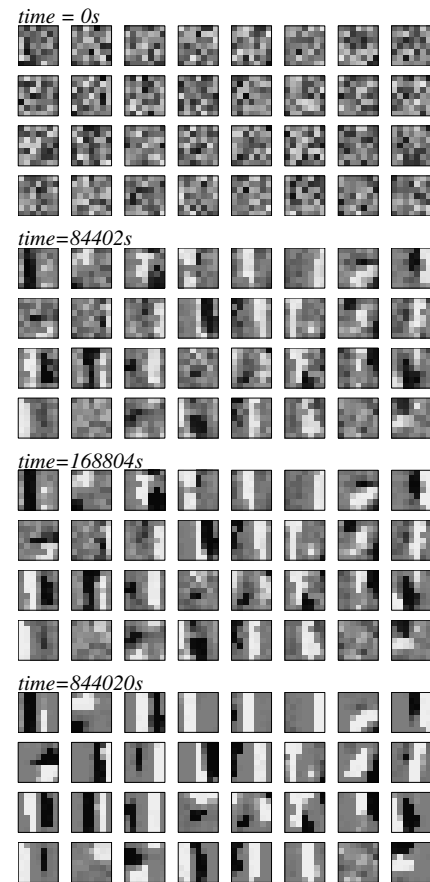


Figure 24 shows the evolution of the receptive fields when using the dedicated event based simulator [case (2) in previous paragraph]. We see the receptive fields of the 32 neurons, where the positive and negative weights are grouped together in the same 7×7 square by assigning “white” to positive weights and “black” to negative weights. The central gray color means zero weight. As can be seen, there is a clear tendency for the receptive fields to become orientation selective.

It is worth mentioning that the type of continuous processing involved here differs from time-to-first-spike (or rank-order) coding schemes, where a stimulus on-set provides a reference time

(Delorme et al., 2001; Guyonneau et al., 2004; Masquelier and Thorpe, 2007, 2010; Weidenbacher and Neumann, 2008). It also differs from Phase-of-Firing coding, where the peak of a population activity oscillation is used as a reference time (Masquelier et al., 2009b). Here there is no oscillation, nor stimulus on-set, nor any reference time, but a continuous flow of spikes, and yet STDP is able to pick patterns that are consistently present in the training data, confirming previous results (Masquelier et al., 2008). Future work, however, will evaluate the use of periodic resets in the AER retina, leading to time-to-first-spike coding with respect to those resets.

We also simulated the network in Spectre using the memristor macro-model in Section 3.1, but for 16 neurons only²¹. However, electrical circuit simulation was very slow. Simulating for just one of the 324 input patches (with about 154 K events) took 559 ks CPU time (6.5 days) running on a SUN Fire X2200 M2 Linux cluster with dual cores at 2.2 GHz and 4 GB RAM. In this case we could only verify that the initial evolution of weights was similar to those of the software programs, as shown in **Figure 25**.

For the circuit simulations we used the topology and spike shapes shown in²² **Figure 26**. There are two input memristor arrays, one for the positive and one for the negative subfields in **Figure 23**. The backward spikes are attenuated by $\alpha_{pos} = 0.97$. The output neurons forward spikes are rectified and sent back through non-trainable fixed value resistors $R_{inh} = 2 \text{ M}\Omega$ to implement the lateral inhibitory connections. The parameters used for the memristors are $w_{max} = 10 \text{ V}$, $w_{min} = -10 \text{ V}$, $C_{MR} = 50 \text{ mF}$, $R_{min} = 8 \text{ M}\Omega$, $R_{max} = 100 \text{ M}\Omega$, $k_R^{-1} = 217 \text{ nA}$, $w_o = 11.74 \text{ V}$, $I_o = 10 \text{ }\mu\text{A}$, $v_o = 0.1 \text{ V}$, $v_{th} = 1 \text{ V}$, and for the neuron $R = 1 \text{ M}\Omega$, $C = 19.2 \text{ nF}$, $V_{refr} = 0.625 \text{ V}$, $V_{REF} = 1 \text{ V}$. With these memristor parameters and the spike shapes in **Figure 26** we were able to characterize the STDP learning function of the memristors and adjust them to the learning function in the event based simulation [case (2) above]. **Figure 27** shows the

²¹Spectre and Matlab files for these simulations are available in folder *Spectre_STDP_V1* in the *AdditionalMaterial* file, downloadable from the Journal web site.

²²This topology and these spike shapes also correspond to the theoretical simulations of case (2) and **Figure 24**. The only difference between the two cases is in the number of neurons used: 32 neurons for the theoretical simulations, and 16 neurons for the circuit simulations.

STDP learning function characterized through electrical spectre simulations (blue dots) to match the ideal function used in the theoretical simulations (red circles)²³.

At this point we would like to highlight an important difference between the memristor-based network of integrate-and-fire neurons with STDP synapses presented here, and an equivalent network as used normally among neurocomputational researchers (see the integrate-and-fire neuron model in Gerstner, 1999). In this latter case, the evolution of membrane voltage following an input spike at t_{spk} is as if the spike injects a current $I_{spk}(t > t_{spk}) = I_m e^{-(t-t_{spk})/\tau_{spk}}$ at node V_{pos} in **Figure 10**. Parameters I_m and τ_{spk} defining this spike contribution are independent of the parameters a^+ , a^- , τ^+ , τ^- in eq. (1) used to characterize the STDP learning function. However, in case of the memristor implementation, each spike injects a current

²³Spectre and Matlab files for these simulations are available in folder *Tune_Memristor_in_Spectre* in the *AdditionalMaterial* file, downloadable from the Journal web site.

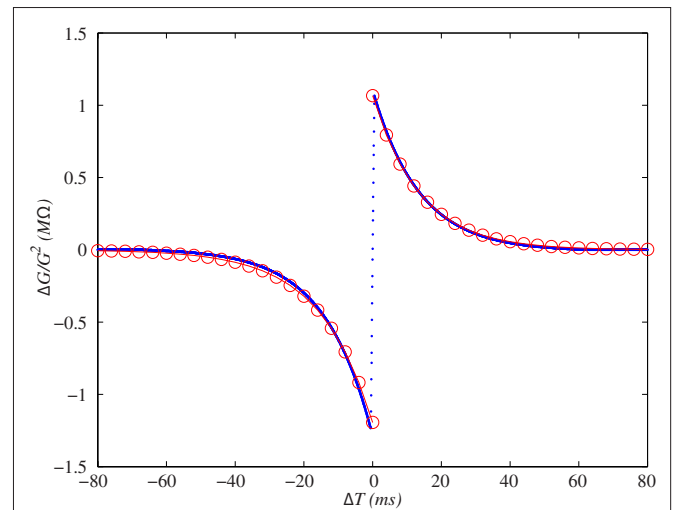


FIGURE 27 | Spectre characterization of STDP learning function of memristors. Blue dots are results obtained from Spectre electrical circuit simulations. Red circles correspond to the theoretical target function used in the theoretical behavioral simulations.

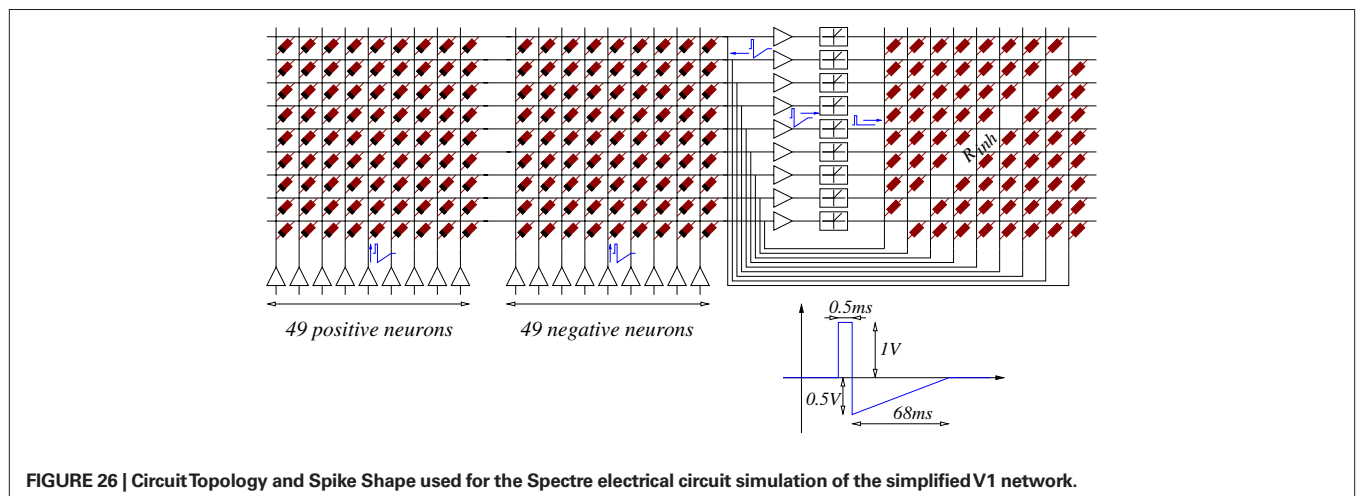


FIGURE 26 | Circuit Topology and Spike Shape used for the Spectre electrical circuit simulation of the simplified V1 network.

at node V_{pos} in **Figure 10** proportional to the spike waveform delivered by the neurons. Since this waveform also determines the shape of the STDP learning function, it turns out that there is now a strong dependency between the parameters defining the evolution of the membrane voltage and those defining the STDP learning function. They are no longer independent and it is consequently more difficult to adjust all true independent parameters properly for a desired behavior.

9 PRACTICAL LIMITATIONS, REALISTIC SIZES, PITCHES, DENSITY, CROSSTALK AND POWER CONSIDERATIONS

Nanoscale memristor technology is still quite incipient and no realistic large scale systems have been reported at the time of writing (as far as we know). However, we can estimate an orientative scale and density of what may realistically be achieved in the near future, and the main limitations which may be encountered in a real physical implementation.

Regarding the wiring density of synaptic memristors, a pitch of 100 nm is conservatively realistic for present day technologies (Jung et al., 2006; Green et al., 2007), while the near future might bring us closer 10 nm (Jeon et al., 2010). Assuming wafer scale dies of 100 nm pitch 2D memristor arrays capable of interfacing reliably with lower CMOS become available some time soon, this would result in a synaptic density of 10^{10} synapses/cm².

In the brain, the number of synapses per neuron is about 10^3 – 10^4 . If we want to maintain the 10^4 ratio, we would need to fabricate CMOS neurons with a pitch of 10 μ m, resulting in 10^6 neurons/cm². Such neuron sizes are quite realistic for present day nanometer scale CMOS (45 or 32 nm), given the complexity of the neurons needed.

Another problem is that of resistance value ranges of the memristors' R_{min} (synapse ON) and R_{max} (synapse OFF). Reported memristors present resistance values from the $k\Omega$ range up to the $M\Omega$ range (Borghetti et al., 2009; Jo et al., 2009, 2010). The memristor resistance value range affects the performance, reliability, crosstalk and power dissipation of a full large scale system. For example, it affects the driving capability of the neurons and their power consumption. If one neuron needs to drive 10^4 synapses of average value 1 $M\Omega$ to an average 1 V level, it has to be able to provide an average current of 10 mA during a spike (of say 20 ms), delivering 10 mW per spike. If there are 10^6 neurons/cm² each firing at an average of 10 Hz (which is similar to biological neurons), the synapses would dissipate a power of about 2 kW. The neurons would need at least the same power, presumably more. It is obvious that such a structure would melt quickly. The resistance range needs to be increased by a minimum factor of 100, so that minimum resistances are at least 100 $M\Omega$, or even larger. As pitches are lowered, resistances would need to increase quadratically with pitch decrease, to maintain the power limitation. Another option would be to scale down voltage, but there is not much range. Even our 1 V maximum voltage assumption is quite optimistic for available present day memristors, which tend to operate between 2 and 10 V (Borghetti et al., 2009; Jo et al., 2009, 2010). Also, we have always assumed so far that voltage sources driving memristor terminals behave as ideal voltage sources, or at least, that the output resistance of such voltage sources is negligible compared to the total resistance they have to drive. Again, this will be achieved more easily if memristors present rather high resistance values. If driving voltage sources are

no longer so ideal, then there will be crosstalk between lines. For example, if a spike is sent to a column then the voltage on all rows would change slightly. The consequence of this is that part of the charge provided by the incoming spike will be lost through non-desired synapses and the impact of the spike on the target neurons will be weaker. During learning, the situation is less severe because for STDP update the memristor voltage has to exceed the learning threshold [v_{th} in eq. (5)]. The effect of having non-ideal voltage sources is that the terminal voltage difference on the memristors needing synaptic update would be slightly less than in the ideal situation and learning would be weaker than expected ideally. However, having non-ideal voltage sources would not induce STDP update in undesired synapses. Another parasitic issue related to crosstalk is parasitic capacitive crosstalk between lines, which can be more pronounced as pitch and line distances decrease.

Also, one highly critical aspect which needs to be evaluated is the influence of component mismatches. Nanoscale devices suffer from high mismatch in general. Consequently, we should expect nanoscale memristors too to suffer from great parameter variations from one to another. It is true that they will operate as adaptive devices that will learn their functionality hopefully compensating for (some) mismatches. However, their learning and adaptation rules will also suffer from mismatch, making some synapses learn faster than others, or in slightly different fashions. In any case, the main sources of mismatch in memristor devices still need to be identified, and then their influence in the overall system learning behavior evaluated. However, to undertake such an initiative, we first need ready access to large arrays of reliable memristors fabricated in a stable and repeatable manner.

In general, an important issue is precise memristor modeling. Throughout this paper we have assumed an idealized voltage-driven memristor macro-model. This is useful to devise possible system architectures to achieve a desired functionality, such as STDP learning. However, to estimate realistic performance figures of resulting systems, it will be necessary to include non-ideal effects, both of the memristors and companion CMOS circuits. In this paper, no high order effects have been modeled, such as those related to noise, mismatch, and other memristor non-idealities not yet reported. For example, we have assumed that the wall model boundary condition is imposed by the voltage dependent function in eqs. (5 and 6). If this function turns out to be dependent on w , has other non-ideal effects, or there are other important mechanisms involved in the boundary conditions, this might affect the resulting behavior of the STDP learning function when integrating eq. (13), questioning the quadratic mSTDP behavior we have assumed.

10 CONCLUSION

In this paper we have shown that STDP learning can be induced by the voltage/flux driven formulation of a memristor device. We have used this formulation to develop fully asynchronous circuit architectures capable of performing STDP, by having neurons send their spikes not only forward but also backward. We have seen that the STDP learning rule induced this way is of a multiplicative, specifically quadratic type. We have shown how the shape of spikes is critical to achieve and modulate a specific STDP learning function. We have provided a memristor macro-model for simulating arrays of memristors efficiently in circuit simulators. Finally,

we have studied an emulation of the V1 visual cortex capable of self-learning how to extract orientations from spiking inputs provided by a real physical AER spiking retina fabricated in CMOS. At the end we have also discussed possible limitations of present day memristors.

The presented results are ideal extrapolations based on behavioral simulations. As memristor devices are further developed and non-ideal effects become known, the impact of non-idealities in the presented architectures and methods can be further assessed. Future work has to evolve toward more realistic memristor models and improved memristor devices, specially devices with much higher resistivities. One critical property that memristors need to provide for efficient STDP and non-volatility is the central dead-zone in **Figure 5B**, which the already reported memristor from Michigan University (Jo et al., 2009) seems to present. Another issue relates to the quadratic type of mSTDP followed by the presented devices and architectures. This is a quite unusual form of STDP, which needs to be further investigated from a theoretical point of view. In general, there might be stability issues with generic STDP when used in complex biological models (Izhikevich and Desai, 2003;

Watt and Desai, 2010). Similarly, since the presented approach allows the shape of the neural spikes, and therefore the shape of the STDP learning curves to be changed in time, further theoretical studies are required to incorporate time varying STDP learning functions for speeding up, stabilizing, or in general improving learning performance.

ACKNOWLEDGMENTS

This work was supported by EU grant 216777 (NABAB), Spanish grants (with support from the European Regional Development Fund) TEC2006-11730-C03-01 (SAMANTA2), TEC2009-10639-C04-01 (VULCANO), and Andalusian grant P06TIC01417 (Brain System). Carlos Zamarreño-Ramos was supported by a national FPU scholarship. Timothée Masquelier was supported by the Fyssen Foundation.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at http://www.frontiersin.org/Neuromorphic_Engineering/10.3389/fnins.2011.00026/abstract

REFERENCES

- Affifi, A., Ayatollahi, A., and Raissi, F. (2009). STDP implementation using memristive nanodevice in CMOS-Nano neuromorphic networks. *IEICE Electron. Express* 6, 148–153.
- Agnus, G., Filoramo, A., Bourgoin, J.-P., Derycke, V., and Zhao, W. (2010). “Carbon nanotube-based programmable devices for adaptive architectures,” in *Proceedings of the IEEE International Symposium on Circuits and System (ISCAS 2010)*, Paris, 1667–1670.
- Argall, F. (1968). Switching phenomena in titanium oxide thin films. *Solid State Electron.* 11, 535–541.
- Azadmehr, M., Abrahamsen, J., and Häfliger, P. (2005). “A foveated AER imager chip,” in *Proceedings of the IEEE International Symposium on Circuits and System (ISCAS 2005)*, Kobe, 2751–2754.
- Berge, H. K. O., and Häfliger, P. (2005). “High-Speed Serial AER on FPGA,” in *Proceedings of the IEEE International Symposium on Circuits and System (New Orleans: ISCAS 2007)*, 857–860.
- Bi, G., and Poo, M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *J. Neurosci.* 18, 10464–10472.
- Bi, G., and Poo, M. M. (2001). Synaptic modification by correlated activity: Hebb’s postulate revisited. *Ann. Rev. Neurosci.* 24, 139–166.
- Bichler, O., Zhao, W., Gamrat, C., Alibart, F., Pleutin, S., and Vuillaume, D. (2010). “Development of a functional model for the nanoparticle-organic memory transistor,” in *Proceedings of the IEEE International Symposium on Circuits and System (ISCAS 2010)*, Paris, 1663–1666.
- Biolek, Z., Biolek, D., and Biolkova, V. (2009). Spice model for memristor with nonlinear dopant drift. *Radioengineering* 18, 210–214.
- Boahen, K. (1996). “Retinomorph vision systems,” in *Microneuro’96: Fifth International Conference on Neural Networks and Fuzzy Systems*, Lausanne.
- Boahen, K. (1999). “Retinomorph chips that see quadruple images,” in *Proceedings of International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems (Microneuro99)*, Granada, 12–20.
- Boahen, K. (2000). Point-to-point connectivity between neuromorphic chips using address events. *IEEE Trans. Circuits Syst. II* 47, 416–434.
- Boahen, K. (2002). A retinomorph chip with parallel pathways: encoding INCREASING, ON, DECREASING, and OFF visual signals. *Int. J. Analog Integr. Circuits Signal Process.* 30, 121–135.
- Borghetti, J., Li, Z., Straznicki, J., Li, X., Ohlberg, D. A. A., Wu, W., Stewart, D. R., and Williams, R. S. (2009). A hybrid nanomemristor/transistor logic circuit capable of self-programming. *Proc. Natl. Acad. Sci. U.S.A.* 106, 1699–1703.
- Camuñas-Mesa, L., Acosta-Jiménez, A., Zamarreño-Ramos, C., Serrano-Gotarredona, T., and Linares-Barranco, B. (in press). A 32×32 convolution processor chip for address event vision sensors with 155ns event latency and 20meps throughput. *IEEE Trans. Circuits Syst. I*. doi: 10.1109/TCSI.2010.2078851
- Camuñas-Mesa, L., Pérez-Carrasco, J. A., Zamarreño-Ramos, C., Serrano-Gotarredona, T., and Linares-Barranco, B. (2010). “On scalable spiking convent hardware for cortex-like visual sensory processing systems,” *IEEE International Symposium On Circuits and System (Paris: ISCAS 2010)*, 249–252.
- Cassenaer, S., and Laurent, G. (2007). Hebbian STDP in mushroom bodies facilitates the synchronous flow of olfactory information in locusts. *Nature* 448, 709–713.
- Cauwenberghs, G., Kumar, N., Himmelbauer, W., and Andreou, A. G. (1998). An analog VLSI Chip with asynchronous interface for auditory feature extraction. *IEEE Trans. Circuits Syst. II* 45, 600–606.
- Chan, V., Liu, S.-C., and van Schaik, A. (2007). AER EAR: a matched silicon cochlea pair with address event representation interface. *IEEE Trans. Circuits Syst. I* 54, 48–59.
- Chen, S., and Bermak, A. (2007). Arbitrated time-to-first spike CMOS image sensor with on-chip histogram equalization. *IEEE Trans. VLSI Syst.* 15, 346–357.
- Chicca, E., Whatley, A. M., Lichtsteiner, P., Dante, V., Delbruck, T., Del Giudice, P., Douglas, R. J., and Indiveri, G. (2007). A multichip pulse-based neuromorphic infrastructure and its application to a model of orientation selectivity. *IEEE Trans. Circuits Syst. I* 54, 981–993.
- Choi, T. Y. W., Merolla, P., Arthur, J., Boahen, K., and Shi, B. E. (2005). Neuromorphic implementation of orientation hypercolumns. *IEEE Trans. Circuits Syst. I* 52, 1049–1060.
- Chua, L. O. (1971). Memristor – the missing circuit element. *IEEE Trans. Circuit Theory* 18, 507–519.
- Chua, L. O., Desoer, C. A., and Kuh, E. S. (1987). *Linear and Nonlinear Circuits*. New York: McGraw-Hill.
- Chua, L. O., and Kang, S. M. (1976). Memristive devices and systems. *Proc. IEEE* 64, 209–223.
- Costas-Santos, J., Serrano-Gotarredona, T., Serrano-Gotarredona, R., and Linares-Barranco, B. (2007). A contrast retina with on-chip calibration for neuromorphic spike-based AER vision systems. *IEEE Trans. Circuits Syst. I: Reg. Papers* 54, 1444–1458.
- Culurciello, E., Etienne-Cummings, R., and Boahen, K. A. (2003). A biomorphic digital image sensor. *IEEE J. Solid-State Circuits* 38, 281–294.
- Delbruck, T. “Driving in pasadena to the post office,” in *Vision Project JAER Download Area*. Available at: <http://sourceforge.net/apps/trac/jaer/wiki/AER%20data>
- Delorme, A., Perrinet, L., and Thorpe, S. J. (2001). Networks of integrate-and-fire neurons using rank order coding B: spike timing dependent plasticity and emergence of orientation selectivity. *Neurocomputing* 38–40, 539–545.
- Feldman, D. (2000). Timing-based LTP and LTD at vertical inputs to layer II/III pyramidal cells in rat barrel cortex. *Neuron* 27, 45–56.
- Finelli, L. A., Haney, S., Bazhenov, M., Stopfer, M., and Sejnowski, T. J. (2008). Synaptic learning rules and sparse

- coding in a model sensory system. *PLoS Comput. Biol.* 4, e1000062. doi: 10.1371/journal.pcbi.1000062
- Gerstner, W. (1999). "Spiking neurons," Chapter 1, in *Pulsed Neural Networks*, eds W. Maass and C. M. Bishop (Cambridge, MA: MIT Press), 3–53.
- Gerstner, W., Kempter, R., Leo van Hemmen, J., and Wagner, H. (1996). A neuronal learning rule for sub-millisecond temporal coding. *Lett. Nat.* 383, 76–78.
- Gerstner, W., Ritz, Z., and Hemmen, J. L. (1993). Why spikes? Hebbian learning and retrieval of time-resolved excitation patterns. *Biol. Cybern.* 69, 503–515.
- Goodman, D., and Brette, R. (2008). Brian: a simulator for spiking neural networks in python. *Front. Neuroinformatics* 2:5. doi: 10.3389/neuro.11.005.2008
- Green, J. E., Choi, J. W., Boukai, A., Bunimovich, Y., Johnston-Halperin, E., DeIonno, E., Luo, Y., Sheriff, B. A., Xu, K., Shin, Y. S., H.-R. Tseng, Stoddart, J. F., and Heath, J. R. (2007). A 160-kilobit molecular electronic memory patterned at bits per square centimetre. *Nature* 445, 414–417.
- Gütig, R., Aharonov, R., Rotter, S., and Sompolinsky, H. (2003). Learning input correlations through nonlinear temporally asymmetric Hebbian plasticity. *J. Neurosci.* 23, 3697–3714.
- Guyonneau, R., VanRullen, R., and Thorpe, S. J. (2004). Temporal codes and sparse representations: a key to understanding rapid processing in the visual system. *J. Physiol. Paris* 98, 487–497.
- Hebb, D. O. (1949). *The Organization of Behavior: A Neuropsychological Study*. New York: Wiley.
- Hubel, D. H., and Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. *J. Physiol.* 148, 574–591.
- Izhikevich, E. M., and Desai, N. S. (2003). Relating STDP to BCM. *Neural Comput.* 15, 1511–1523.
- Jacob, V., Brasier, D. J., Erchova, I., Feldman, D., and Shulz, D. E. (2007). Spike timing-dependent synaptic depression in the in vivo barrel cortex of the rat. *J. Neurosci.* 27, 1271–1284.
- Jeon, H.-J. Kim, K. H., Baek, Y.-K. Kim, D. W., and Jung H. -T. (2010). New top-down approach for fabricating high-aspect-ratio complex nanostructures with 10nm scale features. *Nano Lett.* 10, 3604–3610.
- Jeong, H. Y., Kim, J. Y., Kim, J. W., Hwang, J. O., Kim, J. E., Lee, J. Y., Yoon, T. H., Cho, B. J., Kim, S. O., Ruoff, R. S., and Choi, S. Y. (2010). Graphene oxide thin films for flexible nonvolatile memory applications. *Nano Lett.* 10, 4381–4386.
- Jo, S. H., Chang, T., Ebong, I., Bhadviya, B. B., Mazumder, P., and Lu, W. (2010). Nanoscale memristor device as synapse in neuromorphic systems. *Nano Lett.* 10, 1297–1301.
- Jo, S. H., Kim, K.-H., and Lu, W. (2009). High-density crossbar arrays based on a Si memristive system. *NANO Lett.* 9, 870–874.
- Joglekar, Y. N., and Wolf, S. J. (2009). The elusive memristor: properties of basic electrical circuits. *Eur. J. Phys.* 30, 661–675.
- Jung, G.-Y. Johnston-Halperin, E., Wu, W., Yu, Z., Wang, S.-Y. Tong, W. M., Li, Z., Green, J. E., Sheriff, B. A., Boukai, A., Bunimovich, Y., Heath, J. R., and Williams, R. S. (2006). Circuit fabrication at 17 nm half-pitch by nanoimprint lithography. *Nano Lett.* 351–354.
- Lai, Q., Li, Z., Zhang, L., Li, X., Stickle, W. F., Zhu, Z., Gu, Z., Kamins, T. I., Williams, R. S. and Chen, Y. (2008). An organic/Si nanowire hybrid field configurable transistor. *Nano Lett.* 8, 876–880.
- Lazzaro, J., Wawrzynek, J., Mahowald, M., Silvillotti, M., and Gillespie, D. (1993). Silicon auditory processors as computer peripherals. *IEEE Trans. Neural Netw.* 4, 523–528.
- Leñero-Bardallo, J. A., Serrano-Gotarredona, T., and Linares-Barranco, B. (2010). A five-decade dynamic-range ambient-light-independent calibrated signed-spatial-contrast AER retina with 0.1ms latency and optional time-to-first-spike mode. *IEEE Trans. Circuits Syst. I* 57, 2632–2643.
- Leñero-Bardallo, J. A., Serrano-Gotarredona, T., and Linares-Barranco, B. (2011). A 3.6 us latency frame-free event-driven dynamic vision sensor. *IEEE J. Solid-State Circuits*.
- Lichtsteiner, P., Posch, C., and Delbruck, T. (2008). A 128 × 128 120dB 30mW asynchronous vision sensor that responds to relative intensity change. *IEEE J. Solid-State Circuits* 43, 566–576.
- Linares-Barranco, B., and Serrano-Gotarredona, T. (2009). "Memristance can explain spike-time-dependent plasticity in neural synapses," in *Nature precedings*, Available at: <http://hdl.handle.net/10101/npre.2009.3010.1>
- Lubenov, E. V., and Siapas, A. G. (2008). Decoupling through synchrony in neuronal circuits with propagation delays. *Neuron* 58, 118–131.
- Mahowald, M. A. (1992). *VLSI Analogs of Neuronal Visual Processing: A Synthesis of Form and Function*. Ph.D. Computation and Neural Systems, Caltech, Pasadena, California.
- Mallik, U., Clapp, M., Choi, E., Cauwenberghs, G., and Etienne-Cummings, R. (2005). Temporal change threshold detection imager. *IEEE ISSCC Digest of Technical Papers*, San Francisco, 362–363.
- Markram, H., Lübke, J., Frotscher, M., and Sakmann, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic APS and EPSPS. *Science* 275, 213–215.
- Martin, A. J., and Nyström, M. (2006). Asynchronous techniques for system-on-chip design. *Proc. IEEE* 94, 1089–1120.
- Masquelier, T., Guyonneau, R., and Thorpe, S. J. (2008). Spike timing dependent plasticity finds the start of repeating patterns in continuous spike trains. *PLoS ONE* 3, e1377. doi:10.1371/journal.pone.0001377
- Masquelier, T., Guyonneau, R., and Thorpe, S. J. (2009a). Competitive STDP-based spike pattern learning. *Neural Comput.* 21, 1259–1276.
- Masquelier, T., Hugues, E., Deco, G., and Thorpe, S. J. (2009b). Oscillations, phase-of-firing coding and spike timing-dependent plasticity: an efficient learning scheme. *J. Neurosci.* 29, 13484–13493.
- Masquelier, T., and Thorpe, S. J. (2007). Unsupervised learning of visual features through spike timing dependent plasticity. *PLoS Comput. Biol.* 3, e31. doi: 10.1371/journal.pcbi.0030031
- Masquelier, T., and Thorpe, S. J. (2010). "Learning to recognize objects using waves of spikes and spike timing-dependent plasticity," in *Proceedings of the 2010 IEEE International Joint Conference on Neural Networks*, Barcelona. doi: 10.1109/IJCNN.2010.5596934
- Massari, N., Gottardi, M., and Jawed, S. (2008). A 100uW 64 × 128-Pixel contrast-based asynchronous binary vision sensor for wireless sensor networks. *IEEE ISSCC Digest of Technical Papers*, San Francisco, 588–638.
- Mead, C. (1989). *Analog VLSI and Neural Systems*. Reading, MA: Addison Wesley.
- Mortara, A., Vitto, E. A., and Venier, P. (1995). A communication scheme for analog VLSI perceptive systems. *IEEE J. Solid-State Circuits* 30, 660–669.
- Mu, Y., and Poo, M. M. (2006). Spike timing-dependent LTP/LTD mediates visual experience-dependent plasticity in a developing retinotectal system. *Neuron* 50, 115–125.
- Oster, M., Wang, Y., Douglas, R., and Liu, S.-C. (2008). Quantification of a spike-based winner-take-all VLSI network. *IEEE Trans. Circuits Syst. I* 55, 3160–3169.
- Pérez-Carrasco, J. A., Zamarreño-Ramos, C., Serrano-Gotarredona, T., and Linares-Barranco, B. (2010a). "On neuromorphic spiking architectures for asynchronous stdp memristive systems," in *Proceedings of the IEEE International Symposium on Circuits and System (ISCAS 2010)*, Paris, 1659–1662.
- Pérez-Carrasco, J. A., Acha, B., Serrano-Gotarredona, T., and Linares-Barranco, B. (2010b). Fast vision through frame-less event-based sensing and convolutional processing. application to texture recognition. *IEEE Trans. Neural Netw.* 21, 609–620.
- Pershin, Y. V., and Di Ventra, M. (2010). Practical approach to programmable analog circuits with memristors. *IEEE Trans. Circuits Syst. I* 57, 1857–1864.
- Pershin, Y. V., and Di Ventra, M. (2010). Experimental demonstration of associative memory with memristive neural networks. *Neural Netw.* 23, 881.
- Porr, B., and Wörgötter, F. (2004). How the shape of pre- and postsynaptic signals can influence STDP: A biophysical model. *Neural Comput.* 16, 595–625.
- Posch, C., Hofstatter, M., Matolin, D., Vanstraelen, G., Schon, P., Donath, N., and Litzenberger, M. (2007). A dual-line optical transient sensor with on-chip precision time-stamp generation. *IEEE ISSCC Digest of Technical Papers*, San Francisco, 500–618.
- Posch, C., Matolin, D., and Wohlgenannt, R. (2010). A QVGA 143dB dynamic range asynchronous address-event PWM dynamic image sensor with lossless pixel-level video compression. *ISSCC Digest of Technical Papers*, San Francisco, 400–401.
- Ramakrishnan, S., Hasler, P., and Gordon, C. (2010). "Floating gate synapses with spike time dependent plasticity," in *Proceedings of the IEEE International Symposium on Circuits and System (ISCAS 2010)*, Paris, 369–372.
- Rao, N. R. P., and Sejnowski, T. J. (2001). Spike-time-dependent Hebbian plasticity as temporal difference learning. *Neural Comput.* 13, 2221–2237.
- Rubin, J., Lee, D. D., and Sompolinsky, H. (2001). Equilibrium properties of temporally asymmetric Hebbian plasticity. *Phys. Rev. Lett.* 86, 364–367.
- Rubin, J. E., Gerkin, R. C., G-Bi, Q., and Chow, C. C. (2005). Calcium time course as a signal for spike-timing-dependent plasticity. *J. Neurophysiol.* 93, 2600–2613.
- Ruedi, P. F., Heim, P., Kaess, F., Grenet, E., Heitger, F., Burgi, P.-Y., Gyger, S., and Nussbaum, P. (2003). A 128 × 128, Pixel 120-dB dynamic-range vision-sensor chip for image contrast and orientation extraction. *IEEE J. Solid-State Circuits* 38, 2325–2333.
- Ruedi, P. F., Heim, P., Gyger, S., Kaess, F., Arm, C., Caseiro, R., Nagel, J.-L., and Todeschini, S. (2009). An SoC combining a 132dB QVGA pixel array and a 32b DSP/MCU processor for vision

- applications. *IEEE ISSCC Digest of Technical Papers*, San Francisco, 46–47.
- Sarpeshkar, R., Baker, M. W., Salthouse, C. D., Sit, J.-J., Turicchia, L., and Zhak, S. M. (2005). An analog bionic ear processor with zero-crossing detection. *ISSCC Digest of Technical Papers*, San Francisco, 78–79.
- Serrano-Gotarredona, R., Serrano-Gotarredona, T., Acosta-Jimenez, A., and Linares-Barranco, B. (2006). A neuromorphic cortical-layer microchip for spike-based event processing vision systems. *IEEE Trans. Circuits Syst. I* 53, 2548–2566.
- Serrano-Gotarredona, R., Serrano-Gotarredona, T., Acosta-Jimenez, A., Serrano-Gotarredona, C., Perez-Carrasco, J. A., Linares-Barranco, B., Linares-Barranco, A., Jimenez-Moreno, G., and Civit-Ballcells, A. (2008). On Real-Time AER 2D convolutions hardware for neuromorphic spike based cortical processing. *IEEE Trans. Neural Netw.* 19, 1196–1219.
- Serrano-Gotarredona, R., Oster, M., Lichtsteiner, P., Linares-Barranco, A., Paz-Vicente, R., Gomez-Rodriguez, F., Camunas-Mesa, L., Berner, R., Rivas-Perez, M., Delbruck, T., Liu, S. C., Douglas, R., Haflinger, P., Jimenez-Moreno, G., Ballcells, A. C., Serrano-Gotarredona, T., Acosta-Jimenez, A. J., and Linares-Barranco, B. (2009). CAVIAR: a 45k neuron, 5M synapse, 12G connects/s AER hardware sensory-processing-learning-actuating system for high-speed visual object recognition and tracking. *IEEE Trans. Neural Netw.* 20, 1417–1438.
- Serrano-Gotarredona, T., Andreou, A. G., and Linares-Barranco, B. (1999). AER image filtering architecture for vision processing systems. *IEEE Trans. Circuits Syst. I* 46, 1064–1071.
- Shin, S., Kim, K., and S.-Kang, M. (2010). Compact models for memristors based on charge-flux constitutive relationships. *IEEE Trans. CAD Int. Circuits Syst.* 29, 590–598.
- Sivilotti, M. (1991). *Wiring Considerations in Analog VLSI Systems with Application to Field-Programmable Networks*. Ph.D. Computation and Neural Systems, Caltech, Pasadena, California.
- Sjöström, J., and Gerstner, W. (2010). Spike-timing dependent plasticity. *Scholarpedia* 5, 1362. Available at: <http://www.scholarpedia.org/article/STDP>
- Snider, G. S. (2007). Self-organized computation with unreliable, memristive nanodevices. *Nanotechnology* 18, 365202.
- Snider, G. S. (2008). Spike-timing-dependent learning in memristive nanodevices. *IEEE International Symposium on Nano Architectures (NANOARCH 2008)*, 85–92.
- Sparso, J., and Furber, S. B. (2001). *Principles of Asynchronous Circuit Design: A Systems Perspective*. Dordrecht: Kluwer Academic Publishers.
- Strukov, D. B., and Likharev, K. K. (2005). CMOL FPGA: a reconfigurable architecture for hybrid digital circuits with two-terminal nanodevices. *Nanotechnology* 16, 888–900.
- Strukov, D. B., Snider, G. S., Stewart, D. R., and Williams, R. S. (2008). The missing memristor found. *Nature* 453, 80–83.
- Swaroop, B., West, W. C., Martinez, G., Kozicki, M. N., and Akers, L. A. (1998). “Programmable current mode Hebbian learning neural network using programmable metallization cell,” in *Proceedings of the IEEE International Symposium on Circuits and System (ISCAS 1998)*, Monterey, 33–36.
- Teixeira, T., Andreou, A. G., and Culurciello, E. (2005). “Event-based imaging with active illumination in sensor networks,” in *Proceedings of the IEEE International Symposium on Circuits and System (ISCAS 2005)*, Kobe, 644–647.
- van Rossum, M. C. W., Bi, G. Q., and Turrigiano, G. G. (2000). Stable Hebbian learning from spike timing-dependent plasticity. *J. Neurosci.* 20, 8812–8821.
- Vernier, P., Mortara, A., Arreguit, X., and Vittoz, E. A. (1997). An integrated cortical layer for orientation enhancement. *IEEE J. Solid-State Circuits* 32, 177–186.
- Watt, A. J., and Desai, N. S. (2010). Homeostatic plasticity and STDP: keeping a neuron’s cool in a fluctuating world. *Front. Syn. Neurosci.* 2:5. doi: 10.3389/fnsyn.2010.00005
- Weidenbacher, U., and Neumann, H. (2008). “Unsupervised learning of head pose through spike-timing dependent plasticity,” in *Perception in Multimodal Dialogue Systems, Lecture Notes in Computer Science*, eds E. André, L. Dybkjær, W. Minker, H. Neumann, R. Pieraccini, and M. Weber (Berlin: Springer), 123–131.
- Wen, B., and Boahen, K. (2006). A 360-channel speech preprocessor that emulates the cochlear amplifier. *ISSCC Dig. Tech. Papers*, San Francisco, 556–557.
- White, L. E., Coppola, D. M., and Fitzpatrick, D. (2001). The contribution of sensory experience to the maturation of orientation selectivity in ferret visual cortex. *Lett. Nat.* 411, 1049–1052.
- Young, J. M. (2007). Cortical reorganization consistent with spike timing-but not correlation-dependent plasticity. *Nat. Neurosci.* 10, 887–895.
- Zaghloul, K. A., and Boahen, K. (2004a). Optic nerve signals in a neuromorphic chip: Part 1. *IEEE Trans. Biomed. Eng.* 51, 657–666.
- Zaghloul, K. A., and Boahen, K. (2004b). Optic nerve signals in a neuromorphic chip: Part 2. *IEEE Trans. Biomed. Eng.* 51, 667–675.
- Zhang, L., Tao, H., Holt, C., Harris, W., and Poo, M. (1998). A critical window for cooperation and competition among developing retinotectal synapses. *Nature* 395, 37–44.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Received: 18 September 2010; accepted: 19 February 2011; published online: 17 March 2011.

Citation: Zamarreño-Ramos C, Camuñas-Mesa LA, Pérez-Carrasco JA, Masquelier T, Serrano-Gotarredona T and Linares-Barranco B (2011) On spike-timing-dependent-plasticity, memristive devices, and building a self-learning visual cortex. *Front. Neurosci.* 5:26. doi: 10.3389/fnins.2011.00026

This article was submitted to *Frontiers in Neuromorphic Engineering, a specialty of Frontiers in Neuroscience*.

Copyright © 2011 Zamarreño-Ramos, Camuñas-Mesa, Pérez-Carrasco, Masquelier, Serrano-Gotarredona and Linares-Barranco. This is an open-access article subject to an exclusive license agreement between the authors and *Frontiers Media SA*, which permits unrestricted use, distribution, and reproduction in any medium, provided the original authors and source are credited.