Memristors Empower Spiking Neurons With Stochasticity

Maruan Al-Shedivat, *Student Member, IEEE*, Rawan Naous, *Member, IEEE*, Gert Cauwenberghs, *Fellow, IEEE*, and Khaled Nabil Salama, *Senior Member, IEEE*

***Abstract*—Recent theoretical studies have shown that probabilistic spiking can be interpreted as learning and inference in cortical microcircuits. This interpretation creates new opportunities for building neuromorphic systems driven by probabilistic learning algorithms. However, such systems must have two crucial features: 1) the neurons should follow a specific behavioral model, and 2) stochastic spiking should be implemented efficiently for it to be scalable. This paper proposes a memristor-based stochastically spiking neuron that fulfills these requirements. First, the analytical model of the memristor is enhanced so it can capture the behavioral stochasticity consistent with experimentally observed phenomena. The switching behavior of the memristor model is demonstrated to be akin to the firing of the stochastic spike response neuron model, the primary building block for probabilistic algorithms in spiking neural networks. Furthermore, the paper proposes a neural soma circuit that utilizes the intrinsic nondeterminism of memristive switching for efficient spike generation. The simulations and analysis of the behavior of a single stochastic neuron and a winner-take-all network built of such neurons and trained on handwritten digits confirm that the circuit can be used for building probabilistic sampling and pattern adaptation machinery in spiking networks. The findings constitute an important step towards scalable and efficient probabilistic neuromorphic platforms.**

***Index Terms*—Neuromorphic systems, probabilistic inference, probabilistic learning, spiking neurons, stochastic computing, stochastic memristors, winner-take-all.**

I. Introduction

In RECENT research on artificial and biological neural networks, a variety of neuron models have been proposed and extensively studied [1]. Detailed models based on Hodgkin-Huxley equations were successful in describing the formation of the action potential in biological neurons [2]. On the other hand, more abstract and simpler models based on the leaky integrate-and-fire (I&F) mechanism helped to clarify the pulsing and bursting behavior of neurons and simple neural circuits [3]. The I&F neuron model cut the concept of the spike down to an abstract event and the concept of the neural computation down to a transformation of input spike sequences into an output sequence.

Current thinking in neuroscience suggests that biological neural noise [4] is beneficial to information processing in nonlinear systems [5] and is essential for computation and learning in cortical microcircuits [6]. A recent study showed that intrinsically probabilistic spiking based on the spike response model (SRM) with stochastic firing intensity is in good agreement with experimental cortical data [7]. Subsequently, it was found that the behavior of a network built of probabilistic SRM neurons under a simple spike-timing dependent plasticity (STDP) learning rule could be seen as Bayesian computation [8]. Such neural networks can perform probabilistic sampling, inference, and learning algorithms [9]–[11], and serve as building blocks for biologically plausible implementations of Boltzmann machines [12], [13] and deep belief networks [14]. However, despite rapid advances in this area, the idea that noise is beneficial is rather recent, and deterministic neural models remain dominant in the literature.

Neuromorphic engineering [15] aims to mimic neural architectures directly in hardware and utilize theoretical developments of neuroscience to design efficient silicon neurons as analog or digital very large-scale integration (VLSI) circuits. In this line of research, silicon neurons are decomposed into three functional blocks: the synapse (spike receiver), the soma (spatio-temporal input signal integrator), and the analog or digital spike generator; application-dependent implementations for each of these blocks have been extensively studied [16].

Past research on neuromorphic hardware has mainly focused on generalized deterministic I&F neurons and has overlooked the possibility of building *intrinsically* probabilistic spiking units. Should a hardware implementation of a neural network be endowed with stochasticity, additional uncorrelated background noise must be injected into the circuitry of every neuron [13], [17], [18]. This straightforward approach of noise injections not only lacks power efficiency and constrains scalability, but it also provides only an approximation of the stochastic SRM [19]. Research on neuro-inspired integrated circuits has highlighted the demand for building natively stochastic efficient and scalable hardware [20].

Circuits based on memristors have become a recent trend in computing and neuromorphic engineering due to low-power requirement and compactness of such nano-devices [21], [22]. It was demonstrated that memristance can explain STDP learning

Manuscript received November 01, 2014; revised January 28, 2015 and April 03, 2015; accepted April 27, 2015. Date of current version June 09, 2015. This work was supported by King Abdullah University of Science and Technology (KAUST), Saudi Arabia. This paper was recommended by Guest Editor T. Prodromakis.

M. Al-Shedivat, R. Naous, and K. N. Salama are with the Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955, Saudi Arabia (e-mail: khaled.salama@kaust.edu.sa).

G. Cauwenberghs is with the Institute for Neural Computation and the Department of Bioengineering, University of California, San Diego, La Jolla, CA 92093 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JETCAS.2015.2435512

in biological synapses [23]. Furthermore, memristive neuromorphic synapses have been extensively studied [24]–[26] and compared to other systems [27]–[29]. The deterministic STDP learning rule was found to be challenging to implement due to the complicated multilevel device programming schemes. However, recently discovered nondeterministic behaviors of memristive switching [30], [31] based on the stochasticity of nano-filament formation in thin amorphous silicon and metal–oxide films [32] provided a way to overcome these challenges: it was demonstrated that spiking networks can be trained with alternative probabilistic binary synapses [33], [34] or through the compound multi-device probabilistic STDP rule [35].

Building upon the same phenomenon of memristive nondeterminism, here we propose an implementation of intrinsically stochastic spiking neuron. Previously, deterministic memristor models were used to build action potential formation mechanisms similar to Hodgkin-Huxley model [36], [37]. It was also proposed that memristive stochasticity could be used to build binary and continuous-value nonspiking artificial neurons [38] and for adding variability to spike trains generated by leaky I&F neurons [39]. Among other contributions and in contrast to the previous proposals, our design utilizes abrupt memristive switching directly for generating spontaneous events that can be further shaped into spikes. Importantly, we show that, according to the experimental results [30], [40], probabilistic switching of metal–oxide and amorphous silicon memristors in the sub-threshold regime replicates the firing of the stochastic SRM. Thus, without any noise injection, the proposed memristive circuit provides an efficient, low-power, and scalable alternative method for building stochastic spiking neurons. Through simulations we confirm that our design can be efficiently used in applications.

The remainder of this paper is organized as follows. In Section II, we discuss different types of memristive switching, propose a simple way of enhancing several widely used behavioral memristor models with stochasticity, and provide a comprehensive analysis. In Section III, we describe stochastic SRM and present an implementation of the neural soma that precisely follows this model. Learning and inference capabilities of a probabilistic winner-take-all network built of our spiking neurons are demonstrated in Section IV, followed by a related work discussion. Finally, we compare our proposal with alternative systems in Section V and conclude with a summary in Section VI.

II. Memristor Models and Stochasticity

Two terminal resistive switching devices (memristors) can exhibit nonlinear analog and nondeterministic digital dynamics. These devices are usually fabricated with metal/insulator/metal (MIM) structure [Fig. 1(a)]. They are classified with respect to the switching mechanism and the corresponding type of insulator material [40]: electrochemical metalization (ECM) cells (also known as CBRAM) exhibit resistive switching based on cation migration (e.g., Ag/Cu ions), while oxide-based devices (also named valency change memory (VCM) or resistive RAM) switch due to migration of anions such as oxygen ions or oxygen

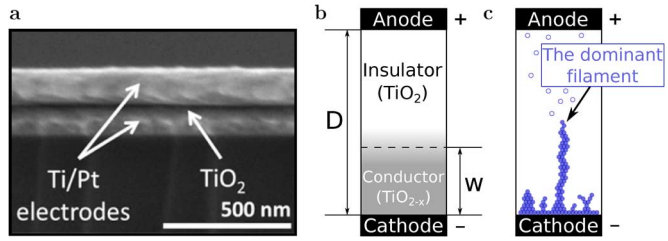

Fig. 1. Memristor microscopy and filament formation. (a) A scanning electron microscope photo of a TiO_2 memristor cross-section (reprinted from [42]). (b) An illustration of a simple barrier memristor model. (c) Filament formation process in a thin-film.

vacancies. Importantly, both types of the devices can dynamically change their resistances due to the conductive filament formation/rupture processes [32], [41] which are intrinsically probabilistic.

Nondeterminism of the memristive switching is at the heart of the proposed spiking neural circuit. Therefore, to simulate the behavior of our neuron, we enhance several popular deterministic models of the memristor to account for probabilistic switching effects.

A. Deterministic Memristor Models

Pickett's model [43] is a highly regarded memristor model because of its experimental nature. It describes the behavior of TiO_2 nano-film resistance through the underlying physical mechanism-a drift/diffusion process of ionized dopants. On the other hand, a number of behavioral models (e.g., those by Biolek [44], [45] or Prodromakis [46] or the boundary condition model (BCM) [47]) simplify the memristance mechanism to some useful abstractions that are eventually fitted to the experimentally observed behaviors.

Most of the simple behavioral models regard the memristor as a two-layer thin film: an insulating layer (e.g., TiO_2) connected in series to a conductive layer (e.g., TiO_{2-x}). Assuming that the width of the first layer is $w(t)$ and the width of the whole memristive film is D [Fig. 1(b)], we can write the following general deterministic model for a voltage-driven memristor [48]–[50]:

$$\begin{cases} I(t) = g_M\left(\frac{w(t)}{D}\right) V(t), \\ \frac{dw}{dt} = f\left(w(t), V(t)\right) \end{cases} \tag{1}$$

where $V(t)$ is the input voltage, $I(t)$ is the current in the device, and $g_M(\cdot)$ is the so-called memconductunce—the memristive conductance that depends on $w(t)$ and hence on the history of the input signal, $V(t)$. Notice that the state variable in such models, $w(t)$, represents the width of the conductive layer that functionally depends on time: $w(t) = \int f(w(t), V(t))dt$. To make our notation convenient, we further denote $V(t)$, $I(t)$, and $w(t)$ as V, I, and w, respectively, omitting explicit dependence on time.

Memconductance can be rewritten in the following form:

$$g_M\left(\frac{w}{D}\right) = \left(R_{\text{off}} - \Delta R\frac{w}{D}\right)^{-1} \tag{2}$$

where $\Delta R = R_{\text{off}} - R_{\text{on}}$, the low resistance state (LRS), R_{on}, and the high resistance state (HRS), R_{off}, are the memristor's resistances corresponding to the whole nano-film enriched with

or depleted of oxygen vacancies, respectively. All behavioral models have the same form, (1)–(2), and diverge only in the choice of the memristance kinetics function, $f(w, V)$ [44]–[47], [51]. These models are designed to capture the *I-V* hysteresis and can closely replicate the experimentally observed behaviors [48], [52].

B. Stochasticity of Switching

Pickett's model is based on a quantitatively supported hypothesis that electronic conduction in metal–oxide films is dominated by an effective tunneling barrier that varies with time under the applied voltage [43]. Under this assumption, the evolution of the model's state variable, $w(t)$, is governed by a nonlinear diffusion process. This leads to deterministic and smooth average conductance evolution.

On the contrary, recent experimental studies suggest that memristive conductance behavior is better described by the formation of conductive filaments in a dielectric film. In the sub-threshold voltage regime (also known as *weak programming conditions* [34]), a single, dominant nano-scale filament is formed [Fig. 1(c)] [32], [40], [53]. Not only is this formation process experimentally shown to be stochastic, but the memristor also experiences an abrupt conductance change once the filament is formed [30], [31]. The nonlinear diffusion process in metal-oxide films can be seen as a result of the formation of conducting nano-filaments by oxygen vacancy migration [54], although it certainly does not account for the evident stochasticity of this process under the sub-threshold voltage conditions.

We show two different ways of enhancing the deterministic memristor model to account for stochastic changes of the internal state, w. While several approaches based on modeling the physics of the filament formation are available in the literature [55], [56], our models are much simpler and less computationally demanding. We aim to capture the experimentally observed stochasticity of the device regardless of the actual physical causes of such behavior. Our first approach approximates stochastic switching with instantaneous jumps of the state, w, and is independent of its deterministic evolution described by $f(w, V)$. The second one, on the contrary, uses $f(w, V)$ to smooth the kinetics of the switching while preserving the stochasticity.

1) Model-independent Instantaneous Stochastic Switching: Once a dominant conductive filament is formed, memristor experiences an abrupt resistance jump from the current value to $R_{\rm on}$, and this jump can hence be seen as instantaneous on a certain timescale. Following the experimental findings in [30], we assume that the memristors switch stochastically. Moreover, the switching events can be described by a Poisson process [57]. If no switching happens spontaneously, we can assume that memristor would follow its average deterministic model, which could be one of the models described in [43]–[47]. In this case, such behavior can be described mathematically as follows:

$$\begin{cases} I = g_M\left(\frac{w}{D}\right) V, \quad \text{where } g_M\left(\frac{w}{D}\right) = \left(R_{\rm off} - \Delta R \frac{w}{D}\right)^{-1}, \\ dw = \underbrace{f(w,V)dt}_{\text{deterministic term}} + \underbrace{(\theta(V)\cdot D - w)\, dN(\tau)}_{\text{stochastic term}} \end{cases} \tag{3}$$

TABLE I
STOCHASTIC BIOLEK'S MODEL PARAMETERS

V_0	τ_0	V_{T_0}	ΔV	$R_{\rm off}$	$R_{\rm on}$	$\Delta R_{\rm on/off}$
0.156 V	$2.85{\cdot}10^5$ s	3.0 V	0.2 V	1 MΩ	10 kΩ	5%

The parameters of the inhomogeneous Poisson process, V_0 and τ_0, are chosen according to [30] and [58]. The parameters of the memristance kinetics $f(w, V)$ are chosen to make the model transitions last about 1 μs. $\Delta R_{\rm on/off}$ is the standard deviation of the $R_{\rm on/off}$ that varies between the switching cycles according to a log-normal distribution [33].

where $N(\tau)$ is an inhomogeneous Poisson process with time constant $\tau(t)$, and $\theta(\cdot)$ is the Heaviside step function. According to (3), for positive input voltage V, the sate jump, $dw = D - w$, immediately leads to $w = D$, i.e., to the $R_{\rm on}$ state. With negative voltage, the state jump is $dw = -w$ and leads to the $R_{\rm off}$ state.

If we integrate (3) for the state variable, w, the deterministic term will describe the normal, smooth evolution of the conductive part of the nano-film, while the stochastic term will account for abrupt jumps that correspond to the dominant filament being fully formed. Notice that since we use the Poisson process, $N(\tau)$, as a mathematical abstraction of the state jumps, sub-threshold switching will be instantaneous. We further demonstrate that such abstraction is essential for neuromorphic applications: the switching time is negligible compared to the neural dynamics timescales; moreover, the memristor inside the neuron is used as a digital device.

According to the experimental measurements on amorphous silicon devices [30], [58], the time constant of inhomogeneous Poisson switching depends on the voltage drop across the memristor. This dependence has the following form:

$$\tau(V) = \tau_0 \exp\left(-\frac{V}{V_0}\right) \tag{4}$$

where τ_0 and V_0 are parameters of time and voltage units, respectively. These parameters can be found by fitting the model (4) to experimental data as in [30], [58]. The parameters used in our simulations are presented in Table I.

2) Model-based Stochastic Switching: The spontaneous switching, (3), disregards the kinetics of the transition from one state to the other and simply approximates it with an instantaneous jump. Even though such abrupt behavior is consistent with the experimentally observed single conductive filament growth [32], the precise physics of the filament formation is still under investigation and can vary for different types of memristive devices. Hence, we also show how the general deterministic model, (1)–(2), can be extended to a stochastic one that has smooth switching kinetics. In this paper, we consider a particular form of $f(w, V)$ that corresponds to Biolek's memristor with a threshold.

The evolution function in the Biolek's model takes the following form in our notation:

$$\begin{cases} f(w,V) = \phi(V)\cdot W(w,V), \\ \phi(V) = \beta\left(V - \frac{1}{2}\left(|V+V_T| - |V-V_T|\right)\right), \\ W(w,V) = \theta(V)\theta(D-w) + \theta(-V)\theta(w). \end{cases} \tag{5}$$

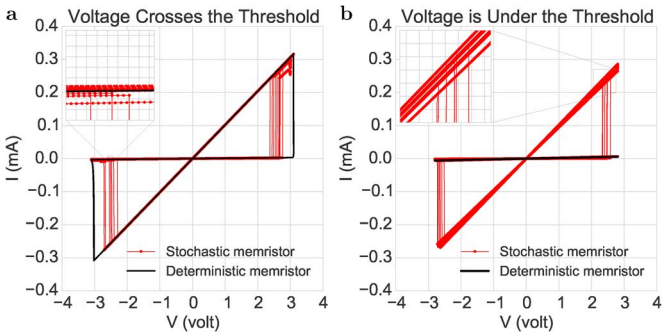

Fig. 2. *I-V* curves for Biolek's deterministic memristor model [44] (black) and its enhanced variant with stochastic switching (red). The deterministic model transits between the states only when the voltage is beyond the threshold of 3 V (a). It is unable to switch in the sub-threshold mode (b). The stochastic model experiences spontaneous jumps between the R_{off} and R_{on} states regardless of whether the applied voltages is above or below the threshold. The insets demonstrate that the model can easily account for the inherent cycle-to-cycle randomness of the LRS and the HRS states.

Here, $W(w, V)$ is a window function that constrains w to be non-negative and less or equal to D, while $\phi(V)$ governs the kinetics of the memristor state transition. The model can change its state only if the voltage crosses a threshold, V_T.

By having a dynamic stochastic threshold V_T, we can make Biolek's model, (5), account for the switching stochasticity while preserving the kinetics of the state transition

$$\begin{cases} dw = f(w,V)dt, \\ dV_T = \underbrace{\alpha \cdot \theta\left(V_{T_0} - V_T\right)dt}_{\text{deterministic term}} + \underbrace{\left(|V| - \Delta V - V_{T_0}\right)dN(\tau)}_{\text{stochastic term}}. \end{cases} \tag{6}$$

In this case, V_T depends on time, V_{T_0} is the initial threshold value, V is the input voltage, δV is a voltage margin parameter, and $N(\tau)$ is the same Poisson process as in (3). When a switching event occurs, V_T jumps immediately to $V - \text{sgn}(V)\delta V$ making $f(w, V) = \beta \text{sgn}(V)\delta V$ and allowing w to transit linearly to the corresponding state. The deterministic part of the V_T dynamics is responsible for bringing it back to V_{T_0}. Notice that when $\beta \to \infty$ (the case of BCM [47]), the switching becomes again instantaneous.

Fig. 2 demonstrates numerically simulated I–V curves for Biolek's model with deterministic and stochastic thresholds. The model can easily account for the cycle-to-cycle randomness of the state parameters, R_{on} and R_{off}, in real memristors [33] by resampling their values from the corresponding distributions whenever switching occurs (see the insets on Fig. 2). Fig. 3(a) illustrates the threshold shifts during the switching process. Pickett's model, which does not have an explicit threshold, can also account for the switching stochasticity through a similar parameter-dynamics scheme [Fig. 3(b)].

The examples considered here demonstrate a general approach to enhancing *any* deterministic behavioral memristor model: stochastic dynamics added to the model parameters will lead to smooth stochastic transitions of the internal state. Nonetheless, we notice that when the state transitions are fast enough on a given timescale any smooth stochastic behavioral model can be well approximated by instantaneous switching.

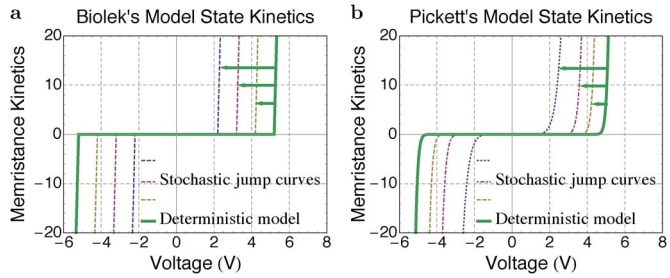

Fig. 3. Plots of the $f(w, V)$ function versus applied voltage for (a) Biolek's [45] and (b) Pickett's [43] models with different parameters. When the applied voltage is below the effective threshold, memristance does not change; however, once a switching event occurs, the effective threshold jumps to a lower value allowing the memristor to transit smoothly to a different state.

This allows to use the simple and efficient instantaneous stochastic switching, (3), in our network-level simulations.

III. Neuron Implementation

The stochastic spike response model (SRM) is a crucial building block for a number of probabilistic learning algorithms that can be implemented with networks of spiking neurons. In this section, we discuss the SRM neuron and match its behavior to the memristive stochasticity. Then, we present a simple memristive circuit that implements a neuromorphic soma and follows SRM. We provide simulation results of the spiking behavior of our circuit and discuss the key differences between our neuron and the alternative proposals.

A. Spike Response Model With Stochastic Firing Intensity

SRM is a generalization the classic I&F neuron [1]. A particular variation of this model (stochastic SRM_0), in which the response kernel is independent of the last spike time and the firing threshold is stochastic, was shown to be in a good agreement with the data recorded from pyramidal neurons of the somatosensory cortex *in vivo* [7].

At any time instant, such a neuron has the following instantaneous firing rate (also called stochastic intensity):

$$r(V - \theta) = \frac{1}{\tau_s} \exp\left(\frac{V - \theta}{\delta V}\right) \tag{7}$$

where θ is the effective threshold voltage, δV is the width of the spike emission zone, and τ_s is the mean time to spike at the threshold. Given that the stochastic firing rate is effectively the inverted time constant of the corresponding inhomogeneous Poisson process, we can rewrite (7) as

$$\tau(V - \theta) = \tau_s \exp\left(-\frac{V - \theta}{\delta V}\right). \tag{8}$$

Notice that (8) matches the memristive switching model (4) when $\tau_0 = \tau_s \exp(\theta/\delta V)$ and $V_0 = \delta V$. Based on this correspondence, we can use the spontaneous memristive switching events for triggering neural spike generation.

Importantly, the stochastic firing rate (7) exponentially depends on the membrane voltage and allows such spiking units to satisfy the neural computability condition of the Boltzmann distribution. This makes stochastic SRM, and hence the memristor-triggered spiking, suitable for implementing spiking

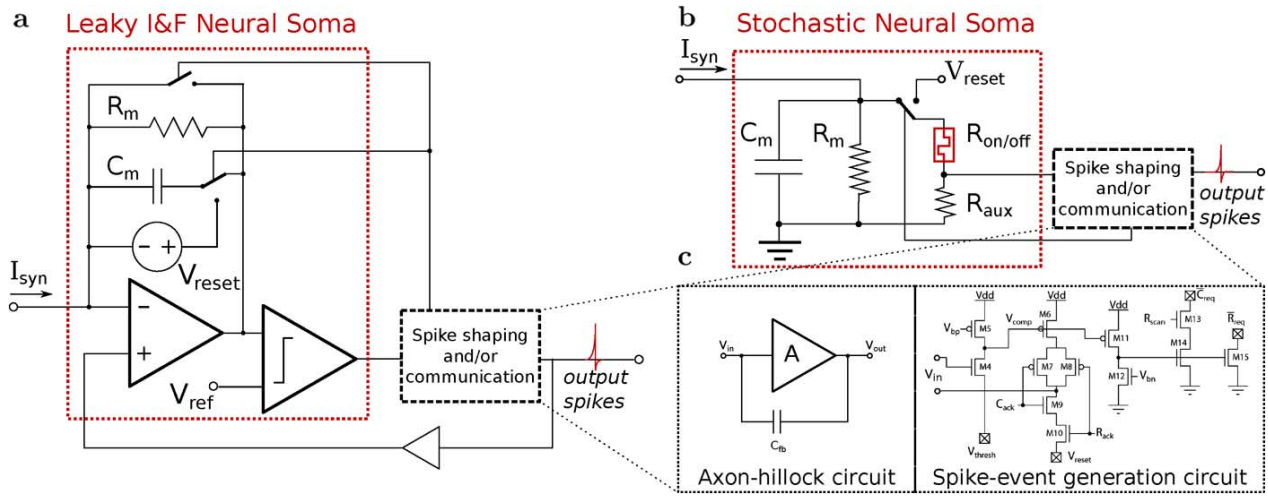

Fig. 4. Neural soma implementations. (a) The standard leaky I&F neural soma circuit connected to an analog or digital spike generator. (b) The proposed memristive stochastic SRM neural soma. (c) Examples of the spike shaping [15] and the spike-event generation [59] circuits which can be interfaced with I&F and stochastic SRM neurons. Using the address-event representation (AER) of spike-events [60], arbitrary STDP learning rules can be implemented in the address domain in an efficient and scalable manner.

Boltzmann machines and performing spike-based sampling and inference [9], [12], [61]. The relationship in (7) is also a key requirement for neural behavior necessary for building probabilistic winner-take-all (WTA) networks. Such neural networks can learn probabilistic multinomial mixture models of their input [8] and can be further used for building general probabilistic graphical models [10].

B. The Memristive Soma Circuit

The soma is a part of the neuron that collects the pre-synaptic input after it is received and integrated by the dendrites of the cell. In simplified point-neuron models, the dendrites and the soma are usually spatially merged. In leaky I&F neurons, the soma is implemented as an integrator of the cumulative synaptic input. It triggers the spike generation process once a certain membrane voltage, V_m, reaches a pre-defined threshold, $V_{\rm ref}$, and resets it back to a resting potential, $V_{\rm reset}$ [Fig. 4(a)].

Instead of comparing the voltages, we propose applying the membrane potential directly to a memristive device by connecting it in parallel to the membrane resistance, R_m [Fig. 4(b)]. Initially, the memristor should be in the high-resistance state, $R_{\rm off}$. Once it switches to the low-resistance state, $R_{\rm on}$, the current in the device experiences a jump, which can be converted into an analog spike of a specific shape [15], [16], or into a digital event [62] by a specific circuit [Fig. 4(c)]. After a spike occurs, the memristor is disconnected from the membrane part of the circuit and is reset back to $R_{\rm off}$ by applying a reversed potential, $V_{\rm reset}$. A refractory period, $\tau_{\rm ref}$, should be programmed through the switch that connects memristor to the membrane and the reset potential.

The voltage across the memristor (the membrane voltage) obeys the standard dynamics of a leaky integrator

$$\tau_m \frac{dV_m}{dt} = -V_m + RI_{\rm syn}(t) \tag{9}$$

TABLE II
STOCHASTIC NEURAL CIRCUIT PARAMETERS

C_m	R_m	$R_{\rm off}$	$R_{\rm on}$	$R_{\rm aux}$	$\Delta R_{\rm on/off}$
20 μF	1 kΩ	1 MΩ	10 kΩ	1 kΩ	5%

The membrane time constant was $\tau_m = RC_m \approx R_m C_m = 20$ ms.

where $\tau_m = RC_m$ is the membrane time constant and R is the total resistance of R_m connected in parallel to the memristor $R_{\rm on/off}$ and an auxiliary readout resistor $R_{\rm aux}$.

Notice that the switching probability,(4), depends on the membrane voltage, V_m, and hence on the synaptic input, $I_{\rm syn}$, through the (9). Since the memristor has a dynamic resistance,[1] it could potentially influence the switching probability, which is undesirable when we aim to implement a specific stochastic neuron model. Fortunately, we can have $R_m \ll R_{\rm off}$ (see Table II), which further makes the following approximation valid:

$$\frac{R_m(R_{\rm off}+R_{\rm aux})}{R_m+R_{\rm off}+R_{\rm aux}} = \frac{R_m}{1+R_m/(R_{\rm off}+R_{\rm aux})} \approx R_m. \tag{10}$$

This implies that the voltage across the memristor V_m is effectively independent of the resistance changes of the latter. Consequently, it follows that the switching probability, (4), exclusively depends on the synaptic input $I_{\rm syn}$.

To verify whether the behavior of the proposed soma circuit [Fig. 4(b)] matches the stochastic SRM, we performed simulations of a single neuron circuit in Python. We used Biolek's model, (6), with parameters that enabled practically instantaneous switching of the memristor (see Tables I and II for details). The reset switch and the feedback control were realized programmatically (i.e., simulated as ideal). A constant current, $I_{\rm syn}$, was injected into the neuron membrane, the system was simulated through a numerical integration with a time step of

[1]Not only is switching between LRS and HRS states possible, but also $R_{\rm on}$ and $R_{\rm off}$ values vary from cycle to cycle and follow some distribution [33].

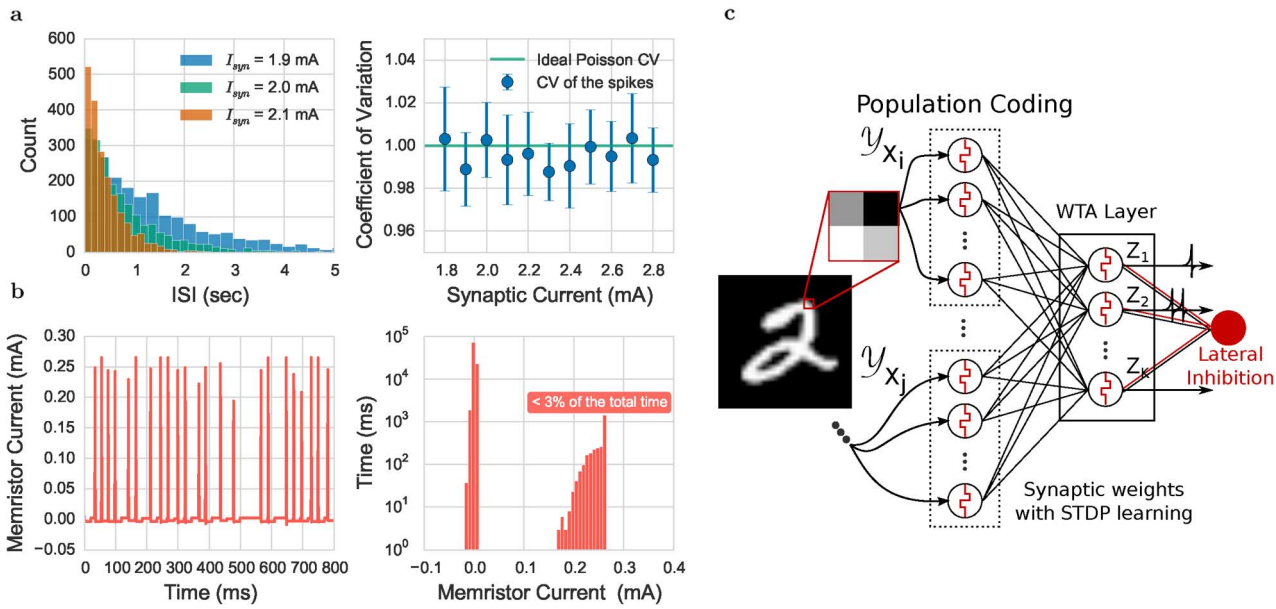

Fig. 5. Spiking behavior and a WTA network. (a) The ISI distributions for the proposed stochastic neuron for three different membrane voltages (left). The coefficient of variation of the ISI is close to 1, which implies that spiking statistics is close to the Poissonian (right). The refractory period was subtracted from the ISI times for these plots. (b) Current in the memristor versus time in single neuron simulations (left). The distribution of the current through the memristor in simulations is clearly separable even when $R_{\mathrm{on/off}}$ vary from cycle to cycle. (c) The WTA network has three layers: the input encoding layer, the WTA output layer, and the inhibitory layer.

0.1 ms, and the switching events along with the current through the memristor were recorded.

It follows from Fig. 5(a) that the spikes generated by our circuit obey the Poissonian statistics almost precisely: the inter-spike intervals (ISI) are distributed exponentially, and their coefficient of variation is close to 1 for different membrane voltages. Also, the ratio between the currents in the memristor before and after switching, I_{off} and I_{on}, respectively, is proportional to the ratio between R_{off} and R_{on}. Depending on fabrication technology, R_{off}-to-R_{on} ratio varies in the range $10^2 - 10^4$ making the switching events easy to detect. Even when R_{off} and R_{on} experience stochastic shifts from cycle to cycle, such high R_{off}-to-R_{on} ratio makes the soma resilient to possible output noise [Fig. 5(b)].

Finally, the power consumed by the soma can be estimated in two steps. Before the neuron spikes, the power dissipation is the same as for the standard leaky I&F unit

$$P_{\mathrm{normal}} = \frac{V_m^2}{\frac{R_m(R_{\mathrm{off}}+R_{\mathrm{aux}})}{R_m+R_{\mathrm{off}}+R_{\mathrm{aux}}}} \approx \frac{V_m^2}{R_m}. \tag{11}$$

After spiking, the neuron moves into a short refractory period, τ_{ref}, before the memristor transits back to the initial R_{off} state. This contributes to power dissipation as a refractory term

$$P_{\mathrm{ref}} = \frac{V_{\mathrm{reset}}^2}{(R_{\mathrm{on}} + R_{\mathrm{aux}})} \tag{12}$$

which is the dissipation on the memristor during its transition back to R_{off}. Since refractory period is short and spike trains are usually sparse [Fig. 5(b)], P_{ref} is negligible.

C. Comparison to the Alternative Proposals

There are several alternative approaches to implementing stochastically spiking neurons. A popular one that was studied theoretically in computational neuroscience [63] is based on injecting uncorrelated noise into the membrane of the I&F neuron or adding noise directly to the firing threshold [18]. Remarkably, in contrast with our memristive neuron, this method replicates the stochastic SRM behavior [7] only in the mean-field approximation, and it also has both the area and the power overheads for noise generation discussed in Section V below. In addition, the natively implemented SRM, in contrast to the I&F model, does not require the membrane voltage to be reset after spiking. Hence, there is no need to dissipate additional power on flushing the membrane capacitance, C_m.

Interestingly, memristive stochasticity has already been proposed to introduce variability in the spike trains generated by I&F neurons through random changes of the membrane time constant [39]. Even though this approach begins with a similar motivation, it leads to a different neuron model, namely, leaky I&F model with a time-dependent time constant [64], and hence is different from our proposal.

IV. Winner-Take-All Network Simulations

We verify that memristive neurons are suitable for probabilistic spike-based unsupervised learning through building and simulating a probabilistic WTA network, and training it on handwritten digits. Following [8], we first summarize the intuitions behind the stochastic learning and inference and give details on the WTA learning algorithm. Then, we present the simulation results and study the effect of different learning conditions, such as the input encoding schemes and possible imperfections in real memristors, on the eventual classification accuracy of the network.

A. Probabilistic Learning and Inference

The WTA architecture that can learn a probabilistic multinomial mixture model is presented in Fig. 5(c). It is composed of three layers: a layer of input encoding neurons connected in a feed-forward fashion to a WTA layer followed by an inhibitory layer; the latter is recursively connected to each WTA neuron.

The network takes a raster image as the input. Every component of the input signal (i.e., each image pixel in this case) is encoded with a small population of neurons which are either stochastically firing as Poisson processes with constant rates, or producing equidistant spikes at a constant frequency. That is, each possible value of input signal component X_i has a corresponding neuron. When X_i is equal to some value A_i, the corresponding neuron from the $\mathcal{Y}_{X_i}$ population has a nonzero average spiking rate while the rest of the population is silent. In our experiments, we used the binary versions of the images. Each pixel was represented by two Poisson spiking neurons that encoded 0 and 1 possible values of the pixel.

The third layer serves to impose inhibition on all the WTA neurons for some period of time σ_{inhib}. In the experiments, the inhibitory layer consisted of a single deterministic neuron with a low enough threshold to be triggered each time a spike is received. The function of inhibition is important for the network to operate in a balanced regime and for controlling the WTA sampling rate [8].

The spikes generated by the middle WTA layer are the output. Due to the inhibition, these neurons are competing: if one of the neurons emits a spike, the entire layer is inhibited preventing the others from firing. The membrane potential of a neuron Z_k from the WTA layer can be written as a sum of the synaptic input

$$u_k(t) = u_{k_0} + \sum_{i=1}^{N} w_{ki} y_i(t) \tag{13}$$

where u_{k_0} is the resting potential, w_{ki} is the synaptic weight between the neurons Y_i and Z_k, and $y_i(t)$ is the post-synaptic potential (PSP) induced by the pre-synaptic spikes from the encoding neuron, i. Given that WTA neurons behave as independent inhomogeneous Poisson processes with firing rates[2] $r_k = (1/\tau_0)\exp(u_k(t) - I(t))$, the probability of an output spike being actually generated by the neuron Z_k at a time point t takes the form of a softmax function

$$P\left[Z_k \text{ spiked at time } t\right] = \frac{\exp\left(u_k(t)\right)}{\sum_{l=1}^{K} \exp\left(u_l(t)\right)}. \tag{14}$$

The authors of [8] show that under a simple STDP rule the weights of such network will eventually converge to a state where $P[Z_k \text{ spiked at time } t]$ becomes equal to the posterior probability of a pattern learned by neuron Z_k from the data.

In other words, stochastically firing SRM neurons in combination with the simple STDP rule cause the WTA network to converge to a probabilistic multinomial mixture model learned from the input data stream. The algorithm realized by such network is called spike-based expectation-maximization (SEM). A rigorous mathematical proof of the convergence of the algorithm together with a comprehensive discussion of this result can be found in the original paper [8]. Importantly, the behavior of both encoding and competing neurons can be implemented with the proposed memristive circuit.

[2]Here, $I(t)$ is the recurrent signal coming form the inhibitory layer.

TABLE III
WTA NETWORK SIMULATION PARAMETERS

Parameter name	Notation	Value	Variation*
Time step	dt	0.01 ms	–
Encoding neurons			
Layer size	N	1568	–
Encoding rate	r_{enc}	40-400 Hz	–
Pattern length	τ_{enc}	40 ms	–
Pattern delay	τ_{delay}	10 ms	–
Population size	N_{enc}	2	–
WTA neurons			
Layer size	K	16-128	–
Memristor parameters	V_0	156 mV	0-20%
	τ_0	$2.85{\cdot}10^5$ s	0-60%
	R_{off}	1 MΩ	5%
	R_{on}	10 kΩ	5%
Refractory period	τ_{ref}	10 ms	–
Membrane time constant	τ_{mem}	20 ms	–
Inhibitory neurons			
Inhibition window	σ_{inhib}	10 ms	–
Inhibition weight	W_{inhib}	-100 V	–
Synapses			
PSP length	τ_{syn}	10 ms	–
STDP window	σ	10 ms	–
STDP multiplier	c	e^3	–
Weights (initial)	W	$[-0.5, 0.5]$	–

*The variation percentages characterize the parameter distributions across the neural populations, i.e., they virtually replicate a possible variation between different physical instances of the neuron circuit.

B. Experiments on MNIST Digits

We simulated the learning process of a number of WTA networks with different sizes of the output layer and 1568 neurons in the input encoding layer. In our simulations, we used spiking neurons based on the instantaneously switching stochastic memristor model, (3). The STDP learning algorithm was not done with memristors. The entire network was numerically integrated using the BRIAN simulator [65]—a popular tool in computational neuroscience known for its flexibility and ease of expressing and building arbitrary spiking neural models and networks. The simulation parameters are summarized in Table III.

The simulation results of a network with 32 output neurons are presented in Fig. 6. We see that the spiking output before learning was random and rather unstructured [Fig. 6(a)]. Further, the network was exposed to data that contained handwritten digits from 0 to 4: The input layer encoded each input pattern for 40 ms with a silent delay of 10 ms between the patterns. We fed the input images [Fig. 6(c)] in a randomly shuffled order. The learning process was simulated with a step of 0.01 ms.

After learning, a number of winning neurons became almost exclusively receptive to one of the patterns from the data which

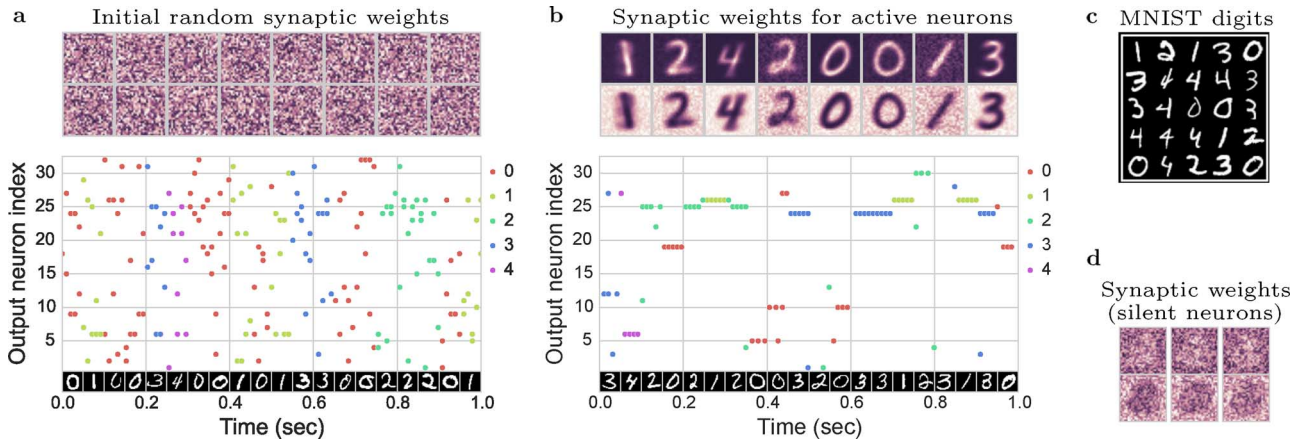

Fig. 6. Synaptic weights and output WTA spiking. (a), (b) The raster square images represent incoming synaptic weights for eight output neurons that became consistently sensitive to different patterns: (a) the weights before training, and (b) the weights after training. Each row consists of images that represent synapses that connect an output neuron to the input neurons that encode nonzero pixel values (upper row) and zero pixel values (bottom row). The scatter-plots show the output neural activity (a) before and (b) after training. Each point corresponds to a spike emitted by an output neuron. The spikes are color-coded according to the classes of input patterns. (c) Examples of the patterns from the training dataset. (d) Synaptic weights of the neurons that became silent after learning.

is evident from the post-learning behavior of the WTA as well as from the final synaptic matrices [Fig. 6(b)]. Several output neurons learned patterns of the same shapes and were spiking interchangeably when such patterns were presented to the network (e.g., zero-shapes learned by neurons 5 and 10). On the other hand, some shapes of the testing digits were rare in the training data (if presented at all), and hence none of the output neurons was active when such a pattern was shown [e.g., digit three that appears during 0.80–0.85 s, Fig. 6(b)]. Also, some WTA neurons adapted to narrow and particular types of patterns (e.g., neuron 19 adapted to right-inclined zeros only), while the others became sensitive to multiple shapes from the same class (e.g., neuron 24 was actively spiking on different forms of digit three).

Note that a small number of the output neurons did not manage to adapt to any pattern and hence became almost silent after training [their synaptic weights are presented in Fig. 6(d)]. The fact that some output neurons become silent after learning is known for multinomial mixture models: due to the stochastic nature of the EM optimization algorithm as well as random weight initialization, there is no guarantee of having eventually all the mixture components adapted to a specific pattern, especially for relatively large number of components [66]. To ameliorate this issue, various strategies of pruning low-weight components and adding new ones can be employed [67]. In our experiments, we did not allow the network architecture change during the learning process and hence did not explore such approaches.

Even though the WTA network represents a generative probabilistic model of the data and is trained in unsupervised fashion, it can be used as a classifier through the following scheme: Once the network is trained, we again stream the training data trough it and count the number of spikes for each of the classes, N_k^{class}, and the total number of spikes, N_k, emitted by each neuron Z_k. Further, we estimate the posterior probabilities of the data classes using the accumulated spiking statistics of the output neurons as follows:

$$P(\text{class}|Z_k) = \frac{N_k^{\text{class}}}{N_k}. \tag{15}$$

At classification time, we show the testing patterns, X_{test}, to the network, each for some time (50–100 ms), and predict the class label of the pattern according to the majority vote of the output neurons weighted by the precomputed posterior class probabilities, $P(\text{class}|Z_k)$. Again, we count the number of spikes, N_{ki}^{test}, emitted by each output neuron Z_k for a given testing pattern, $x_i \in X_{\text{test}}$, and make a class prediction as follows:

$$\text{prediction}(x_i) = \arg\max_{\text{class}} \sum_k P(\text{class}|Z_k) N_{ki}^{\text{test}}. \tag{16}$$

The classification accuracy can be computed on a held-out testing set in a standard manner as a ratio of the number of correctly predicted classes to the total number of test patterns

$$\text{Accuracy} = \frac{|\{x \in X_{\text{test}} : \text{prediction}(x) = \text{class}(x)\}|}{|\{x \in X_{\text{test}}\}|}. \tag{17}$$

We trained the WTA network with different sizes of the output layer (16, 32, 64, and 128 neurons) and tested its performance following the described procedure. The networks with larger number of output competing units demonstrated better performance in the classification task. This is because more neurons can capture more different patterns that underlie the data, as well as the majority voting procedure becomes more consistent for larger ensembles of voting units [68]. Our network with 128 output neurons achieved 78.4% of accuracy on the testing set, which is comparable with the results of the original paper [8]that reported 80.1% of accuracy.

Further, we conducted a number of robustness tests to analyze the factors that might affect the network performance. We evaluated the impact of 1) the variations in the memristor parameters and 2) the data encoding schemes and spike delivery failures. All the results are summarized in Table IV.

TABLE IV
WTA PERFORMANCE UNDER DIFFERENT CONDITIONS

Parameter name	Parameter value & classification accuracy			
Output layer size	16	32	64	128
Accuracy	58.9%	64.2%	73.9%	78.4%
Robustness to memristor imperfections (32 output neurons)				
Variability in τ_0	0%	20%	40%	60%
Accuracy	64.3%	64.2%	62.5%	63.0%
Variability in V_0	5%	10%	15%	20%
Accuracy	54.1%	42.0%	27.8%	16.6%
Robustness to encoding scheme (32 output neurons)				
Encoding scheme	Constant frequency		Poissonian constant ra	
Frequency/rate*	low	high	low	high
Accuracy	62.2%	64.1%	59.5%	64.3%
Robustness to synaptic failures (32 output neurons)				
Synaptic failure	5%	20%	35%	50%
Accuracy	64.2%	64.2%	62.0%	59.6%

*Low frequency/rate was 40 Hz and high frequency/rate was 400 Hz.

1) Robustness to Memristor Imperfections: The proposed stochastic neuron implementation relies on the memristive behavior. However, memristors are nano-scale devices and are prone to have fabrication-specific variations of their characteristics. In the previous section, we demonstrated that the high $R_{\rm off}$-to-$R_{\rm on}$ ratio makes our circuit immune to variations in the $R_{\rm off}$ and $R_{\rm on}$ values. Other parameters that can significantly affect the stochastic behavior of the neuron model are V_0 and τ_0. Even though they are fixed for a single memristor, they can vary across different devices, and hence potentially affect the performance of a spiking network.

We tested the robustness of the WTA network to variations in V_0 and τ_0. We found that differences in τ_0 across different output neurons do not affect the accuracy much, while the variations in V_0 lead to a significant deterioration of the performance (Table IV). Intuitively, this can be explained in the following way: V_0 accounts for the neuron sensitivity to the membrane voltage, while τ_0 only introduces a small additive bias. In other words, when the variation of V_0 is large, some WTA neurons become much more sensitive to *any* input pattern and always manage to suppress their competitors. This leads to a small number of active neurons in the output layer after training and eventually to poor performance. However, according to the experimental studies of the amorphous silicon memristors V_0 changes only by 35% if the width of the original 30 nm insulator film is doubled [58]. In other words, small 2%–5% error in the geometry of a memristor will probably cause a negligible variation in V_0.

2) Robustness to Encoding and Spike Delivery Failures: We tested the network's robustness to different input encoding schemes—population coding with constant frequency and with Poisson spiking—and found that its performance was stable (see Table IV; robustness to encoding scheme).

Also, neural architectures should be robust to potential spike delivery failures for them to be scalable. Thanks to probabilistic spiking, it is known that random failures do not affect the statistics of Poisson spike trains [62]: if the transmission of pre-synaptic Poisson spikes with rate r fails with probability $p_{\rm fail}$, the post-synaptic neuron sees incoming spikes as if they were generated by a Poisson neuron with a smaller rate, $r(1-p_{\rm fail})$. In other words, while frequent failures may slow down the overall learning, they should not affect the network performance. Our simulations confirmed resilience of the probabilistic WTA architecture with respect to such failures (see Table IV; robustness to synaptic failures).

C. Comparison With Other Winner-Take-All Implementations

In our simulations, we focused on the probabilistic WTA network that necessarily requires having probabilistically spiking neurons in the output layer. A related approach uses deterministic I&F neurons in the WTA layer [69]. The algorithm performed by such deterministic networks is similar to the online k-mean clustering [70], and it works with a different, rate-based input encoding scheme.

The major advantage of the stochastic WTA network is the ability to interpret its activity from the probabilistic perspective. Remarkably, it allows to combine such networks into complex structures and even implement arbitrary probabilistic Bayesian models [10].

V. DISCUSSION

The most popular alternative approach to implementing networks of stochastically spiking neurons is based on uncorrelated noise injection into each I&F unit. This requires supplying each neuron with an individualized noise source [13]. For this purpose, a multichannel, uncorrelated, pseudorandom bit stream generator based on a pair of linear feedback shift registers accompanied by a global clocking mechanism can be used [71]. Then, each neuron should be equipped with an XOR element, a low-pass filter, and a low-gain amplifier to convert the bit stream into an analog Gaussian noise signal [72]. In this case, not only the noise generation overhead makes scaling the system problematic, but also the additional circuitry occupies ample space on chip.

On the contrary, our proposed implementation exploits the intrinsic stochasticity of memristive switching, avoids noise generation expenses, and is extremely compact compared to the alternative approaches.[3] Currently, the only major drawback of the designs that exploit probabilistic switching of memristors is the lack of comprehensive experimental analysis and quantification of such behaviors for different materials and types of devices. Therefore, such experimental studies are highly relevant and important in future work.

Another possible approach to endowing deterministic neurons with effective stochastic behavior is based on exploiting network-generated variability [74]. However, this approach is less attractive due to the lack of a direct control over the stochastic properties of each neuron in such network (e.g., the voltage-dependent inhomogeneous Poissonian spiking rate cannot be straightforwardly mapped onto such network, and can be only approximated [19]). Moreover, it was shown that Poissonian statistics generated by a network itself is usually

[3]In the latest CMOS technologies, the reported die sizes of such additional elements are orders of magnitude larger than that of the memristor [73].

stable only for a limited input range if no additional noise generation mechanism is used [75].

VI. Conclusion

In this work, we demonstrated how the behavior of a stochastic SRM neuron can be implemented using memristors. First, we enhanced the general deterministic model of the memristor to account for abrupt spontaneous switching. According to the experimental evidence, such switching was approximated by an inhomogeneous Poisson process. We found that our model was capable of exhibiting deterministic behavior along with sharp spontaneous state jumps with a probability that depended on the applied voltage. We also showed how to account for the smooth transition kinetics of stochastic switching if necessary. Based on the model of this behavior, we proposed an implementation of a neuromorphic neural soma that integrated the synaptic input and triggered spike generation using memristive switching. The spiking behavior of the memristive neuron was akin to the stochastic firing intensity of the SRM. Finally, we implemented a probabilistic winner-take-all network and demonstrated with simulations that our simple stochastic neuron can be used for building spiking neural networks for probabilistic learning.

Acknowledgment

The authors would like to thank E. Neftci and M. Affan Zidan for helpful discussions and advice.

References

[1] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge, U.K.: Cambridge Univ. Press, 2002.

[2] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol.*, vol. 117, no. 4, p. 500, 1952.

[3] E. M. Izhikevich, *Dynamical Systems in Neuroscience*. Cambridge, MA: MIT Press, 2007.

[4] A. A. Faisal, L. P. Selen, and D. M. Wolpert, "Noise in the nervous system," *Nature Rev. Neurosci.*, vol. 9, no. 4, pp. 292–303, 2008.

[5] M. McDonnell and L. Ward, "The benefits of noise in neural systems: Bridging theory and experiment," *Nature Rev. Neurosci.*, vol. 12, no. 7, pp. 415–426, 2011.

[6] W. Maass, "Noise as a resource for computation and learning in networks of spiking neurons," *Proc. IEEE*, vol. 102, no. 5, pp. 860–880, May 2014.

[7] R. Jolivet, A. Rauch, H.-R. Lüscher, and W. Gerstner, "Predicting spike timing of neocortical pyramidal neurons by simple threshold models," *J. Comput. Neurosci.*, vol. 21, no. 1, pp. 35–49, 2006.

[8] B. Nessler, M. Pfeiffer, L. Buesing, and W. Maass, "Bayesian computation emerges in generic cortical microcircuits through spike-timing-dependent plasticity," *PLoS Comput. Biol.*, vol. 9, no. 4, p. e1003037, 2013.

[9] L. Buesing, J. Bill, B. Nessler, and W. Maass, "Neural dynamics as sampling: A model for stochastic computation in recurrent networks of spiking neurons," *PLoS Comput. Biol.*, vol. 7, no. 11, p. e1002211, 2011.

[10] D. Pecevski, L. Buesing, and W. Maass, "Probabilistic inference in general graphical models through sampling in stochastic networks of spiking neurons," *PLoS Comput. Biol.*, vol. 7, no. 12, p. e1002294, 2011.

[11] D. Kappel, B. Nessler, and W. Maass, "STDP installs in winner-take-all circuits an online approximation to hidden Markov model learning," *PLoS Comput. Biol.*, vol. 10, no. 3, p. e1003511, 2014.

[12] B. U. Pedroni, S. Das, E. Neftci, K. Kreutz-Delgado, and G. Cauwenberghs, "Neuromorphic adaptations of restricted Boltzmann machines and deep belief networks," in *Proc. Int.l Joint Conf. IEEE Neural Netw.*, 2013, pp. 1–6.

[13] E. Neftci, S. Das, B. Pedroni, K. Kreutz-Delgado, and G. Cauwenberghs, "Event-driven contrastive divergence for spiking neuromorphic systems," *Front. Neurosci.*, vol. 7, 2014.

[14] P. O'Connor, D. Neil, S.-C. Liu, T. Delbruck, and M. Pfeiffer, "Real-time classification and sensor fusion with a spiking deep belief network," *Front. Neurosci.*, vol. 7, 2013.

[15] C. Mead and M. Ismail, *Analog VLSI Implementation of Neural Systems*. New York: Springer, 1989.

[16] G. Indiveri *et al.*, "Neuromorphic silicon neuron circuits," *Front. Neurosci.*, vol. 5, 2011.

[17] H. E. Plesser and W. Gerstner, "Noise in integrate-and-fire neurons: From stochastic input to escape rates," *Neural Comput.*, vol. 12, no. 2, pp. 367–384, 2000.

[18] A. S. Cassidy *et al.*, "Cognitive computing building block: A versatile and efficient digital neuron model for neurosynaptic cores," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2013, pp. 1–10.

[19] M. A. Petrovici, J. Bill, I. Bytschok, J. Schemmel, and K. Meier, Stochastic inference with deterministic spiking neurons arXiv preprint arXiv:1311.3211, 2013.

[20] T. Hamilton, S. Afshar, A. van Schaik, and J. Tapson, "Stochastic electronics: A neuro-inspired design paradigm for integrated circuits," *Proc. IEEE*, vol. 102, no. 5, pp. 843–859, May 2014.

[21] J. J. Yang, D. B. Strukov, and D. R. Stewart, "Memristive devices for computing," *Nature Nanotechnol.*, vol. 8, no. 1, pp. 13–24, 2013.

[22] F. Alibart, E. Zamanidoost, and D. B. Strukov, "Pattern classification by memristive crossbar circuits using ex situ and in situ training," *Nature Commun.*, vol. 4, 2013.

[23] B. Linares-Barranco and T. Serrano-Gotarredona, "Memristance can explain spike-time-dependent-plasticity in neural synapses," *Nature Precedings*, pp. 1–4, 2009.

[24] S. H. Jo *et al.*, "Nanoscale memristor device as synapse in neuromorphic systems," *Nano Lett.*, vol. 4, pp. 1297–1301, 2010.

[25] C. Zamarre no-Ramos *et al.*, "On spike-timing-dependent-plasticity, memristive devices, building a self-learning visual cortex," *Front. Neurosci.*, vol. 5, 2011.

[26] T. Serrano-Gotarredona, T. Masquelier, and T. Prodromakis, "STDP and STDP variations with memristors for spiking neuromorphic learning systems," *Front. Neurosci.*, vol. 5, pp. 1–15, 2013.

[27] M. Rahimi Azghadi, N. Iannella, S. F. Al-Sarawi, G. Indiveri, and D. Abbott, "Spike-based synaptic plasticity in silicon: Design, implementation, application, challenges," *Proc. IEEE*, vol. 102, no. 5, pp. 717–737, May 2014.

[28] G. Indiveri, B. Linares-Barranco, R. Legenstein, G. Deligeorgis, and T. Prodromakis, "Integration of nanoscale memristor synapses in neuromorphic computing architectures," *Nanotechnology*, vol. 24, no. 38, p. 384010, 2013.

[29] T. Serrano-Gotarredona, T. Prodromakis, and B. Linares-Barranco, "A proposal for hybrid memristor-CMOS spiking neuromorphic learning systems," *Circuits Syst. Mag.*, vol. 13, pp. 74–88, 2013.

[30] S. Gaba, P. Sheridan, J. Zhou, S. Choi, and W. Lu, "Stochastic memristive devices for computing and neuromorphic applications," *Nanoscale*, vol. 5, no. 13, p. 5872, 2013.

[31] Q. Li, A. Khiat, I. Salaoru, H. Xu, and T. Prodromakis, "Stochastic switching of TiO2-based memristive devices with identical initial memory states," *Nanoscale Res. Lett.*, vol. 9, no. 1, pp. 1–5, 2014.

[32] Y. Yang *et al.*, "Observation of conducting filament growth in nanoscale resistive memories," *Nature Commun.*, vol. 3, p. 732, 2012.

[33] M. Suri *et al.*, "Bio-inspired stochastic computing using binary CBRAM synapses," *IEEE Trans. Electron Devices*, vol. 60, no. 7, pp. 2402–2409, Jul. 2013.

[34] S. Yu *et al.*, "Stochastic learning in oxide binary synaptic device for neuromorphic computing," *Front. Neurosci.*, vol. 7, 2013.

[35] J. Bill and R. Legenstein, "A compound memristive synapse model for statistical learning through STDP in spiking neural networks," *Front. Neurosci.*, vol. 8, 2014.

[36] S. Shin, K. Kim, and S.-M. S. Kang, "Memristor macromodel and its application to neuronal spike generation," in *IEEE Eur. Conf. Circuit Theory Design*, 2013, pp. 1–4.

[37] M. D. Pickett, G. Medeiros-Ribeiro, and R. S. Williams, "A scalable neuristor built with Mott memristors," *Nature Mater.*, vol. 12, no. 2, pp. 114–117, 2013.
[38] M. Hu, Y. Wang, Q. Qiu, Y. Chen, and H. Li, "The stochastic modeling of TiO2 memristor and its usage in neuromorphic system design," in *Proc. 9th Asia South Pacific Design Automat. Conf.*, Jan. 2014, pp. 831–836.
[39] G. Palma, M. Suri, D. Querlioz, E. Vianello, and B. De Salvo, "Stochastic neuron design using conductive bridge RAM," in *Proc. IEEE/ACM Int. Symp. IEEE Nanoscale Archit.*, 2013, pp. 95–100.
[40] Y. Yang and W. Lu, "Nanoscale resistive switching devices: Mechanisms and modeling," *Nanoscale*, vol. 5, no. 21, pp. 10 076–10 092, 2013.
[41] J. P. Strachan *et al.*, "Direct identification of the conducting channels in a functioning memristive device," *Adv. Mater.*, vol. 22, no. 32, pp. 3573–3577, 2010.
[42] M. A. Zidan *et al.*, "A family of memristor-based reactance-less oscillators," *Int. J. Circuit Theory Appl.*, 2013.
[43] M. D. Pickett *et al.*, "Switching dynamics in titanium dioxide memristive devices," *J. Appl. Phys.*, vol. 106, no. 7, p. 074508, 2009.
[44] Z. Biolek, D. Biolek, and V. Biolkova, "Spice model of memristor with nonlinear dopant drift," *Radioengineering*, vol. 18, no. 2, 2009.
[45] D. Biolek, M. Di Ventra, and Y. V. Pershin, Reliable spice simulations of memristors, memcapacitors and meminductors arXiv preprint arXiv:1307.2717, 2013.
[46] T. Prodromakis, B. P. Peh, C. Papavassiliou, and C. Toumazou, "A versatile memristor model with nonlinear dopant kinetics," *IEEE Trans. Electron Devices*, vol. 58, no. 9, pp. 3099–3105, Sep. 2011.
[47] F. Corinto and A. Ascoli, "A boundary condition-based approach to the modeling of memristor nanostructures," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 11, pp. 2713–2726, Nov. 2012.
[48] A. Ascoli, F. Corinto, V. Senger, and R. Tetzlaff, "Memristor model comparison," *IEEE Circuits Syst. Mag.*, vol. 13, no. 2, 2013.
[49] A. Radwan, M. A. Zidan, and K. Salama, "On the mathematical modeling of memristors," in *Proc. IEEE Int. Conf. Microelectron.*, 2010, pp. 284–287.
[50] F. Garcia, M. Lopez-Vallejo, and P. Ituero, "Building memristor applications: From device model to circuit design," *IEEE Trans. Nanotechnol.*, vol. 13, no. 6, pp. 1154–1162, Nov. 2014.
[51] A. G. Radwan, M. A. Zidan, and K. Salama, "HP memristor mathematical model for periodic signals and DC," in *Proc. 53rd IEEE Int. Midwest Symp. Circuits Syst.*, 2010, pp. 861–864.
[52] O. Kavehei *et al.*, "Fabrication and modeling of ag/tio2/ito memristor," in *Proc. 54th IEEE Int. Midwest Sympo. Circuits Syst.*, 2011, pp. 1–4.
[53] S. Kim, S. Choi, and W. Lu, "Comprehensive physical model of dynamic resistive switching in an oxide memristor," *ACS Nano*, vol. 8, 2014.
[54] Q. Lv *et al.*, "Conducting nanofilaments formed by oxygen vacancy migration in Ti/TiO2/TiN/MgO memristive device," *J. Appl. Phys.*, vol. 110, no. 10, p. 104511, 2011.
[55] S. Yu, X. Guan, and H.-S. Wong, "On the stochastic nature of resistive switching in metal oxide RRAM: Physical modeling, Monte Carlo simulation, experimental characterization," in *Proc. IEEE Int. Electron Devices Meet.*, 2011, pp. 17–3.
[56] D. Ielmini, "Modeling the universal set/reset characteristics of bipolar RRAM by field-and temperature-driven filament growth," *IEEE Trans. Electron Devices*, vol. 58, no. 12, pp. 4309–4317, Dec. 2011.
[57] D. R. Cox and V. Isham, *Point Processes*. Boca Raton, FL: CRC Press, 1980, vol. 12.
[58] S. H. Jo, K.-H. Kim, and W. Lu, "Programmable resistance switching in nanoscale two-terminal devices," *Nano Lett.*, vol. 9, no. 1, 2008.
[59] R. J. Vogelstein, U. Mallik, J. T. Vogelstein, and G. Cauwenberghs, "Dynamically reconfigurable silicon array of spiking neurons with conductance-based synapses," *IEEE Trans. Neural Netw.*, vol. 18, no. 1, pp. 253–265, Jan. 2007.
[60] R. J. Vogelstein *et al.*, "Spike timing-dependent plasticity in the address domain," *Adv. Neural Inf. Process. Syst.*, 2002.
[61] M. Al-Shedivat, R. Naous, E. Neftci, G. Cauwenberghs, and K. N. Salama, "Inherently stochastic spiking neurons for probabilistic neural computation," in *Proc. 7th Int. IEEE/EMBS Conf. IEEE Neural Eng.*, 2013.
[62] D. H. Goldberg, G. Cauwenberghs, and A. G. Andreou, "Probabilistic synaptic weighting in a reconfigurable network of VLSI integrate-and-fire neurons," *Neural Netw.*, vol. 14, no. 6, pp. 781–793, 2001.
[63] A. Renart, N. Brunel, and X.-J. Wang, "Mean-field theory of irregularly spiking neuronal populations and working memory in recurrent cortical networks," *Comput. Neurosci., Comprehensive Approach*, pp. 431–490, 2004.
[64] W. Gerstner, "Spike-response model," *Scholarpedia*, vol. 3, no. 12, 2008.
[65] M. Stimberg, D. F. Goodman, V. Benichoux, and R. Brette, "Equation-oriented specification of neural models for simulations," *Front. Neuroinformat.*, vol. 8, 2014.
[66] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J. R. Stat. Soc. Ser. B (Methodol.)*, pp. 1–38, 1977.
[67] D. Lowd and P. Domingos, "Naive Bayes models for probability estimation," in *Proc. 22nd ACM Int. Conf. Mach. Learn.*, 2005, pp. 529–536.
[68] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*. New York: Springer, 2000, pp. 1–15.
[69] D. Querlioz, O. Bichler, and C. Gamrat, "Simulation of a memristor-based spiking neural network immune to device variations," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2011, pp. 1775–1781.
[70] S. Zhong, "Efficient online spherical k-means clustering," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2005, vol. 5, pp. 3180–3185.
[71] G. Cauwenberghs, "An analog VLSI recurrent neural network learning a continuous-time trajectory," *IEEE Trans. Neural Netw.*, vol. 7, no. 2, pp. 346–361, Mar. 1996.
[72] M. B. Parker and R. Chu, "A VLSI-efficient technique for generating multiple uncorrelated noise sources and its application to stochastic neural networks," *IEEE Trans. Circuits Syst.*, vol. 38, no. 1, p. 109, Jan. 1991.
[73] J. A. Montiel-Nelson, V. Navarro, J. Sosa, and T. Bautista, "Analysis and optimization of dynamically reconfigurable regenerative comparators for ultra-low power 6-bit TC-ADCs in 90 nm CMOS technologies," *Microelectronics*, vol. 45, pp. 1247–1253, 2014.
[74] E. Chicca and S. Fusi, "Stochastic synaptic plasticity in deterministic aVLSI networks of spiking neurons," *Proc. World Congr. Neuroinformat.*, 2001.
[75] R. Moreno-Bote, "Poisson-like spiking in circuits with probabilistic synapses," *PLoS Comput. Biol.*, vol. 10, no. 7, p. e1003522, 2014.

Maruan Al-Shedivat (S'14) received the B.Sc. degree (with honors) in physics from Lomonosov Moscow State University, Moscow, Russia, the M.Eng. degree in data analysis from Yandex Data School, Moscow Institute of Physics and Technology, Moscow, Russia, in 2013, and the M.Sc. degree in computer science from King Abdullah University of Science and Technology, Saudi Arabia, in 2015, where he is currently a graduate student.

His research interests are in machine learning, probabilistic models, transfer learning, and stochastic neural networks.

Rawan Naous (M'05) received the B.E. degree in electrical engineering from the American University of Beirut, Beirut, Lebanon, in 2005, and the M.Sc. degree in communications electronics (with honors) from the Technical University of Munich, Munich, Germany, in 2007. She is currently working toward the Ph.D. degree in electrical engineering at King Abdullah University of Science and Technology, Saudi Arabia.

Her research interests include mixed analog–digital IC design, memristors, stochastic electronics, nonvolatile memory, neuromorphic system architectures, and embedded machine learning.

Gert Cauwenberghs (S'89–M'94–SM'04–F'11) received the Ph.D. degree in electrical engineering from California Institute of Technology, Pasadena, CA, USA, in 1994.

He is Professor of Bioengineering and Co-Director of the Institute for Neural Computation at the University of California–San Diego, La Jolla, CA, USA. He was previously Professor of Electrical and Computer Engineering at Johns Hopkins University, Baltimore, MD, USA, and Visiting Professor of Brain and Cognitive Science at Massachusetts Institute of Technology, Cambridge, MA, USA. He co-founded Cognionics Inc., and chairs its Scientific Advisory Board. His research focuses on micropower biomedical instrumentation, neuron–silicon and brain–machine interfaces, neuromorphic engineering, and adaptive intelligent systems.

Dr. Cauwenberghs received the NSF Career Award in 1997, ONR Young Investigator Award in 1999, and Presidential Early Career Award for Scientists and Engineers in 2000. He serves IEEE in a variety of roles including as General Chair of the IEEE Biomedical Circuits and Systems Conference (BioCAS 2011, San Diego, CA, USA), as Program Chair of the IEEE Engineering in Medicine and Biology Conference (EMBC 2012, San Diego, CA, USA), and as Editor-in-Chief of the IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS.

Khaled Nabil Salama (S'97–M'05–SM'10) received the B.S. degree (with honors) from the Department Electronics and Communications, Cairo University, Cairo, Egypt, in 1997, and the M.S. and Ph.D. degrees from the Department of Electrical Engineering, Stanford University, Stanford, CA, USA, in 2000 and 2005, respectively.

He was an Asistant Professor at Rensselaer Polytechnic Institute, NY, USA, between 2005 and 2009. He joined King Abdullah University of Science and Technology in January 2009 and was the founding Program Chair until August 2011. His work on CMOS sensors for molecular detection has been funded by the National Institutes of Health (NIH) and the Defense Advanced Research Projects Agency (DARPA), awarded the Stanford–Berkeley Innovators Challenge Award in biological sciences and was acquired by Lumina Inc. He is the author of 100 papers and eight patents on low-power mixed-signal circuits for intelligent fully integrated sensors and nonlinear electronics specially memristor devices.