Programmable Fading Memory in Atomic Switch Systems for Error Checking Applications

Renato Aguilera, Henry O. Sillin, Adam Z. Stieg, and James K. Gimzewski

Abstract Disruptive technology in computational devices is required as the universal computing machines approach quantum mechanical limits. Integration of state-of-the-art memristive devices provides optimal scaling of current technologies beyond this limit through the adoption of neuromorphic models. Universal computing machines pioneered by Alan Turing are strictly based on top-down intelligent design. Neuromorphic models instead engage in bottom-up programmability by emulating mammalian brain design and characteristics. Here we show the design, characterization, and implementation of a massively parallel memristor neuromorphic network based on metal chalcogenide atomic switch network (ASN) systems with key characteristics such as short- and long-term potentiation, power-law dynamics, and scale-free topology.

1 Introduction

Fundamental work by Carver Mead and colleagues (Mead 1990) in the development of the concept neuromorphic technology enabled a disruptive paradigm shift in computing technologies. Unlike other conceptions of machine learning, neuromorphic computing attempted to completely emulate neuron functionality within a physically realizable computing hardware. In doing so, the power-efficiency and complexity of neurons can be harnessed without bottlenecking data as in CMOS technology (Backus 1978). Additionally, neuron clusters in the brain can recognize patterns and are capable of performing unconventional computing similar to Alan

R. Aguilera · H. O. Sillin · J. K. Gimzewski (✉)
Department of Chemistry and Biochemistry, University of California, Los Angeles 607 Charles E. Young Drive East, Los Angeles, CA 90095, USA
e-mail: gimzewski@cnsi.ucla.edu

A. Z. Stieg · J. K. Gimzewski
California NanoSystems Institute, 570 Westwood Plaza, Los Angeles, CA 90095, USA

National Institute for Materials Science (NIMS), WPI Center for Materials Nanoarchitectonics (MANA), 1-1 Namiki, Tsukuba, Ibaraki 305-0044, Japan

Turing's B-machine (Turing 1950). The evolutionary optimization of the brain in both structure and functionality provides an exciting new zeitgeist in a fast stalling technology (Waldrop 2016).

Current works using learning dedicated computer hardware provide possible throttling pass the information bottleneck (Husband et al. 2003; Tour et al. 2003). Additionally, developments of beyond-CMOS devices such as magnetic tunneling junctions (Furuta et al. 2018), photoelectronic (Hermans et al. 2015), and memristors (Ascoli et al. 2017; Du et al. 2017; Strukov et al. 2008) explored computing architectures outside of typical transistor-based models. In 2000, the International Center for Materials Nanoarchitectonics (MANA) commenced investigation of viable neuromorphic materials utilizing nanowire mechanisms and constraints for material design. Specifically, Professor Masakazu Aono developed the atomic switch as a nano-equivalent neuron operating under quantum mechanical limits at GHz speeds (Tsuchiya et al. 2015; Tsuruoka et al. 2017). Aono's work on atomic switches developed the underlying principles for integration and development of nanowires for neuromorphic computing elucidating nanowire plasticity and memory capabilities. Single transistor-like atomic switches were introduced into memory storage devices by Nippon Electric Company (NEC 2009) using non-dynamic Cu-TiO$_2$/TaSiO atomic switches for 32 nm CMOS technology. However, the plasticity present in neurological functions is inherently non-static and dynamic (Büsing et al. 2010; Lukoševičius and Jaeger 2009). Further development of scalable neuromorphic atomic switch devices required a holistic nanoarchitectonic design incorporating dynamic and nonlinear network behaviors.

An emerging mathematical model developed by Leon Chua sought to integrate CMOS technology with nonlinear and chaotic systems (Chua 1980, 1988). In 1971, Chua theorized that, in addition to the 3 fundamental elements in the lumped element circuit model, there was a 4th element he called the *memristor*. Complementing the relations provided by the resistor, capacitor, and inductor, the memristor was able to relate the magnetic flux with electric charge. In order for circuit theory to utilize this 4th element, the model required adaptation of a purely nonlinear circuit theory more akin to Turing's B-machine. Particularly, the model emphasized harnessing emerging behaviors due to coupled circuits similar to the ideals of neuromorphic computing. Here, we present the fabrication of a physical random neural network via growth of memristive atomic switch networks, harnessing phenomenological fractal growth to directly imitate neural evolution for neuromorphic computing.

Unlike conventional neuromorphic platforms which require meticulous design, atomic switch networks (ASN) produced by interconnected nanowires introduce a unique methodology of controlled evolution. Meticulous design of a system required a complete model of understanding such as the CMOS computer architecture modeled by the universal Turing machine. However, it was more practical to develop a scheme of top-down adaptation than by bottom-up selective modification toward the desired function due to hardware limitations. A combined effort of theoretical predictions and experimental verification is presented here to design a methodology that was physically practical in implementing a computation referred to as *reservoir computing*. A physically realizable recurrent network comprised of

gapless-type atomic switches with Ag_2S as the active material previously demonstrated device tenability as functionally compatible neurons (Demis et al. 2015, 2016; Sillin et al. 2013).

An analysis of atomic switch networks at the interface of theory and experiment was accomplished by implementing theoretical paradigms of computation from the perspective of experimental feasibility. Considerations of physical practicality and CMOS compatibility were given priority over an ideal model with microscopic details. Reservoir computing was implemented on the proposed device to accomplish a series of error checking parity tasks and activation control to assess its computational capability.

We proposed to answer the following questions:

1. What is the relationship between the dynamical properties of a random system and its computational capability as a reservoir?
2. How do these dynamical properties emerge in macroscopic tools available to an experimentalist?

1.1 The Atomic Switch

Atomic switches were a class of devices that enabled the use of quantum tunneling for signal transduction. The first experiment to measure the transition from an electron quantum tunneling to single point contact regime was reported in 1987 using a scanning tunneling microscope (STM) in ultra-high-vacuum (UHV) on a silver surface (Möller 1987). Current-distance characteristics showed an abrupt increase in conductance, $G \sim \frac{e^2}{2h} \approx \frac{1}{13}$ kΩ, at sufficiently small tip-surface gaps, establishing the quantized unit of conductance. Subsequent theoretical analysis verified that at small gap distance the effective tunnel barrier collapses prior to point contact via ballistic electron injection (Lang 1986). Later work demonstrated further jumps of $G \sim n\frac{2e^2}{h}$, where $n = 1, 2, 3\ldots$, in the conductance occur as the contact area is increased. Such observations were not limited to STM experiments; even two macroscopic wires brought in contact also displayed this effect, albeit in a less controlled manner. Houten *et al.* provide an excellent review of quantized conduction, which also introduces Landauer's concept of transmission $G = \frac{2e^2}{h} \sum_n t_n$, where the term t is the transmission (van Houten et al. 1996).

In 2002, experiments by Terabe et al. found that Ag atoms could be transported through a STM tip made of silver coated with silver sulfide and deposited on a surface in a controlled manner (Terabe et al. 2002). The characteristics of this process also occurred via quantized conduction, however, the mechanism involved ion migration under the influence of an electric field, a process called "electroionics", meaning that in addition to electron motion, ion motion also occurs simultaneously. Normally, ionic diffusion processes on the macro-scale are considered to be slow, but when they are induced on the nanometer scale, they are actually quite fast and can occur on a (sub-) nanosecond scale depending on the geometry and dimensions of the junction.

In 2005, using junctions fabricated using conventional microelectronics, Terabe et al. demonstrated atomic switching in silver sulfide junctions with discrete and reversible quantized jumps from n = 1 to 10. This was the birth of the "atomic switch". Since that date, quantized conduction has been observed by a number of researchers in a wide range of materials including sulfide junctions of copper, tungsten sub-oxides as well as various metal-doped polymers.

Aside from the fundamental science of their quantization, the interesting electronic features of atomic switches were hysteresis, on/off conduction ratio, switching speed and volatility characteristics as well as CMOS compatibility because of their potential in digital electronic memory applications. Indeed, NEC recently has incorporated atomic switch technology into field programmable gate arrays (FPGAs) (Aramaki et al. 1991; NEC 2009) where a reduction in device footprint, speed, and energy consumption was achieved by replacing certain memory tasks, normally using transistors, into the circuitry.

Additional atomic switch functionality was reported in 2011 when studying switching near-threshold conditions (Ohno et al. 2011; Hasegawa et al. 2011). It was found that atomic switches have an on-off memorization property of past switching events. For instance if switching is performed infrequently, the switches remain in the on-state only briefly whereas if frequent switching events are made in rapid succession then the on-state persists for a longer time. A series of careful experiments were able to relate these physical observations to a psychological model of learning called the Akinson-Schriffin multi-store model (Atkinson and Shiffrin 1968). The essence of the model involves sensory memory (SM), short-term memory (STM), and long-term memory (LTM). New information arrives in the brain as sensory memory and that information was passed to short-term memory. In the absence of similar information it was forgotten. However, if the process was repeated many times, the information was moved into long-term memory. In terms of bio-inspiration, the operational characteristics of the atomic switch under threshold switching were also related to characteristics of biological synapses. The atomic switch therefore has also been called a synthetic synapse where memory was represented by conduction state.

The next step in creating a "brain inspired" device was the fabrication of networks of synthetic synapses (Atomic Switches). Taking the neocortex as a biologically inspiration, self-assembly was used to incorporate atomic switches into a dense dendritic tangle of silver nanowires resulting in a density of ~10^8 connections/cm^2 (Avizienis et al. 2013). In response to electrical inputs, which inject energy into the network, these networks exhibited self-organization and critical power-law dynamics and spatio-temporal nonlinear outputs at multiple electrodes.

1.2   Neuromorphic Atomic Switch Networks

The clear desire for neuromorphic architectures has led to further investigations and developments of different synthetic synapse models. Establishing specific connections between patterns of electrical activity and brain function was a difficult task

that requires studying general features of neuronal structure in order to determine the essential properties required to construct a device capable of learning in a physical sense. These features are believed to include at least synaptic plasticity, allowing physical reconfiguration of the network to enable functional differentiation and the development of hierarchical structures in which all possess correlated memory distributed throughout the dynamically coupled synapses. Therefore, learning and computational capacity in the brain are connected to dynamic activity and functional connectivity. Specifically, a near-critical or "edge-of-chaos" operational (Langton 1990) regime has been associated with the fast, correlated response to stimulation necessary for computation and learning. Developing computational machinery whose operation results from intrinsic critical dynamics was a complex task; with Atomic Switch Network (ASN) devices demonstrated such functionality (Fig. 1).

Utilizing the theoretical concepts presented in the previous section, we designed a neuromorphic device by incorporating atomic switches in a highly recursive interconnected network. The work of Aono (Terabe et al. 2002; Ohno et al. 2011; Hasegawa et al. 2011) established the fundamental principles and design of atomic switches. Observation of plasticity and information retention in atomic switches enabled us to successfully implement them in neuromorphic hardware for reservoir computing (Demis et al. 2016; Sillin et al. 2013). A number of materials were available and various functionalities may be tuned depending on the active material. Here, metal-chalcogenides were chosen due to their ready integration into CMOS technology and capability for spontaneous fractal growth.

— platinum wires ● SU-8 holes ■ copper posts
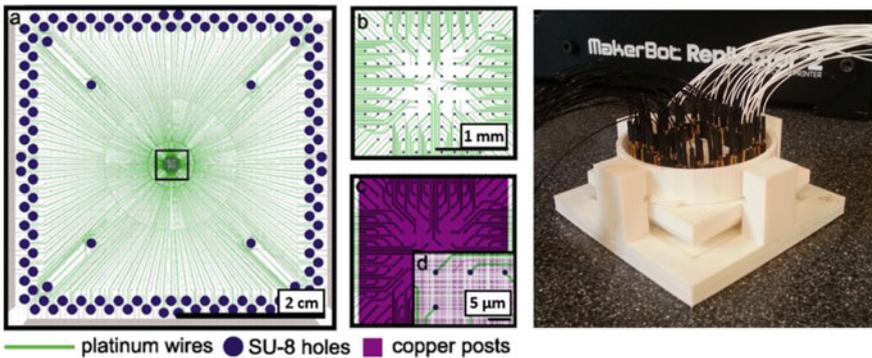
Fig. 1 Network diagram and analogue interface. A circuit schematic is shown in **a** showing the platinum circuit (green wires) and readout PC interface (blue circles). The silver network was placed in the central region (boxed) and a closer inspection of the region wiring is shown in **b**. Pre-patterned posts were lithographically placed within the boxed region in **c** ready for network growth. The device was interfaced to a National Instruments PXI—e using a custom device housing

2 Theoretical Constraints and Consideration

The concept of neuromorphic hardware as conceived by Mead (1990) intended to emulate the problem-solving capability of biology, which has been evolutionary optimized by nature. Observations of DNA folding and editing (Romero et al. 2017) demonstrated the capabilities of evolutionary algorithms to enable DNA to execute complex protein interaction and regulation. Natural selection has inherently optimized these systems in their task-specific function, thereby minimizing energy, maximizing information transfer, and encoding fault-tolerant and adaptive behaviors (Kim and Han 2002). Neuromorphic hardware attempted to adapt this architectural design within the context of circuit theory and analysis. A premier model for computational design was the human–brain, where complex computation such as speech, multisensory control, and chaotic predictions was commonly executed while operating under relatively simple rules.

Fundamentally, the brain utilized synaptically interconnected neurons to transfer and process information. Each neuron operated under the Hebbian fire-diffuse-fire principal (Hebb 1950; Timofeeva and Coombes 2003) which activated neurons with a sigmoidal function profile similar to transistors with voltage replacing ions in the latter case. Unlike contemporary digital transistors, each neuron was heavily coupled to other elements behaving effectively as a history-dependent nonlinear device. Circuit network theory has analogous examples of coupled inductive circuits communicating across devices, but such circuits were typically designed to eliminate coupled cross-talk and the overall circuit was linearized in its functionality. However, simplification of these interactions invariably destroys emergent behaviors observed in complex systems (Chen and Wang 2003; Goudarzi et al. 2012), which was capable of accomplishing complex computation. Nonlinear circuit design and analytical models by Chua (Ascoli et al. 2017; Chua and Wu 1995) attempted to utilize these complex interactions but have limited integration within information technology. Instead, machine learning algorithms were implemented in software which mimics the design and learning rules of biological systems. Here, a combination of machine learning architecture and nonlinear circuit design is briefly presented and discussion was restricted within a feedforward network for brevity; however, a formal and comprehensive discussion has been previously published (Lukoševičius and Jaeger 2009; Verstraeten et al. 2009).

The neural network machine learning paradigm traditionally attempted to achieve learning by modification of network topology and connectivity via adjustments to neuronal coupling strength. The general architecture of a neural network was designed similar to the human–brain—a collection of nodes or neurons interconnected with synapses to other neurons in a hierarchal layered structure (Graves et al. 2013; Krizhevsky et al. 2012; Abraham 2005; Schrauwen 2007; Hassoun 1995; Hopfield 1988). Neuronal nodes were typically designed to integrate incoming signals and transform them using a sigmoidal transfer function. The integration was the weighted sum of all signals received by the neuron from a predetermined set of input neurons from the previous layer:

$$\vec{I}^{0}(t) = \hat{N}_0 \vec{I}^{D}(t) \equiv \vec{f}^{0}\left(\vec{I}^{D}(t)\right), \tag{1}$$

where $\vec{I}^{0}(t)$ was a vector with elements of the transduced signal of each neuron in layer 0, $\hat{N}_0$ was a matrix representing the transformation of the driving signal, $\vec{I}^{D}(t)$ the driving signal, and $\vec{f}^{0}$ was a vector whose elements are the transfer functions of each neuron. The neuron activated according to its designed transfer function, then propagated the signal to its pre-designated output neurons in the next layer, i.e., layer 1:

$$\vec{I}^{1}(t) = \hat{N}_1 \vec{I}^{0}(t) \equiv \vec{f}^{1}\left(\hat{w}_1 \cdot \vec{I}^{D}(t)\right) \tag{2}$$

Here, $\vec{w}_i^{1}$ was the coupling strength between neuron i in layer 1 and all other neurons. This process was repeated from neuron to neuron in a hierarchal structure composed of layers or networks of neurons until it arrives at an output neuron layer, represented by $\vec{I}^{T}(t)$, where the user observes and process the final signal $I^{F}(t)$:

$$\vec{I}^{T}(t) = \prod_{k}^{\forall} \hat{N}_k \vec{I}^{D}(t) \equiv \hat{O}\vec{I}^{D}(t) \tag{3}$$

$$I^{F}(t) = \hat{w}_D \cdot \vec{I}^{T}(t). \tag{4}$$

The operator $\hat{O}$ represented the overall transformation by the network, and $\hat{w}_D$ was the design matrix in the output layer whose rows were the number of observable parameters and columns were the coupling strength to the sensors. While the above assumed a feedforward architecture, Eq. (3) and onwards may be generalized for any network if one allows $\hat{O}$ to represent any network transformation. Learning was achieved by designing the network connections in such a way that the output signal was transformed into a desired target signal. Each desired computational process corresponded to a desired signal, $I_{\text{desired}}^{\text{test}}$, for a given input signal, $I^{D}(t)$, and network transformation, $I^{F}(t)$. The synaptic strength of individual connections was adjusted in incremental corrective steps according to a learning rule using a training dataset until the network's effective function was the desired mathematical operation. Various learning techniques exist and depend on network type, connectivity structure, neuronal transfer function, I/O implementation, task complexity, and computational constraints (Büsing et al. 2010; Haimovici et al. 2013; Nedaaee Oskoee and Sahimi 2011; Sporns 2006). Here, we focused on the linear-regression learning rule as it was the typical and simplest learning rule:

$$\hat{w}_D \hat{O} = \left(\left[\vec{I}^T(t) \right]^\dagger \left[\vec{I}^T(t) \right] \right)^{-1} \left[\vec{I}^T(t) \right]^\dagger I_{\text{desired}}^{\text{training}}(t). \tag{5}$$

The process was done recursively by using a number of controlled training datasets to determine error propagation and correction. Defining a metric for error was nontrivial and clever designs and calculations of error existed that can drastically determine learning performance. However, we focused on the most commonly used and simplest definition of signal error which was the normalized mean squared error (NMSE) and adopted accuracy as a more intuitive measure of performance.

$$error \equiv \frac{E\left[\left(I_{\text{desired}}^{\text{test}} - I^F(t) \right)^2 \right]}{E\left[\left(I_{\text{desired}}^{\text{test}} - E\left[I_{\text{desired}}^{\text{test}} \right] \right)^2 \right]}; \; accuracy \equiv 1 - error \tag{6}$$

A network's computational capability was nearly defined by its network size, size of $\hat{O}$, while simultaneously increasing the complexity in learning. Unfortunately, implementing such a model using traditional photolithography manufacturing inevitably approached the Abbe diffraction limit (Abbe 1873), which was incapable of physically addressing elements on similar scale as current software implemented neural networks. Reservoir computing (Schrauwen et al. 2007; Verstraeten et al. 2009) was a distinguished computational model for scalable neuromorphic hardware as it does not require comprehensive control of the network, omitting the $\hat{O}$ in (5). Learning algorithms only required training on the output layer of neurons, $\hat{w}_D$, while the inner "reservoir" neurons, $\hat{O}$, are unattended and replaced $\vec{I}^T(t)$ with $I^F(t)$ in (5) (Lukoševičius and Jaeger 2009; Schrauwen et al. 2007; Verstraeten et al. 2009). Here, we utilized the reservoir computing paradigm as the functional model for computation in our ASN experiments.

2.1 Nonlinear Circuits

Regardless of the network construction or stimulation, a neural network was not capable of performing complex calculations if individual neurons behaved linearly (Carbajal et al. 2015). A brief proof of the desire for nonlinearity was by *contra positive* and to logically investigate the behavior of a network with purely linear elements. We constrained discussion using the above mathematical formalism where systems were represented only by neurons, and any post-processing or contributions from instrumentation were represented as an appropriate neuron layer. Let individual neurons be defined by the gain of an op-amp circuit to simulate linearity, which simply rescales the amplitude of the input signal as in Eq. (7). Suppose the neurons were fully connected to every other neuron by a fully populated network, maximizing the rank of the transformation matrix $\hat{O}$. Inevitably, a linear combination and convolution

of such neurons only resulted in a linearly behaving network, regardless of network connectivity:

$$let\, \vec{I}^{\,T}(t) = \hat{O}\vec{I}^{\,D}(t) = \hat{\lambda}\vec{I}^{\,D}(t),$$ (7)

$$I^F(t) = \hat{w}_D \cdot \vec{I}^{\,T}(t) = \hat{w}_D \cdot \hat{\lambda}\vec{I}^{\,D}(t) = \vec{w}_{D\text{eff}} \cdot \vec{I}^{\,D}(t),$$ (8)

where the transformation in Eq. (7) was replaced by a linear function, while the final signal in Eq. (8) was a linear combination of the driving signal with $\vec{w}_{D\text{eff}} = \sum_i^N w_{Di}\lambda_i$. The above demonstrated linear neurons' limited computational capability and be completely defined by the input signal from Eq. (8), while the design matrix $\vec{w}_{D\text{eff}}$ merely scales the input. To enable the network to implement complex computation, a neuronal behavior with nonlinear characteristics and robust mathematical formalism was adopted. Chua's nonlinear circuit analysis (Chua and Kennedy 1988) introduced the concept of memristive systems as a neuron-like two-terminal element with characteristic nonlinear and memory qualities. The memristor nonlinearly related the integrated voltage (magnetic flux, φ) with charge and acts similar to a charge dependent resistor:

$$d\varphi = M\,dq\, or\, M(q) = \frac{d\varphi}{dt}\Big/\frac{dq}{dt} = V/I.$$ (9)

The relation was strictly nonlinear and solved differentially, which required holistic circuit analysis when the element was incorporated within a network (Chua 1980). A memory attribute was readily illustrated in the memristor's dependency on charge accumulation, which was desirable to any learning system. Discovery of physical memristor devices (Strukov et al. 2008) and complex circuit oscillations depicting chaotic trajectories (Chua et al. 1993; Chua and Itoh 2008) has enabled the construction of nonlinear circuits capable of harnessing emergent chaotic behaviors. Observations of neurons physically adapted to environmental changes through a recurrent feedback mechanism (Carbajal et al. 2015) paralleled the oscillatory behavior found in Chua circuits. Likewise, incorporation of a continuous feedback enabled adaptive and responsive computing (Hermans et al. 2015).

2.2 Characterization: Power-Law Dynamics

The above mathematical construct illustrated the importance and central role of the network connectivity and functional topology described by the transformation $\hat{O}$ within the machine learning platform. As described by Maass and Legenstein (2005), Maass et al. (2002), neuroscience concluded that a small-world network

maximized information transfer while minimizing energy usage. This phenomenon was observed throughout the natural world—occurring in cases such as complex geological formations, flock behavior, and disease proliferation—and continues to be a central topic within chaos and network theory. Heuristic evidence concluded that network design ascribed with such features enable optimal performance. A defining characteristic of a small-world topology was the length distribution of interacting elements to behave as a power-law, i.e., the strength of $\vec{w}_i^n$ scales as $d^{-\beta}$, where d is the interneuron distance:

$$\left| \vec{w}_j^m |d| \vec{w}_i^n \right| \propto d^{-\beta} \, or \, \left| fft\left(\vec{w}_j^m\right) |t| fft\left(\vec{w}_i^n\right) \right| \propto f^{-\beta}, \tag{10}$$

where the second relation utilized Pontryagin duality, $d \rightarrow c \times t$ with c as the speed of light, and the Fourier transform of the first. A network adhering to these constraints was capable of sustained persistent activity even due to small perturbations (Goudarzi et al. 2012; Haimovici et al. 2013; Sussillo and Abbott 2009) and was in a "critical" state which allowed for maximal information transfer.

We examined the device for emergent nonlinear properties considered fundamental to brain function, which were not observed for individual atomic switches operating in simpler geometries—namely, recurrent dynamics and the activation of feedforward sub-networks. The presence of small-world dynamics within the ASN devices was demonstrated by applying a constant DC bias (Fig. 2a) across a particular region of the network. This produced persistent, bidirectional fluctuations—both increases and decreases—in network conductivity. In the absence of complex structures within the network, conductivity would increase monotonically under constant DC bias, as in the case of a single atomic switch. Previously reported (Stieg et al. 2012) current fluctuations of this kind are ascribed to recurrent loops in the network
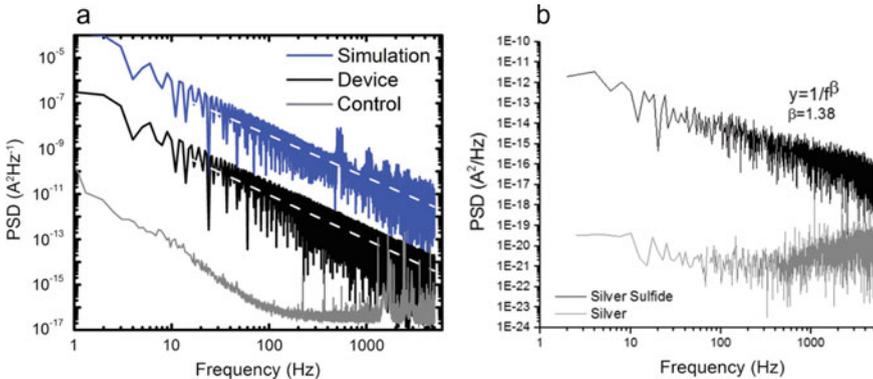
Fig. 2 Dissipative power-law behavior indicative of self-organized criticality. The electrical current response of a physical and simulated ASN device in **a** under constant external voltage bias was used to characterize network activity. Network switching/activity timescales showed **a** dissipative power-law response **b** indicative of a scale-free network

that create complex couplings between switches, resulting in network dynamics that chaotically converge to a semi-steady state even under constant bias. A single switch turning ON did not simply lead to an increased potential drop across the next junction in a serial chain, but entropically redistributed voltage across many recurrent connections that can ultimately perturb the system into a new equilibrium as a net change in network conductivity. These fluctuations were not attributable to uncorrelated flicker noise, as shown by comparing the Fourier transformed current responses in Fig. 2b of the devices to constant voltage before and after sulfurization. The formation of atomic switch junctions expanded the degree of correlation in current fluctuations, producing small-world 1/f-like behavior across the entire sampled range. This behavior was distinct from that of control devices which flattens to white noise and some high energy, high frequency fluctuations attributed to arcing between neighboring wires.

2.3 Atomic Switch Plasticity

In conjunction with optimizing the reservoir's transformation capability in $\hat{O}$, reservoir learning inherently required a memory quality and plasticity for selective information storing (Jaeger 2001). A powerful feature of atomic switches and memristive-like devices was the observation of a brain-like physical phenomena known as *Long-Term Potentiation* (LTP) and *Short-Term Potentiation* (STP). Both function and memory have been ascribed to STP and LTP dynamics in neurological studies (Maass and Legenstein 2005; Maass et al. 2002). Neuron signal transduction through potentiation spikes showed timing dependencies which directly encoded information within the spike's line shape. Simultaneously, brain functionality and behaviors developed as neuron ensembles cooperatively spiked to adopt specific emergent behaviors.

These neurological phenomena were observed (Nelson and Abbott 2000; Sussillo and Abbott 2009) within the active Ag_2S region in the atomic switch as aggregations of Ag^+ cations. Observation of a large impedance change in the atomic switch under an external voltage was attributed to a crystal transition of the active material Ag_2S (Gusev and Sadovnikov 2018). This transition gave rise to a weakly memristive behavior prior to the formation of Ag filaments across the interface. In the absence of continued applied bias, the conductive filaments eventually returned to their stoichiometric, thermodynamically favored equilibrium state, reverting the atomic switch to its initial high OFF resistance (Fig. 3a). Continued application of bias voltage resulted in a concurrent increase in electric current through the device, which then further drove migration of silver cations toward the cathode. At the cathode mobile silver cations were subsequently reduced to Ag^0, forming a highly conductive Ag nanofilamentary wire. The completion of this filament resulted in a strong transition to an ON state (Fig. 3a) with a dramatic increase in conductivity (Fig. 3b). Removal of the applied bias resulted in filament dissolution as the device again returns its thermodynamic equilibrium state (Fig. 3b). The completion and dissolution of this
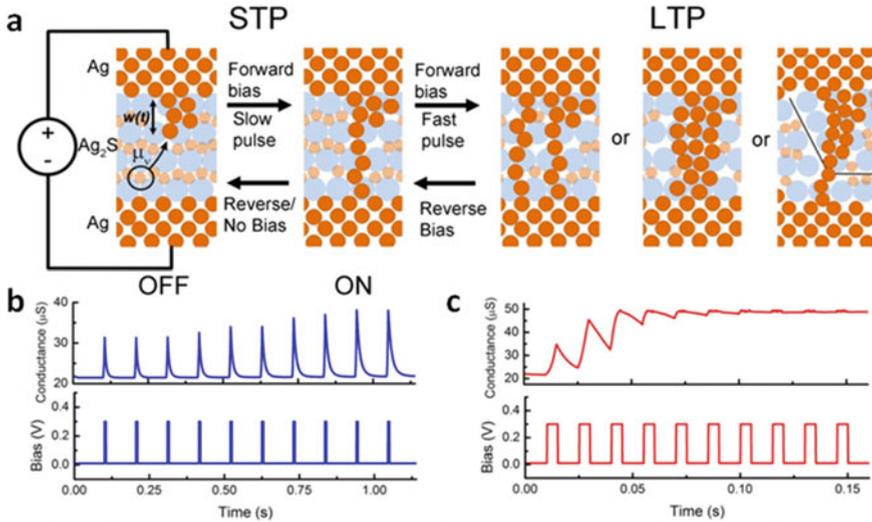
Fig. 3 Spike-time dependent plasticity in a single atomic switch. Continued stimulation of the atomic switch caused formation of metallic filaments across the gap/active layer in **a**. The electrical response became increasingly dominated by tunneling mechanisms derived from single atom "contact". A 300 mV spike 5 ms width voltage train at a period of 100 ms in **b** stimulated the atomic switch to form a single Ag filament. Single atom contact increased conductance to the ON state during stimulation while thermodynamic dissolution drove the system back to the OFF state. In **c**, the pulse train period was shortened to 10 ms allowing multiple filament formations. Measured conductance monotonically increased before reaching a stable conductance state. Filament structure and stability modulated the electrical response and emerge as empirically determined as Short-Term Potentiation (STP) in **b** and Long-Term Potentiation (LTP) in **c**

filament characterized strongly memristive behavior. Continuous application of a bias voltage served to increase filament thickness as additional silver cations was reduced, causing thickening of the metallic filament (Fig. 3c). This dynamic process has been shown to alter the dissolution time constant and can be externally controlled by changing the input bias pattern (e.g., pulse frequency). Such changes in volatility can be interpreted as short-term or long-term potentiation (STP and LTP).

2.4 Resistance Training

The network's ability to physically encode information within the filament led us to develop a resistance training algorithm to control the network's memory capabilities. The dependency of filament formation on voltage history and charge accumulation illustrated memristive behavior within the atomic switch. Circuits utilizing memristive behavior tend to have complex trajectories with nondeterministic solutions and are classified as Chua circuits (Chua et al. 1993). Initial conditions and

stochastic fluctuations helped determine the circuit's trajectories and operational regime, thereby having statistical control on its operation. The circuit parameters of impedance, inductance, and capacitance were used to determine the trajectories of Chua circuits, but other driven systems have included filters, op-amps, and other sources for noise. Though the atomic switches' equivalent parameters evolved with operation, impedance change dominated most of the activity while periphery parameters were treated using thermodynamic approximations (Sillin et al. 2013). A resistance training algorithm was constructed to tune the network operational regime, while using resistance stability as a thermodynamic approximation of the periphery parameters in simulated models, see Sect. 2.5.

The resistance training experiments were performed using a precision source measure unit (National Instruments 4132) and a high-speed switch matrix (National Instruments 2532) within a PXIe unit (National Instruments 8108), enabling rapid resistance measurements between any combinations of 16 chosen electrodes. Resistance training was implemented through repetition of a two-step process as shown in Fig. 4. In the first step, an electrode A was selected randomly and the resistance between this reference and every other electrode was measured using a small (200 mV, 10 ms) bipolar pulse in order to minimize influence on network resistances, as shown in Fig. 4a. The individual resistances of electrode A with each of the other 15 electrodes, R_{Aj}, defined the network state by calculating the total resistance between
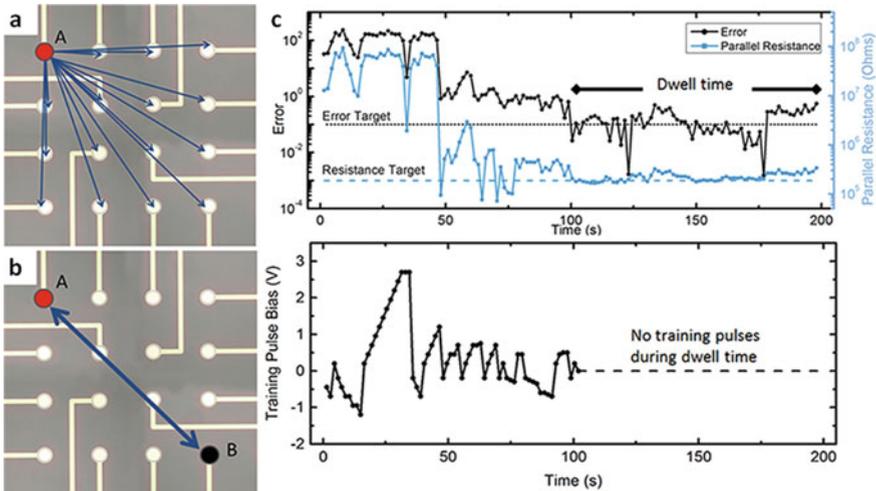
Fig. 4 Resistance learning algorithm. Determination of network-wide stability/activity under operating conditions was conducted using a target resistance learning algorithm. A schematic of the write and verify training scheme, and typical results for an individual training trial. **a** Sub-threshold measurement pulses establish the parallel resistance of A, followed by **b** a larger training/write pulse between A and B. **c** The parallel resistance of A is recorded and compared to the target after each training pulse, when error is minimized the training ceases and the duration of the achieved target state is recorded as the dwell time

electrode A and the rest of the network, as though the paths from electrode A to every other electrode were resistors in parallel:

$$R(i) = \left(\sum_{\substack{j=0 \\ j \neq A}}^{15} \frac{1}{R_{Aj}} \right)^{-1}. \tag{11}$$

This quantity is hereafter referred to as the "parallel resistance". In the second step, a second electrode B was selected randomly, and a large unipolar training pulse (100 ms, $>\pm 200$ mV) was applied to influence the parallel resistance of electrode A, as shown in Fig. 4b. Using the same electrode I/O scheme, the measure/training cycle was repeated until the parallel resistance of A reached the target resistance. For all trials the target resistance was predetermined, irrespective of the initial network resistance.

In order to achieve training, an error function and rule set was devised. This system was designed to create sensible and consistent voltage adjustments even when both target resistance and parallel resistance error could vary by several orders of magnitude. The error function and rule set also correctly accounted for events in which the parallel resistance overshot the target. Convergence of the parallel resistance to the target resistance was evaluated using an error function:

$$E(i) = \frac{1}{2} \left(\frac{R(i)}{R_g} - \frac{R_g}{R(i)} \right), \tag{12}$$

where R_g was the target resistance, and $R(i)$ was the parallel resistance. The error $E(i)$ was calculated after each pulse/measure cycle, and adjustments to the training pulse bias were made by evaluating the relative change in error $C(i) = \frac{E(i)}{E(i-1)}$ from one cycle to the next using Eqs. (13) and (14), which are described below.

Equation (13) concerned changes in the absolute magnitude of $C(i)$ to evaluate changes in the absolute magnitude of the training pulse, $V(i)$. If the previous training pulse resulted in a large decrease in error, $|C(i)|$ would be less than 1. If significantly less than 1, as determined by an empirically determined threshold, $C_m = 0.6$, then the training pulse $V(i)$ was considered productive and no changes were made. If the previous pulse produced a significant increase in error, $|C(i)|$ would be greater than 1. If $|C(i)|$ was greater than $\frac{1}{C_m}$, the pulse was considered counterproductive and the training pulse magnitude was reset to a minimum value, $V_{\min}$. If $|C(i)|$ was between C_m and $\frac{1}{C_m}$ (i.e., approximately equal to 1) then the error had not significantly changed as a result of the previous pulse, indicating little influence on the parallel resistance. The pulse magnitude was then increased by V_{inc}.

$$V(i+1) = \begin{cases} V(i), & if \, |C(i)| < C_m \\ V_{\min}, & if \, |C(i)| > \frac{1}{C_m} \\ V(i) + V_{\text{inc}}, & if \, C_m < |C(i)| < \frac{1}{C_m} \end{cases} \tag{13}$$

Next, Eq. (14) was used to determine the need for changes to the polarity of the training pulse. If $R(i)$ and $R(i-1)$ were both greater or both less than R_g then there was no overshoot and no need to reverse the bias, which is reflected by positive value for $C(i)$. However if $R(i)$ changed enough with respect to $R(i-1)$ that it overshot R_g, $C(i)$ would be negative. In this case the training pulse voltage $V(i)$ was reversed in sign, and its magnitude was automatically reset to the minimum pulse bias V_{min}.

$$sgn(V(i+1)) = \left\{ \begin{array}{l} sgn(V(i)), if\, C(i) > 0 \\ -sgn(V(i)), if\, C(i) < 0 \end{array} \right\} \tag{14}$$

A single pulse/measurement cycle lasted 1.5 s, and the time required to reach the target resistance state was defined as the "convergence time". Upon reaching the target resistance, training pulses ceased and network resistances were measured every 0.5 s until the parallel resistance decayed away from the target and the error exceeded 0.5 (roughly equivalent to 50% error). This duration was defined as the "dwell time". The entire convergence/dwell time sequence constituted a single resistance training trial, an example of which is presented in Fig. 4c. When a trial completed, new electrodes would be randomly selected and the training process was repeated after a 30 s delay.

Individual resistance states were the result of conductive silver filaments which bridge the Ag|Ag$_2$S|Ag gaps, and each filament was vulnerable to thermodynamically driven dissolution. Not surprisingly, a deterministic model of interacting thermodynamic variables was not available, and stability of target resistance was hard to predict. Figure 5a shows the distribution of dwell times for networks at the target resistance (R_g=200 kΩ). The distribution suggests a power-law dependency, with
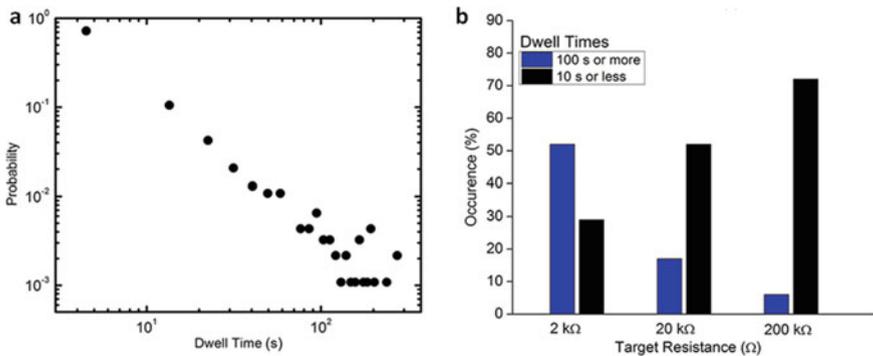
Fig. 5 **Dwell times** vary widely but depend on the target resistance. In **a,** networks are repeatedly trained to 200 kΩ and their dwell times are recorded. By repeating the training program many times on different networks, statistical distributions suggest that the probability $P(D)$ of a dwell time lasting for duration D follows a power-law relationship. Dwell times are generally 10 s or less, with occasional states lasting 100 s or more. As in **b,** at low target resistances, the final configurations are stable, with over 50% of trials resulting in a final state lifetime of 100 s or more. As target resistance increases, the final states are proportionately less stable

dwell times of less than 10 s being most common and occasionally lasting 10 times longer. This distribution was found to depend heavily on the target resistance value, as shown in Fig. 5b. When R_g=200 kΩ, <10 s dwell times accounted for 72% of trials, but at 2 kΩ, dwell times of 100 s occur in more than 50% of trials. This is the expected result given the underlying operational mechanism of individual atomic switches. Lower resistances were achieved when an individual switch has a thicker conductive filament across the insulating layer, making them more resistant to thermodynamically driven dissolution. In the ASN, lower network resistances are more likely to have an abundance of parallel filamentary pathways, making the target state more resilient against changes from an individual filament. These factors of solved state stability outweigh any effects from repeatedly training the network.

2.5 Simulation of Atomic Switch Network

A complementary study on the effects of global stimulation was done in simulation to form a microscopic understanding of the device dynamics. The simulated network was comprised of interconnected atomic switches using a modified state equation (Joshua Yang et al. 2013; Strukov et al. 2008). A current controlled memristor model was adopted undergoing ionic drift dynamics at the Ag|Ag$_2$S|Ag interface based on previously published works (Demis et al. 2015; Sillin et al. 2013; Biolek et al. 2009). The state variable, $w(t)$, represented the doped region produced by migration of Ag$^+$ mobile ions from pure Ag into the Ag$_2$S layer. Reduction of Ag$^+$ at the cathode precipitated Ag nanowire formation with its physical dimensions determining its impedance and characteristic memristive behavior. The atomic switch was observed to have at least two operational regimes characterized by a low and high resistance state, ON/OFF, respectively. Simple linear super positioning of the two states captured memristive behavior eloquently and a state variable $w(t)$ was defined:

$$V(t) = \left[R_{on} \frac{w(t)}{w_0} + R_{off} \left(1 - \frac{w(t)}{w_0} \right) \right] I(t). \tag{15}$$

Above is the classical Ohm's law equation with $w(t)$, the characteristic filament length, capturing filament formation, and determined using the ionic drift model:

$$\frac{dw(t)}{dt} = \left[\mu_v \frac{R_{on}}{w_0} I(t) \right] \Omega(w). \tag{16}$$

A physical restraint was imposed on $w(t)$ to account for finite dimensions through the use of a window function Ω:

$$\Omega(w) = \frac{w(w_0 - w)}{w_0^2}. \tag{17}$$

Modifications were made to the above model to account for nanowires forming Ag|Ag$_2$S|Ag interfaces. The formulation for voltage-induced α/β phase transition of the Ag$_2$S from monoclinic acanthite to the more conductive body-centered cubic argentite was introduced as well as formation/dissolution of conductive filaments (Gusev and Sadovnikov 2018). Applied voltage triggers the α/β phase transition creating a more conductive Ag|β-Ag$_2$S|Ag junction and, more importantly, allowed for Ag cation migration within the Ag$_2$S in the direction of the electric field. Reduction of Ag cations into Ag^0 occurs at the cathode thereby creating substructures within the β-Ag$_2$S to ultimately form conductive filaments. The removal of the applied voltage no longer induced Ag cation migration and the system was allowed to return to its thermodynamically favored equilibrium state. A stochastic term and dissolution term incorporated this thermodynamic behavior to the system. The term further modeled any variability among the nanowires and the structural stability of the Ag filament. Stratonovich integrals were employed to solve the stochastic differential equations. The network was numerically solved as an ordinary differential equation using standard Kirchhoff's current laws with each node–node connection considered as a single atomic switch.

$$\frac{dw(t)}{dt} = \left[\mu_v \frac{R_{on}}{w_0} I(t) \right] \Omega - \tau(w(t) - w_0) + \eta(t) \tag{18}$$

A numerical simulation was constructed based on experimentally determined parameters to model and verify theoretical propositions. Emulating the construction of the device, voltage nodes/electrodes were arranged in a square grid and subsequent node–node connections were introduced to represent nanowires (Fig. 6b). Connections were categorized either as short-range, within a lattice constant, or long-range and randomly assigned to produce characteristics of nearest neighbor or random network topologies (Sillin et al. 2013). The initial strength of each atomic switch was randomly sampled following a power-law distribution in (10) (Maass and Legenstein 2005) with $\beta = 1.38$.

Resistance training was successfully conducted using the simulated ASN device. Network connectivity was created by randomly distributing 250 connections with 10% of the links constrained to a length of a lattice constant within a 5 × 5 grid. The grid size was increased as previously published SEM images showed connections outside the 4 × 4 area (Avizienis et al. 2012a, b; Demis et al. 2015; Stieg et al. 2012). Training pulses were administered between two nodes using the scheme described in Eqs. (11)–(14). Resistance training in the simulated network proceeded as observed in the device (Fig. 6) and could involve a direct approach to the target, or through a series of overshoots. The simulation allowed a complete analysis of every change in resistance in each link, and Fig. 6b showed the net change that occurred in each link during the training process. The changes were widespread rather than localized along a single conductive pathway, which supports the hypothesis that network training was achieved by global interactions.
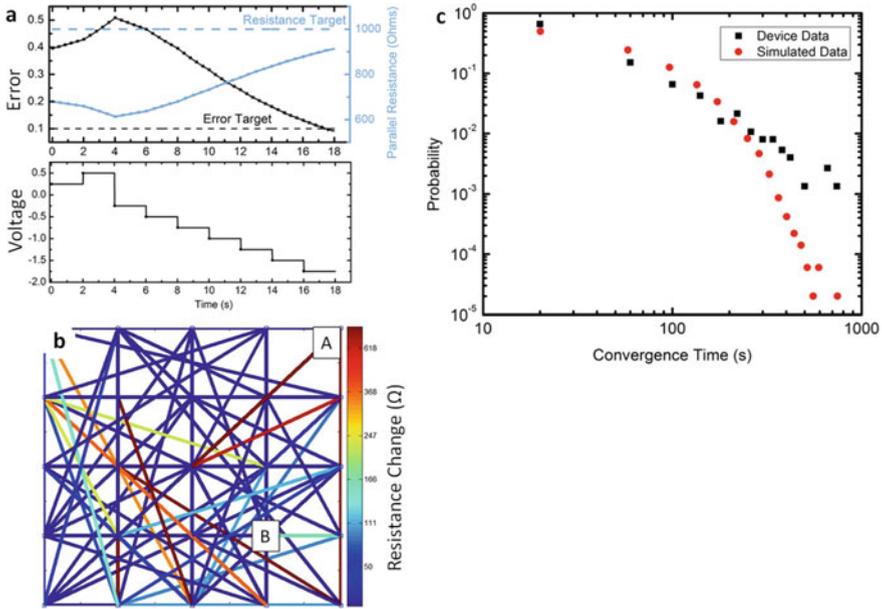
Fig. 6 Resistance learning algorithm convergence of models. A simulated ASN shows similar behavior in resistance training, and network-wide changes in resistance. A parallel resistance training program identical to the experimental one was used to successfully train parallel resistance. **a** Target resistance was 1000 Ω, error target was 0.1, training pulses were 100 ms in 250 mV increments, measurement pulses are not necessary in simulation. The effects of resistance training are presented in **b**, which shows the net resistance change in each link from start to finish. The simulation shows network-wide changes in resistance even though training pulses were applied exclusively from *A* to *B*

2.6 Implementation: Error Checking

As an illustration of the ASN's utility as a reservoir, the benchmark task of determining bit parity was taken to both measure memory quality and network tenability. As outlined in Furuta et al. (2018), Natschläger and Bertschinger (2004), the task was a fundamental algorithm in signal processing and error checking. Typical data streams of bytes of bits required one bit, the parity bit, to record the parity of the overall byte. Information transfer across multiple servers can corrupt data by inverting one bit thereby changing the overall parity of the data byte. The parity bit ensured identification of corrupted bytes and subsequent repairing to allow for reliable data transfers. Typical data bytes are 8 bits long, which our experiment adapted as time-separated binary pulse sequences.

Previously published work (Demis et al. 2015, 2016; Sillin et al. 2013) enabled us to conclude that reservoir computing was not a universal computing paradigm, but more similar to a B-machine as imagined by Turing (1950). As such, the reservoir and task needs to be tailored for optimal utility in performing the parity test. Simulations
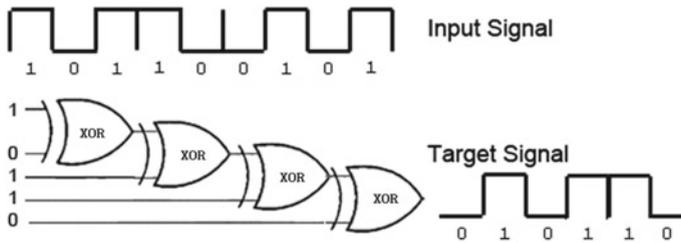
Fig. 7 **Error checking task**. Presented is an illustration of the parity check used in data transmission for error checking process. The parity of the number of 1's within a 5-bit byte is evaluated with a sliding window 5 bits wide to generate multiple tasks. The initial input shows an odd parity and evaluated as 0 for the desired target signal. As the 5-bit window moves across the signal, the parity changes and reflected in the target signal. The above task was encoded as a voltage pulse sequence into the ASN device where each bit was represented by V_0 or V_1 voltages in a time-separated series. Task complexity increased with increasing number of bits per bytes rather than number of bytes as the check was only executed once per byte

of the ASN were highly leveraged for this purpose for its ease of use, device editing, and high throughput despite statistically underperforming w.r.t. the ASN device. A number of simulations were conducted to determine optimal signal encoding, activation regime, and processing timescales (Fig. 7).

2.7 Simulated ASN Error Checking Results

Implementation of machine learning tasks required the design of an encoding technique such that signal transduction stimulates the network into an excited state with the proper mathematical transformation. As outlined in Sect. 2, reservoir computation can be represented into a mathematical design matrix $\hat{O}$ through spectral analysis where its rank and eigenvalue determine the complexity of the transformation (Verstraeten et al. 2009). However, reservoir size limits the reservoir's computational capability as the rank of the design matrix cannot exceed the readout layer. This limitation is typically overcome by ensuring overlap of the design matrix within the desired mathematical operating regime by applying constraints to network activity. However, signal transduction can perturb the reservoir outside desired activity and clever design of transduction was required such that the signal can both encode information while maintaining the reservoir at a specific state.

Encoding of digital information was explored by modulating the signal in either the amplitude, frequency, or phase space. For the parity test, digital information was spread among 8 bits with each bit in a binary state of either 0 or 1. Bytes of digital information were represented as pulse voltages, Gaussian wave packets, and phase-shifted sine waves for amplitude, frequency, and phase modulation, respectively. The binary states 0 and 1 were assigned to preset voltage amplitudes (V_0, V_1), wave packet frequency shifts (f_0, f_1), and phase shifts (φ_0, φ_1) and the Euclidean

distance between the binary states empirically optimized. Inspection of the state equation in (9) and (16) reveals the dependencies of these parameters w.r.t. network activity. Indeed, simulation results concluded that the total voltage was the deterministic factor on reservoir activity while changes in frequency and phase negligibly perturbed the reservoir. An amplitude-modulated encoding procedure was adopted for all subsequent experiments with voltages ranging from 0.1 to 7.0 V.

Reservoir activity was initialized in simulations using the resistance training algorithm in Sect. 2.4 (Sellers 2007). Resistance values with short convergence times were desired as the resistance training algorithm invariably encoded unnecessary information from the procedure which limited the reservoir's memory capacity. These states coincided with resistance values within R_{ON} or R_{OFF} as well as states that had been thermodynamically stable after repeated approaches. Introducing a highly stochastic signal while maintaining the resistance state was capable of cleansing any information encoded by the resistance training algorithm and was incorporated into a post-experiment protocol. Optimal network activity was determined heuristically while prioritizing reliability over performance.

A typical proportional–integral–derivative (PID) loop algorithm provided a constant feedback voltage which maintained target reservoir activity. Constant stimulation by application of the driving signal eventually accumulated charge and excited the reservoir outside the target resistance state, observed as LTP in Sect. 2.3. Conversely, STP dynamics concluded that inactivity or sub-threshold voltages unable to counterbalance the thermodynamic inhibitive processes relaxed the system. Stability of the target resistance state was controlled by a PID feedback loop by dedicating one of the I/O nodes for this purpose. The feedback applied a constant DC signal for an integral time equal to the training time. Maintenance of the resistance state followed identical trends as the resistance training algorithm in Sect. 2.4.

Optimal training times were determined by maximizing the dwell times at target resistances and empirically investigated in simulation. Learning was implemented on the reservoir using a number of training datasets, following the mathematics in Sect. 2 and details found in Sect. 4.3. Each training set was followed by a testing dataset to determine the effectiveness of the learning algorithm using the accuracy in Eq. (6) as a metric of success. The procedure of providing a training dataset for the learning algorithm and subsequent testing of performance was repeated, while constraining reservoir activity using the PID feedback loop.

The ASN's performance dependencies w.r.t. dataset size and number of learning repetition was investigated in simulation to determine optimal dwell times. Simulations of the ASN device revealed an occurrence of under-learning at 0.250 s (Fig. 8a) and over-learning at 4.000 s (Fig. 8b). This was observed as drastic increases in the NMSE at these timescales as well as a deterioration of signal propagation. The occurrence of over-learning was theoretically predicted as we approached the network's memory capacity by saturating it with training data. Under-learning manifested as fluctuations in performance across various reservoir sizes due to limited memory retention times. The over-learning occurred as the learning algorithm became ill-posed and over-determined with excessive training sets. Optimum dataset lengths
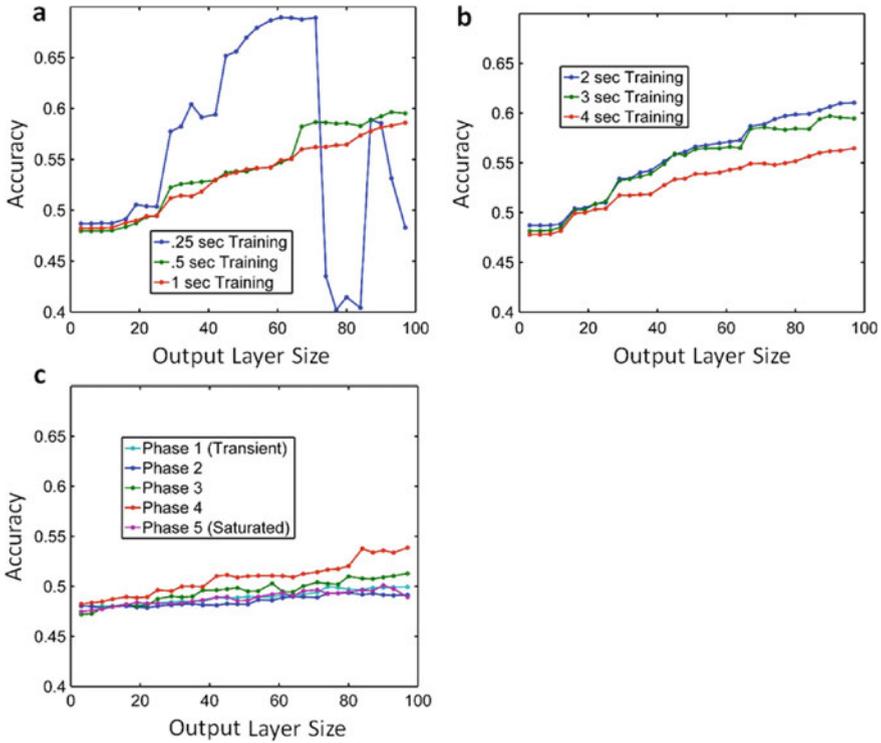
Fig. 8 Encoding optimization using ASN simulation platform. A simulation of the ASN device performing the parity check task was conducted to determine optimal operating parameters. Temporal memory quality was evaluated w.r.t. the size of the output layer, length of the learning sequence used a, b, and operating time c. Under-learning was observed at 0.25 s (blue) length datasets as chaotic performance was measured regardless of network size. Over-learning in b at 4.00 s (red) as continued increase in the dataset length reduced reservoir performance. Subsequent phases of operation c each 1.00 s in duration determined optimal operating time. Omitting the transient phase (light blue), subsequent phases monotonically increased performance and peaked at 4.00 s (red) while further operation in phase 5 decreased performance

were discovered to be 1.000 s while optimal total operating time to be 4.000 s (Fig. 8c). Subsequent experiments were thus encoded as amplitude-modulated datasets 1.000 s in length with 0.250 s pulse width and learning applied within a 2.000 s window.

2.8 Neuromorphic ASN Device Error Checking Results

The optimal parameters found from simulation were implemented on the ASN device and investigated for routes of optimization. Identical instrumentations were used as the resistance training algorithm while incorporating a PID feedback mechanism

for sustaining reservoir activity (Fig. 9a). The error checking task was implemented over a population of 5 devices using all possible permutations of the 16 I/O electrodes and followed similar trends as depicted in Fig. 9. Initial experimentation on
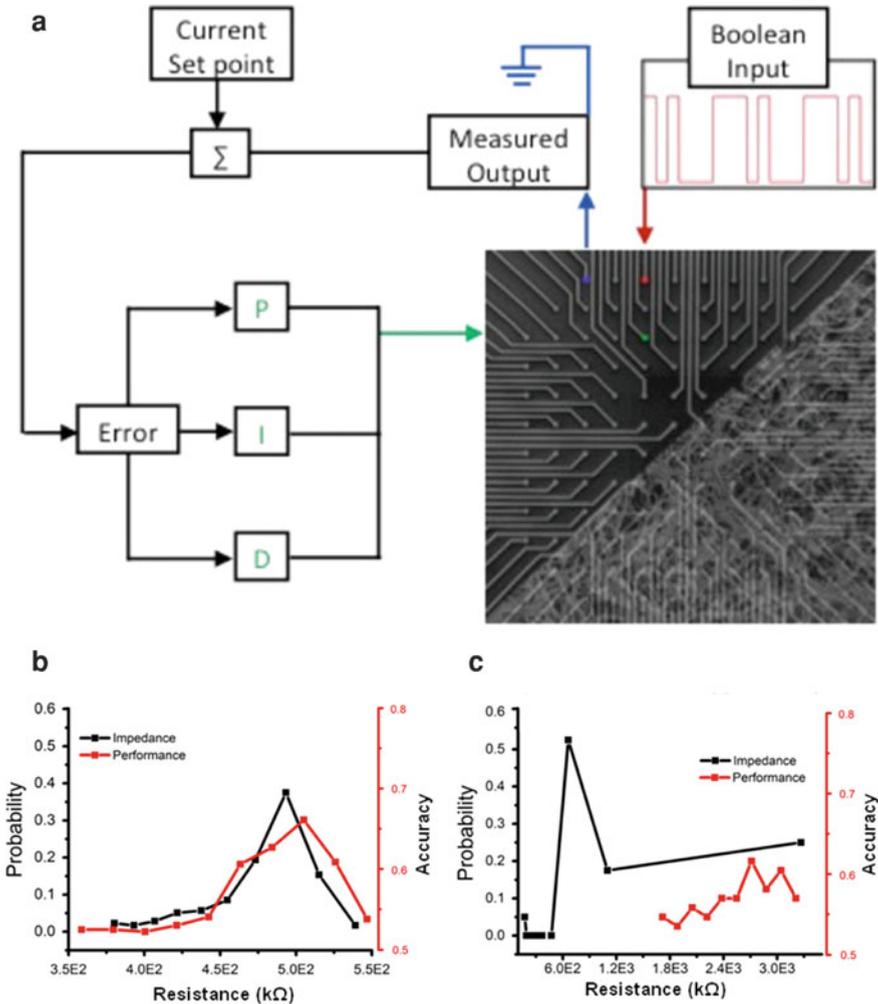
Fig. 9 Error checking of ASN platform. Schematic of RC using ASN devices: Three I/O electrodes are selected to form the stimulus/control loop for RC: Boolean input streams are delivered to an individual I/O electrode underlying the ASN network (red); a system ground (blue) enables real-time monitoring of current flowing through the network controlled by a feedback-driven bias voltage delivered to (green) a nearby location. The ASN was stimulated with a statistical survey of pulse widths (nΔt) and pulse heights (nΔV) ranging from 250 ms to 0.01–7.00 V. Testing occurred immediately after resistance training with a fixed weight configuration. The datasets above achieved accuracies **a**, **b** between 65 and 78% from ~5,000 trials compared to ~50% from a purely stochastic reservoir

individual ASN devices was performed to determine relevant optimal amplitude and timescales using identical procedures as in simulation. The simulation's predicted optimal parameters were corrected to include amplitude scales of 0.01–7.0 V while other parameters were retained.

Device performance and reliability were heavily dependent on the device's resistance state following their description in Sect. 2.4. Shorter dwell time devices at correspondingly higher resistance states performed with increasing reliability and accuracy, despite implementing similar training and operating times with devices at lower resistance states. A bimodal distribution of metastable resistance states was found with dwell times that exceeded 100 s for 3 different devices that followed similar dynamics to Fig. 5a. The presence of these metastable states and similar power-law behavior indicated the device activating toward a self-organized critical state (Goudarzi et al. 2012; Stieg et al. 2012) with the two resistance states centered at 500 kΩ and 600 kΩ possible chaotic attractor states. However, true verification of device criticality required statistical experimentation using exact and identical parameters which proved impractical.

Continued trials revealed devices initialized outside of the near-critical resistance states performed poorly with accuracies below 50%, which prompted subsequent device testing to operate within the bimodal states to explore device optimization. Devices initialized below 500 kΩ (Fig. 9b) performed at 71.35% ± 6.38% accuracy while those initialized above 600 kΩ attained a similar and maximized performance of 73% ± 5%. Despite seemingly small differences, this trend manifested throughout all trials alongside a characteristic high dispersion in the distribution with kurtosis values of 2.73 and 5.94 for devices at 600 kΩ and 500 kΩ, respectively. Kurtosis values beyond 3 indicated a non-Gaussian distribution and increasingly became dominant below 500 kΩ. Rapid bipolar switching manifesting as abrupt changes in current supply was observed below this range and simulation experiments revealed increased filaments forming under similar conditions. The non-Gaussian statistics and filament completion events indicated a shift in the operational characteristics of atomic switches and decreased performance metrics. Consequently, network resistance state became increasingly complex and divergent thereby driving network dynamics toward increasingly nonlinear behaviors and outside target functionality. Past results (Demis et al. 2016; Sillin et al. 2013) and similar experiments (Carbajal et al. 2015; Hermans et al. 2015) clearly indicated the requirement for task-specific network design. Diverging resistance states, dynamic changes in atomic switch behaviors, and poor performance concluded the network was being driven outside of its error checking design.

Further experimentation evaluating other device parameters such as stimulation amplitude, size, and learning timescales resulted in minor changes to network performance, highlighting the importance of network dynamics. The stability of the network resistance state was an evident metric in controlling computational capability and network state. Spontaneous organization of 2 convergent resistance states highlights the underlying critical dynamics which maximized device performance. Controlling such device dynamics through the use of mechanisms such as a feedback loop (Hermans et al. 2015) seems evident for further progress.

3 Outlook

We concluded that we were able to use network activity and stability via resistance initialization to describe the network state for a thermodynamically driven reservoir, the ASN. Due to the task-specificity inherit in machine learning, it was paramount to characterize and catalogue a reservoir's "state" that corresponds to task-specific functionalities. Likewise, previous results (Sillin et al. 2013) developed a map for pattern recognition using higher harmonics. Typical reservoir characterization in the literature utilized entropy and Shannon theory, which requires repeated experiments under identical conditions. Current devices utilizing memristor-like reservoirs are difficult to control with such precision, thus, a desire for an alternative characterization of reservoir state has been necessary. In general, characterization of reservoir functionality has proven difficult for real "edge-of-chaos" systems. Although this requirement has strictly not been within the reservoir computing framework, the development of reservoirs with diverse and rich functionality expands the framework's utility.

Despite limited addressable electrodes, the ASN device was capable of outperforming simulated networks as network complexity, density, and critical dynamics were utilized more effectively in the device. We have presented a clear methodology to implement reservoir computing on a neuromorphic device by developing observable metrics such as power-law behavior, activation of STP/LTP, and resistance state. As outlined in Sect. 2, reservoir performance was theoretically predicted to depend on nonlinear dynamics, network topology, and task design. The commensurate development of simulations aided in implementing theoretical models onto neuromorphic a platform and task evaluation. Previously accomplished tasks such as pattern classification, bit logic, and T-maze decision-making task highlighted the capabilities of the atomic switch as an integrated memory and logic component.

4 Methods

Structurally complex networks comprising of highly interconnected, functional nanostructures are fabricated using varying degrees of top-down and bottom-up processes, ranging from the random deposition of monodisperse nanowires to the electroless deposition (ELD) of metallic nanostructures. By identifying the benefits and limitations of each technique, a nanoarchitectonics approach was adopted (Demis et al. 2015, 2016) whereby the size of nucleation sites for ELD was used as a control parameter for network growth in order to maximize atomic switch connectivity while retaining control over network topology. The silver networks were functionalized to have atomic switch interfaces at the junctions of their component nanostructures. This process produced a complex network of interacting elements whose operational properties provide a basis for the memorization and transformation of environmental information. Further, their inherent volatility results in patterns of robust electrical

activity. Through this specific combination of nanoscale elements and design principles for the production of structurally complex systems, ASN devices provide the balance of intrinsic memory capacity and nonlinear operation required for advanced hardware implementations of neuromorphic computing, specifically in the reservoir computing paradigm.

Patterned seed networks proved the most versatile fabrication method, and utilized a combination of top-down with bottom-up fabrication, a powerful fabrication method described as nanoarchitectonics. Random neural networks are grown on a SiO_2 substrate with Cu post nucleation sites via electroless deposition producing a massively interconnected nanowire network with functionally brain-like features (Avizienis et al. 2012a, b). A patterned 150 nm layer of platinum electrodes are prepared on the substrate with a 500 nm layer of SU-8 polymer insulating the platinum, only circular nodes 30–50 μm in diameter are left exposed for electrical contact and arranged in a square grid. Standard lithographic techniques using a negative resist were used to then deposit pre-pattern Cu posts serving as nucleation sites. Dendritic nanowires are grown via electroless deposition with a 50 mM $AgNO_3$ solution controlling the size and shape of Cu posts to control the topological distributions of the Ag nanowire network (Avizienis et al. 2013). Exposure to sulfur gas at 10^{-1} Torr at 130º C for 3 min developed functional Ag|Ag$_2$S|Ag interfaces.

With a density controlled network in mind, our group started using electrochemistry to grow a recurrent silver network using copper seeds. Network growth occurs through an electroless deposition (ELD) reaction through individual atom displacement reactions between Ag^+ and Cu^0 based on respective electric potentials. A spontaneous ELD reaction is preferred over an electric one due to the lack of a need for external power and the delicate nature of electrochemical reactions. In this particular case, silver atoms are oxidized while copper is reduced during the galvanic displacement reaction:

$$Cu^0(s) + 2Ag^+(aq) \rightarrow 2Ag^0(s) + Cu^{2+}(aq) \; E_{red} = -1.26 \text{ V}.$$
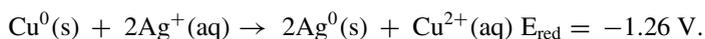
Successful implementations of the ELD reaction above allowed us to design a technique using highly patterned top-down photolithography combined with the complex spontaneous growth provided from the reaction above. These patterned seed networks consist of a 2 μm layer of AZ nLOF 2020 (a negative photoresist), a soft bake, followed by UV photolithography, and a post-exposure bake. This resist is developed in MF26A, rinsed with isopropanol, and a 300 nm layer of copper is then deposited and lifted off overnight in acetone. At the end of this process, a patterned grid of copper posts 300 nm high is left. The size and pitch of these posts were refined over time to give the most desirable silver crystal growth.

When first designing a purpose-built device to emulate mammalian brain activity, dendritic silver structures were desired. However, over time it was realized that the connections provided by these structures were unreliable and difficult to reproduce. Through changing the size of the copper posts, a morphological transition was found showing that a seed site of 1×1 μm^2 up to 3×3 μm^2 leads to fine nanowires. Seeds

between $3 \times 3\ \mu m^2$ and $10 \times 10\ \mu m^2$ yield a mixture of nanowires with branched dendritic structures, while posts larger than $10 \times 10\ \mu m^2$ produce only dendrites.

4.1 Design Optimization Results

Directed nanowire growth to create dendritic structures followed diffusion-limited aggregation with Mullins–Sekerka instabilities. For sparse concentrations of $AgNO_3$, diffusion-limited aggregation (DLA) dynamics prevail where Ag^+ cations displace Cu^0 atoms in discrete non-interacting reactions. Reduced silver atoms accumulate on the surface of the copper posts and develop, in a steady-state evolution, metallic nanostructures. Solidification of silver particles undergoing DLA obeys the mathematical formulations of Fick's Law, modified by Mullins–Sekerka instabilities which describe pattern formation of accumulated metal nanostructures. We describe the kinetics of formation using ion clusters to describe the heterogeneity of the solution's concentration. Clusters of ions diffuse through the solution, creating a wave of ions that initiate the ELD process at the seed. Starting with Fick's law to describe the diffusion:

$$D_{Ag}\nabla^2\mu_{Ag} = \frac{\partial\mu_{Ag}}{\partial t}; \ D_{Cu}\nabla^2\mu_{Cu} = \frac{\partial\mu_{Cu}}{\partial t}; \tag{19}$$

Here we use D_{Ag} and D_{Cu} the diffusion constants for $AgNO_3$ and pure copper, respectively, with as the diffusion potential. As the wavefront of silver reacts with copper, aggregated silver atoms at the seed sites accumulate, pushing the growth front toward the wavefront. The solid–liquid interface perturbs the diffusion field, moving slowly and continuously renormalizing the ion gradient in solution. Growth of the solid–liquid interface, via the non-equilibrium process of electroless deposition, is mediated by the continuity equation:

$$M v_n = \left[\delta\mu_{Ag} D_{Ag}\nabla\mu_{Ag} - \delta\mu_{Cu} D_{Cu}\nabla\mu_{Cu}\right] \cdot \hat{n}. \tag{20}$$

The miscibility gap, M, and the normal velocity, v_n, of the interface, determine the population exchange during single displacement reactions with $\delta\mu_{Ag}$ as the fluctuation in chemical potential due to concentration heterogeneity. Growth of the solid–liquid interface results in the Mullins–Sekerka instability which is due to competing dynamics between steady silver nanostructure growth and dynamical expansion of the growth front. Once the rate of metal nanostructure growth exceeds the diffusion rate, a depletion region emerges that no longer contains enough silver atoms for sustainable displacement. Regions adjacent to the initial growth front contain sufficient ion concentration to participate in ELD, forming side branches. Depending on the rate of formation, the chemical potential at the interface is described by the Gibbs–Thomson boundary condition:

$$\mu(r_0) = -d_0\varepsilon. \tag{21}$$

Here, the chemical potential at the interface, $\mu(r_0)$, is dependent on the surface curvature, ε, and d_0 the characteristic length of the seed. In the simplest case, the value of Eq. (21) was approximated to be the value of Eq. (20) during equilibrium. The non-equilibrium process at the solid–solution interface determines a characteristic length scale:

$$d_0 = \frac{\gamma}{M}. \tag{22}$$

where γ is the surface tension. Solutions to Eqs. (19)–(22) for a planar interface with small perturbations in ionic concentrations are solved in (Langer 1980). Extending this model for multiple perturbations will show dendritic growth that we experimentally demonstrate in controlled fabrication of the ASN. Equations (21) and (22) show parameters of control over the morphology of seed-directed nanowire growth. The reactivity dependence on curvature can be controlled by varying shape, size, and pitch of copper seeds. Surface tension and miscibility gap can be controlled through varying the copper spacing and distribution. Understanding DLA under Mullins–Sekerka instability conditions provides control and reproducibility over self-organizing nanowire networks. Pattern formation due to Mullins–Sekerka instabilities presented here is a linear approximation of the dynamical behavior of dendrite formation. Experimental testing confirmed that when the size of the copper seed is on the order of 1–5μ Mullins–Sekerka instabilities are suppressed, and the growth of metallic nanowires continues without nucleation of side branches.

In order to explore the concept of fabrication through self-organization, the mathematical principles of diffusion-limited aggregation (DLA) and ELD are combined to guide a nanoarchitectonics approach using the electroless deposition of silver. An extensive experimental study of this fabrication method found that the critical parameter for the growth of nanowires was the size of the copper seed post which is theoretically predicted in Eq. (22) due to the factors of surface curvature and surface tension (Avizienis et al. 2013). The diverse wire lengths included long-range and short-range atomic switches, facilitating both globally and locally distributed patterns of switching activity in the ASN. Due to the variation in nanowire diameters, we infer each junction to have a variable gap size and subsequent atomic switch size, thereby increasing the number of available resistance states to the ASN (Avizienis et al. 2012a, b; Stieg et al. 2012). This fabrication method offers control over network density and structure by introducing two important parameters: seed size and spacing, which nucleate wire growth.

4.2 Hardware and Instrumentation

Electrical characterization of the devices was conducted through current–voltage (I–V) spectroscopy using a bipotentiostat (Pine Instruments model AFCBP1) in conjunction with either a data acquisition module (National Instruments USB 6259) or a multiplexed (National Instruments PXI 1073) source-measurement unit (National Instruments PXI 4130). The maximum bandwidth of the measurement systems was 1 MHz and 10 kHz enabling 2 Ms and 20 ks s^{-1} with 16-bit resolution, respectively. ASN devices were designed to accommodate 64 electrode 40 μm Pt contacts within a 2.5×2.5 mm^2 grid where atomic switches were grown. Subsequent data analyses were carried out using MATLAB 2018b (MathWorks) and Origin 8.1 (OriginLab Corporation).

4.3 Reservoir Computing Implementation

All reservoir experiments were conducted on an 8×8 grid containing an estimated 10^8 atomic switch junctions using the 64 electrodes as I/O interface layers. A single electrode was selected to inject the electrical input signal, while another electrode was chosen as the counter electrode as shown in Fig. 9a. The control signal delivered a feedback voltage to an electrode in proximity to the input electrode. Voltage signals were simultaneously measured from the remaining 61 electrodes using the data acquisition module (National Instruments USB 6259). Reservoir computing was implemented following the mathematics presented in Sect. 2 with the input layer consisting of only the input electrode and the output layer constructed from the 61 measuring electrodes.

References

E. Abbe, *Contributions to the Theory of the Microscope and the Microscopic Perception* (Springer, 1873)

A. Abraham, Artificial neural networks, in *Handbook of Measuring System Design*, ed. by P.H. Sydenham, R. Thorn (Wiley, 2005)

N. Aramaki, Y. Shimokawa, Y. Fuwa, A parallel ASIC VLSI neurocomputer for a large number of neurons and billion connections per second speed, in *IEEE International Joint Conference on Neural Networks* (Singapore, 1991)

A. Ascoli, R. Tetzlaff, L.O. Chua, *Continuous and Differentiable Approximation of a TaO Memristor Model for Robust Numerical Simulations.* Springer Proceedings in Physics (2017)

R.C. Atkinson, R.M. Shiffrin, Human memory: a proposed system and its control processes. Psychol. Learn. Motiv. **2**, 89–195 (1968)

A.V. Avizienis, H.O. Sillin, C. Martin-Olmos, H.H. Shieh, M. Aono, A.Z. Stieg, J.K. Gimzewski, Neuromorphic atomic switch networks. PLoS ONE **7**(8), e42772 (2012a)

A.V. Avizienis, H.O. Sillin, C. Martin-Olmos, H.H. Shieh, M. Aono, A.Z. Stieg, J.K. Gimzewski, Neuromorphic atomic switch networks. PloS One **7**, e42772 (2012b)

A.V. Avizienis, C.M.-O. Henry, O. Sillin, M. Aono, J.K. Gimzewski, A.Z. Stieg, Morphological transitions from dendrites to nanowires in the electroless deposition of silver. Cryst. Growth Des. 13 (2013)

J.W. Backus, Can programming be liberated from the von Neumann style? A functional style and its algebra of programs. Commun. ACM **21** (1978)

Z. Biolek, D. Biolek, V. Biolkova, SPICE model of memristor with nonlinear dopant drift. Radioengineering **18** (2009)

L. Büsing, B. Schrauwen, R. Legenstein, Connectivity, dynamics, and memory in reservoir computing with binary and analog neurons. Neural Comput. **22**, 1272–1311 (2010)

J.P. Carbajal, J. Dambre, M. Hermans, B. Schrauwen, Memristor models for machine learning. Neural Comput. **27**, 725–747 (2015)

G. Chen, X.F. Wang, Complex networks: small-world, scale-free and beyond. IEEE Circuits Syst. Mag. **3**, 6–20 (2003)

L.O. Chua, Device modeling via basic nonlinear circuit elements. IEEE Trans. Circuits Syst. **27** (1980)

L.O. Chua, M.P. Kennedy, Neural networks for nonlinear programming. IEEE Trans. Circuits Syst. **35** (1988)

L.O. Chua, C.W. Wu, Synchronization in an array of linearly coupled dynamical systems. IEEE Trans. Circuits Syst.-I Fundam. Theory Appl. **42** (1995)

L.O. Chua, M. Itoh, Memristor oscillators. Int. J. Bifurc. Chaos **18** (2008)

L.O. Chua, C.W. Wu, A. Huang, G.-Q. Zhong, A universal circuit for studying and generating chaos-Part I: routes to chaos. IEEE Trans. Circuits Syst.-I: Fundam. Theory Appl. **40** (1993)

E.C. Demis, R. Aguilera, H.O. Sillin, K. Scharnhorst, E.J. Sandouk, M. Aono, A.Z. Stieg, J.K. Gimzewski, Atomic switch networks nanoarchitectonic design of a complex system for natural computing. Nanotechnology **26** (2015)

E.C. Demis, R. Aguilera, K. Scharnhorst, M. Aono, A.Z. Stieg, J.K. Gimzewski, Nanoarchitectonic atomic switch networks for unconventional computing. Jpn. J. Appl. Phys. **55** (2016)

C. Du, F. Cai, M.A. Zidan, W. Ma, S.H. Lee, W.D. Lu, Reservoir computing using dynamic memristors for temporal information processing. Nat. Commun. **8** (2017)

T. Furuta, K. Fujii, K. Nakajima, S. Tsunegi, H. Kubota, Y. Suzuki, S. Miwa, Macromagnetic simulation for reservoir computing utilizing spin dynamics in magnetic tunnel junctions. Phys. Rev. Appl. **10**, 034063 (2018)

A. Graves, A.M. Mohamed, G. Hinton, Speech Recognition with Deep Recurrent Neural Netw. (2013)

A. Goudarzi, C.T. N. Gulbahce, T. Rohlf, Emergent criticality through adaptive information processing in Boolean networks. Phys. Rev. Lett. 108 (2012)

A.I. Gusev, S.I. Sadovnikov, Effect of small size of particles on thermal expansion and heat capacity of Ag2S silver sulfide. Thermochim. Acta 660 (2018)

A. Haimovici, E.T. Pablo Balenzuela, D.R. Chialvo, Brain organization into resting state networks emerges at criticality on a model of the human connectome. Phys. Rev. Lett. 110 (2013)

T. Hasegawa, A. Nayak, T. Ohno, K. Terabe, T. Tsuruoka, J.K. Gimzewski, M. Aono, Memristive operations demonstrated by gap-type atomic switches. Appl. Phys. A **102**, 811–815 (2011)

M.H. Hassoun, *Fundamentals of Artificial Neural Networks* (MIT Press, 1995)

D.O. Hebb, *Organization of Behavior* (Wiley, New York, 1950)

M. Hermans, M. Burm, T. Van Vaerenbergh, J. Dambre, P. Bienstman, Trainable hardware for dynamical computing using error backpropagation through physical media. Nat. Commun. **6** (2015)

J.J. Hopfield, Artificial neural networks. IEEE Circuits Devices Mag. **4**, 3–10 (1988)

C.P. Husband, S.M. Husband, J.S. Daniels, J.M. Tour, Logic and memory with nanocell circuits. IEEE Trans. Electron Devices **50** (2003)

H. Jaeger, The "echo state" approach to analysing and training recurrent neural networks - with an Erratum note. *GMD Report 148.* German National Research Center for Information Technology (2001)

J. Joshua Yang, D.B. Strukov, D.R. Stewart, Memristive devices for computing. Nat. Nanotechnol. **8**, 13–24 (2013)

J.-H. Kim, K.-H. Han, Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. IEEE Trans. Evol. Comput. **6**, 580–593 (2002)

A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. (2012)

Lang, (1986) https://doi.org/10.1063/1.97114

J.S. Langer, Instabilities and pattern formation in crystal growth. Rev Mod Phys. **52** (1980)

C.G. Langton, Computation at the edge of chaos: phase transitions and emergent computation. Phys. D **42**, 12–37 (1990)

M. Lukoševičius, H. Jaeger, Reservoir computing approaches to recurrent neural network training. Comput. Sci. Rev. **3**, 127–149 (2009)

W. Maass, R. Legenstein, What makes a dynamical system computationally powerful? in *New Directions in Statistical Signal Processing: From Systems to Brain*, ed. by P. Haykin, T.J. Sejnowski, J. Mcwhirter (2005)

W. Maass, T. Natschläger, H. Markram, Real-time computing without stable states: a new framework for neural computation based on perturbations. Neural Comput. **14**, 2531–2560 (2002)

C. Mead, Neuromorphic electronic systems, in *IEEE* (IEEE, 1990)

Möller, (1987) https://doi.org/10.1103/PhysRevB.36.1284

T. Natschläger, N. Bertschinger, Real-time computation at the edge of chaos in recurrent neural networks. Neural Comput. **16** (2004)

NEC, NEC integrates NanoBridge in the Cu interconnects of Si LSI (2009), https://phys.org/news/2009-12-nec-nanobridge-cu-interconnects-si.html

E. Nedaaee Oskoee, M. Sahimi, Electric currents in networks of interconnected memristors. Phys. Rev. E **83** (2011)

L.F. Nelson, S.B. Abbott, Synaptic plasticity: taming the beast. Nat. Neurosci. 3, 1178 (2000)

T. Ohno, T. Hasegawa, T. Tsuruoka, K. Terabe, J.K. Gimzewski, M. Aono, Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. Nat. Mater. **10**, 591–595 (2011)

A. Romero, P.L. Carrier, A. Erraqabi, T. Sylvain, A. Auvolat, E. Dejoie, M.-A. Legault, M.-P. Dubé, J.G. Hussin, Y. Bengio, Diet networks: thin parameters for fat genomics (2017)

B. Schrauwen, D. Verstraeten, J. Van Campenhout, An overview of reservoir computing: theory, applications and implementations, in *15th European Symposium on Artificial Neural Networks* (2007), pp. 471–482

D. Sellers, An overview of proportional plus integral plus derivative control and suggestions for its successful application and implementation (2007), https://web.archive.org/web/20070307161741/http://www.peci.org/library/PECI_ControlOverview1_1002.pdf

H.O. Sillin, R. Aguilera, H.-H. Shieh, A.V. Avizienis, M. Aono, A.Z. Stieg, J.K. Gimzewski, A theoretical and experimental study of neuromorphic atomic switch networks for reservoir computing. Nanotechnology **24**, 384004 (2013)

O. Sporns, Small-world connectivity, motif composition, and complexity of fractal neuronal connections. Biosystems **85**, 55–64 (2006)

A.Z. Stieg, A.V. Avizienis, H.O. Sillin, C. Martin-Olmos, M. Aono, J.K. Gimzewski, Emergent criticality in complex Turing B-type atomic switch networks. Adv. Mater. **24**, 286–293 (2012)

D.B. Strukov, G.S. Snider, D.R. Stewart, R.S. Williams, The missing memristor found. Nature **453**, 80–83 (2008)

D. Sussillo, L.F. Abbott, Generating coherent patterns of activity from chaotic neural networks. Neuron **63** (2009)

K. Terabe, T. Nakayama, T. Hasegawa, M. Aono, Formation and disappearance of a nanoscale silver cluster realized by solid electrochemical reaction. J. Appl. Phys. **91**, 10110–10114 (2002)

Y. Timofeeva, S. Coombes, Sparks and waves in a stochastic fire-diffuse-fire model of Ca2 release. Phys. Rev. E **68** (2003)

J.M. Tour, L. Cheng, D.P. Nackashi, Y. Yao, A.K. Flatt, S.K. St. Angelo, T.E. Mallouk, P.D. Franzon, NanoCell electronic memories. J. Am. Chem. Soc. **125**, 13279–13283 (2003)

T. Tsuchiya, M. Ochi, T. Higuchi, K. Terabe, M. Aono, Effect of ionic conductivity on response speed of SrTiO3-based allsolid-state electric-double-layer transistor. ACS Appl. Mater. Interfaces **7** (2015)

T. Tsuruoka, T. Hasegawa, K. Terabe, M. Aono, Operating mechanism and resistive switching characteristics of two- and three-terminal atomic switches using a thin metal oxide layer. J. Electroceram. **39** (2017)

A.M. Turing, Computing machinery and intelligence. Mind **59**, 433–460 (1950)

H. van Houten, C. Beenakker, Quantum point contacts. Phys. Today **49** (1996)

D. Verstraeten, Reservoir computing: computation with dynamical systems. PhD thesis, Ghent University (2009)

M.M. Waldrop, The chips are down for Moore's law. Nature **530**, 144–147 (2016)