PERSPECTIVE
https://doi.org/10.1038/s41928-017-0006-8

nature electronics

The future of electronics based on memristive systems

Mohammed A. Zidan[1], John Paul Strachan[2]* and Wei D. Lu[1]*

A memristor is a resistive device with an inherent memory. The theoretical concept of a memristor was connected to physically measured devices in 2008 and since then there has been rapid progress in the development of such devices, leading to a series of recent demonstrations of memristor-based neuromorphic hardware systems. Here, we evaluate the state of the art in memristor-based electronics and explore where the future of the field lies. We highlight three areas of potential technological impact: on-chip memory and storage, biologically inspired computing and general-purpose in-memory computing. We analyse the challenges, and possible solutions, associated with scaling the systems up for practical applications, and consider the benefits of scaling the devices down in terms of geometry and also in terms of obtaining fundamental control of the atomic-level dynamics. Finally, we discuss the ways we believe biology will continue to provide guiding principles for device innovation and system optimization in the field.

The way we work, live and interact has been fundamentally changed by the exponential growth of electronics and computing capacity over the past few decades, with even more dramatic changes envisioned in the future. Historically, advances in computing capabilities have been driven by device scaling, where reduction in device size has led to improved cost, speed and power consumption. In return, tremendous resources have been invested to sustain the scaling trend, with billions of nanoscale transistors powering everything from smartphones to supercomputers today. However, with the increased fabrication cost and impending fundamental physical limits, device scaling alone can no longer provide the desired performance gains. New devices and, perhaps equally importantly, new computing principles are needed to satisfy our ever-growing appetite for data and information.

With the end of Moore's law in sight, the semiconductor industry has been in a 'the King is dying' phase, with many technologies looking to fill the ensuing power vacuum. Memristors[1,2] are one such technology. A memristor typically has the simple form of a two-terminal structure, where a total of only three layers — two electrodes that send and receive electrical signals and a 'storage' layer in between — are needed. From the outside, the device looks like a resistor, and thus offers the potential for very-high-density integration and low-cost fabrication. However, unlike a static resistor, the storage layer can be dynamically reconfigured when stimulated by electrical inputs[2,3]. This material reconfiguration leads to memory effects, where changes in physical parameters, such as the device's resistance, can be used to store data and also directly process data.

This resistive device with an inherent memory effect is appropriately termed a memristor (memory + resistor), or more broadly defined as a memristive system. A class of memristors used in memory applications is also often called resistive random access memory (RRAM)[4,5]. Fundamental device studies have shown that the device can be scaled to sub-10 nm feature sizes[6] and retain memory states for years[7], while offering desirable device properties such as sub-nanosecond switching speed[8,9], long write–erase endurance[10] and low programming energy (for example, nanoamperes[11]). It should be noted that while many of the above favourable properties have been realized repeatedly, a single material system that combines them all simultaneously remains an open challenge.

Fundamental device and materials characterizations have shown that the reconfiguration in memristors is typically driven by internal ion redistribution[3,5]. Specifically, the storage layer in a memristor is typically a few nanometres thick, thus even a moderate voltage drop across it can create a large enough electric field to drive the ionic processes to alter the ionic configuration of the material. A typical process involves the oxidation, migration and reduction of cation or anion species in the storage layer, leading to changes of the local conductivity, normally in the form of the creation and annihilation of a conductive filament[3,5]. This process can be either abrupt (binary) or gradual (analogue), with different physical processes evolving at different timescales, leading to rich device behaviours in this seemingly simple device structure[12,13].

Here, we aim to evaluate the memristor's capability to drive new computing systems beyond Moore's law, and speculate on what may happen in the future. We see three categories that may significantly benefit from memristor developments: on-chip memory and storage, biologically inspired computing and in-memory computing (Fig. 1). These approaches can help overcome the obstacles facing today's computing architectures, and are of particular relevance to current and future computing needs: cognitive processing, big-data analysis and low-power intelligent systems based on the Internet of Things.

State of the art

The challenges for classical computing architectures today originate from the memory bottleneck and the high (energy and speed) costs associated with constant data movements between the memory and the processor, commonly referred to as the von Neumann bottleneck. In the most straightforward approach, memristors offer a solution as an ultrahigh-density memory layer that can be directly integrated on the processor chip, thus significantly reducing the memory bottleneck and improving the energy efficiency and speed of the system.

For instance, memristors (in the form of RRAMs) are much faster than hard disk drives and flash memory, while offering higher

[1]Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, USA. [2]Hewlett Packard Labs, Palo Alto, CA, USA.
*e-mail: john-paul.strachan@hpe.com; wluee@eecs.umich.edu

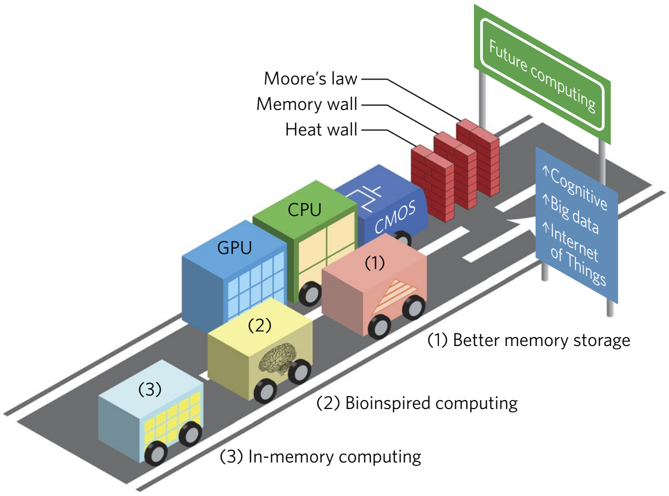

Fig. 1 | The race towards future computing solutions. Conventional computing architectures face challenges including the heat wall, the memory wall and the end of Moore's law. Developments in memristor technology may provide an alternative path that enables hybrid memory–logic integration, bioinspired computing and efficient reconfigurable in-memory computing systems. CMOS, complementary metal–oxide–semiconductor; GPU, graphics processing unit; CPU, central processing unit.

density, lower cost and nonvolatility compared with static random access memory (SRAM) and dynamic random access memory (DRAM). Moreover, unlike transistor-based memory elements, such as SRAM and DRAM, memristors can be directly integrated with a low thermal budget over the processor through very-high-density local interconnects, thus eliminating the slow and energy-hungry off-chip communications between memory and processor. Such properties allow memristors to simplify the memory and storage hierarchy, and introduce a significant boost to the computing system performance. A high-performance, three-dimensional (3D) integrated memristor memory can thus extend the lifetime of conventional von Neumann computing systems and improve the system's ability to process large amounts of information — a critical need in the big-data era.

Extensive research efforts have been carried out to facilitate practical use of memristors as a memory storage system. The primary goal of these efforts is to enhance the device performance and address challenges associated with large-scale implementations. This includes increasing the device speed, ON/OFF ratio, cycling endurance and data retention time. Further improvements include reducing the operating voltage and current and addressing the device variability challenges. At the circuit and system levels, large-scale implementations of memristor memory (RRAM) also need to address challenges such as sneak current and wire resistance. The sneak currents flow through unselected memory cells in a memristor crossbar array, and can cause errors during read and increased power consumption during write[14]. This problem can be addressed by engineering devices to have high current–voltage nonlinearities[15], by adding a selector device to the memory cell[16,17] or through system-level techniques to compensate for the read current distortion[14]. The finite wiring resistance of the metal lines connecting the memory cells poses another challenge during read and write[18]. This problem worsens at smaller feature sizes and can significantly impact the system operation if not handled properly. While there are still obstacles before RRAM captures a sizable market share from classical memory and storage technologies, the initial efforts are considered fruitful. At present, several companies have started offering RRAM products in the market[19–21], with a path towards 16 Gb already demonstrated[22]. The first commercial market for RRAM devices is likely to be embedded memories, while further developments can eventually bring RRAM products to the stand-alone memory and storage market[23].

Remarkably, memristors may play a larger role in computing systems beyond memory or storage. Owing to their ability to co-locate memory and compute in the same physical device, memristors are ideally suited to realize highly efficient bioinspired neural networks in hardware. Artificial neural networks have shown superior performance over classical systems in processing cognitive and data-intensive tasks, and recent advances in algorithm developments have led to performance even surpassing that of humans in specific complex tasks such as playing the game Go[24]. A neural network in its simplest form is a set of neurons connected by weighted synaptic connections (Fig. 2). Each synapse transmits information from the pre-synaptic neuron to the post-synaptic neuron, scaled by the synaptic weight. Typically, the network is trained by updating its synaptic weights to perform a specific task. Modern networks can have multiple (over 100) hidden layers, and thus require training and storage of an enormous number of synaptic connections. Up until now, implementations of neural networks have been mainly based on conventional computing hardware where the synaptic weights are stored in (off-chip) memory and need to be constantly loaded into the processing unit to compute the desired output to the next neuron. As a result, the performance is still fundamentally limited by the von Neumann bottleneck and requires enormous computing hardware resources and high power consumption during operation. In contrast, in a memristor-based implementation, a single device can simultaneously store the synaptic weight and modulate the transmitted signal[25] (Fig. 2). In this case, the transmitted signal (that is, current into the post-neuron) is determined by the product of the input signal (that is, voltage pulse from the pre-neuron) and the synaptic weight (represented by the memristor conductance), natively through Ohm's law. The natural co-location of memory and compute in the same memristor device eliminates the constant data movement, and can thus significantly improve the system efficiency.

As noted, the network structure can be directly mapped into a crossbar form in hardware (Fig. 2), where the inputs are connected to the rows of the memristor crossbar and the outputs connected to the columns. Furthermore, all inputs can be computed simultaneously in a single read operation, where the output current at a specific column is determined by the summed currents through all the memristors connecting the inputs to the particular column, through Ohm's law and Kirchhoff's law. In other words, a single read operation of an $N \times M$ memristor crossbar with N inputs and M outputs performs an $N \times (N \times M)$ vector-matrix multiplication, obtained naturally through physics. The same task will require $N \times M$ multiply-accumulate operations in a conventional system, highlighting the high degrees of parallelism in the memristor-based approach. The co-location of memory and logic and high level of parallelism are two of the most attractive features that make memristor-based neural network hardware highly efficient. Memristor-based hardware is also compatible with online learning, where the weights (memristor conductances) can be changed incrementally by applied voltage pulses following desired learning rules. In addition, for applications requiring processing raw signals from sensors and other devices, the compute can remain in the analogue domain and can thus further reduce energy, latency and chip area by eliminating the need for expensive conversion to and from digital signals.

Neuromorphic hardware is a particularly attractive area for memristor research, since at the system-level neural networks can tolerate many of the device non-idealities that are present today, such as inherent device variations (including device-to-device variabilities due to fabrication non-uniformity, and cycle-to-cycle variabilities due to the stochastic switching process[7]). In fact, device runtime stochasticity may be considered a favourable property

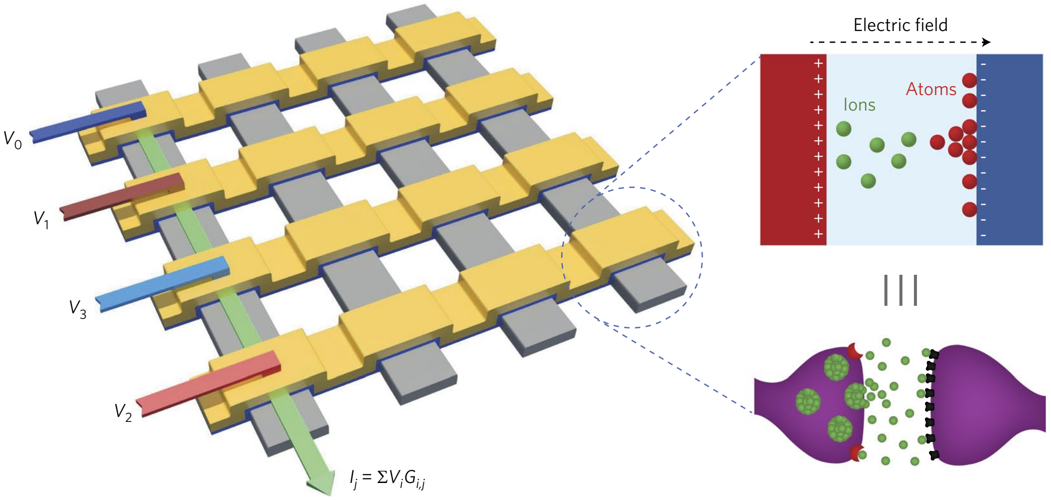

Fig. 2 | Hardware implementation of artificial neural networks in a memristor crossbar. A memristor is formed at each crosspoint and can be used to simultaneously store data and process information. In this approach, vector-matrix multiplication can be obtained through Ohm's law and Kirchhoff's law through a simple read operation. In addition, the internal dynamics of memristors can be utilized to faithfully emulate underlying processes in biological synapses. V_i, voltage applied at row *i*; I_j, current through column *j*; $G_{i,j}$, conductance of the memristor at the intersection of row *i* and column *j*.

that mimics real biological synapses and can act as a regularizer during training[26]. In addition, practical network operations do not require years of data retention, as in the case of storage systems, and requirement of device endurance may also be relaxed, since weight updates are often infrequent[27].

Mathematically, neuromorphic computations can be decomposed into a series of vector-matrix multiplication operations that are naturally implemented using the memristor crossbar structure. This structure can support a range of inputs and outputs, including standard feedforward networks such as perceptrons[28], as well as systems that mimic the spiking events from pre- to post-synaptic neurons, that is, spiking neural networks (SNNs)[29]. In these systems, 'learning' takes place through the relative strengthening and weakening of the synaptic connections (Fig. 2). Several examples of memristor-based neuromorphic hardware have already been demonstrated in the past couple of years. For example, memristor hardware performing pattern classification has been demonstrated, initially using a 2 × 10 crossbar[28] and later expanded to a 12 × 12 crossbar[30]. A generic dot-product engine using memristor arrays for neuromorphic applications was introduced in 2016[18], and a sparse coding chip that allows lateral neuron inhibition was developed using a 32 × 32 crossbar[31], followed by the demonstration of principal component analysis though online learning in a 9 × 2 crossbar[32]. Large-scale neural networks have also been demonstrated using phase-change memory, following the same principle[33].

In SNNs, a common learning rule implemented is spike-timing-dependent plasticity — the synapse between two neurons is strengthened when the pre-synaptic neuron spike precedes the post-synaptic neuron spike, and is weakened if the reverse. Indeed, it has even been suggested that memristance effects can explain spike-timing-dependent plasticity behaviour[34]. To date, many researchers have investigated memristor-based SNNs through full-system simulations using experimental device parameters[29,35,36], although large-scale implementations are still limited.

Another interesting example is the cellular neural/nonlinear network, developed in 1988 and supporting many of the properties observed in spatiotemporal sensing systems in the brain[37]. Here, a grid (typically 2D) of cells compute by following only local interactions (typically only nearest neighbours), and the state of all cells evolves dynamically in time. Computing applications for cellular networks include image processing and pattern recognition, and recent efficient implementations with memristors have brought new life and area-efficient physical realizations to this field[38].

Restricted Boltzmann machines are stochastic neural networks used in both supervised and unsupervised (without labelled data) modes. The key computations for restricted Boltzmann machines, including the contrastive divergence most frequently used for training, are strongly dominated by fetching weight values and computing vector operations with them, similar in that sense to other machine learning systems[39] described above. Hence, computations in memristor arrays[40] can in principle provide significant accelerations in this computing application. In particular, ref. [41] describes a memristor-based architecture for restricted Boltzmann machines that solves combinatorial optimization problems with over 50× increased performance and 25× lower energy than a single-threaded multicore system.

Overall, the development and exploration of brain-inspired computing models is an active area of research. A variety of approaches are being explored using different neuron, synapse and network models[42]. The different approaches often have different goals. For example, utilizing spiking in SNNs is believed to be (but still to be proven) key in achieving high energy and computational efficiency, as in biological systems. Meanwhile, the state-of-the-art object classification accuracies are currently realized in more loosely brain-inspired deep learning techniques[43]. As discussed here, memristors offer advantages as a hardware system to the wide variety of approaches in neuromorphic computing and machine learning models. Beyond this, we note that the same vector-matrix operations described above can be utilized to solve classical problems such as vector arithmetic functions and linear algebra[44], and other arithmetic and logic operations[45]. Hence, memristors can enable a promising in-memory computing solution that eliminates the memory bottleneck and data congestion, and lead to low-power, highly efficient hardware systems for different types of data-intensive tasks.

Device challenges and possible solutions

While memristors show great promise for a broad range of memory, computing and neuromorphic applications, there are clear materials and device challenges to be solved. These can vary based on the specific application. For example, in high-performance memory applications (such as DRAM replacement) it is critical to lower the programming current and voltage, raise the endurance and improve selector performance to minimize sneak currents, and to accomplish all of this with minimal device-to-device and cycle-to-cycle variability. This clearly poses challenges for the research community, but they are not unlike those already faced and conquered in CMOS scaling. Fortunately, for neuromorphic and similar computing applications, some of these specifications can be relaxed while new requirements, such as the stability of analogue states, become important[7].

In the case of online training of neural networks (via back propagation) in memristor arrays, it has been shown[46] that critical device issues include the programming bit precision (roughly 6 bits, or 64 conductance levels, are needed) and asymmetry in the ON versus OFF switching, since even a small asymmetry can degrade classification accuracy significantly. Other work[47] has shown that selectively optimizing the operating point for ON versus OFF switching, which can involve modifying the applied voltages and pulse widths, can produce acceptable results, but that the percentage of fully stuck ON or OFF devices can, in turn, lead to large errors. Nonetheless, many studies have shown that by retraining the network, in the presence of stuck ON or OFF cells, can almost fully compensate for the defects and regain the classification accuracy, even for up to 20% defects[48].

Some researchers have also studied offline trained systems, focusing on developing inference-only neural network accelerators[39,49] that can outperform the current state-of-the-art systems based on graphics processing units. In this case, the critical issues are maintaining high yield and low variability (as the system does not offer training around any defects or variability), thus maximizing the accuracy of the computed matrix operations. Because multiplication in memristor arrays is implemented by Ohm's law and Kirchoff's law, non-idealities in every memristor, or additional wire resistance in the array (leading to voltage drops on the rows and columns), generates computational errors. To this end, studies have shown that if a good model is known for these non-ideal behaviours and finite wire resistances are known, a compensating algorithm that maps the matrix values to appropriate conductance values can effectively eliminate the non-ideal effects[18]. This mapping, for example, not only takes into account the finite wire resistances, but also takes into account nonlinear resistance behaviour.

Scaling up and scaling down

With all of the advances, we note that memristor research is still in its infancy, with the first paper directly linking the device technology with the memristor concept published around ten years ago (although many devices had been studied previously with resistive switching characteristics)[2]. Most studies on memristor hardware systems are still carried out in academic research groups, with the majority of the demonstrations focusing on proof of concepts rather than aiming to build practical systems (Fig. 3). To bring memristor-based computing hardware to real-world applications, these systems need to be scaled up, possibly along three axes discussed below.

The first approach is to increase the size of the functional memristor networks. Scaling up towards a practical system size largely depends on the number of devices one can integrate into a system. A practical memory or computing system may require billions of functional memristive devices. Achieving this level of integration requires improving the yield of memristor device fabrication and close collaboration of university researchers with industry partners. In addition, system hierarchy needs to be developed and optimized to improve the scalability of the hardware. It is encouraging that research is already moving in this direction (Fig. 3).

Another aspect of scaling up is to improve system functionality by performing multiple tasks in the same hardware system. For example, the same physical fabric can be utilized to perform different functions, that is, neural networks, arithmetic operations and data storage, depending on the task and the data structure. Such an approach can produce natively scalable computing systems that can be dynamically reconfigured to fit different workloads[44]. In such a case, the function of the same physical memristive fabric can be dynamically reconfigured (redefined) in runtime purely through software, without any physical hardware modifications. Several challenges still need to be addressed to bring such a system to reality. For example, performing arithmetic operations using memristors requires tighter device distributions compared with storage and neural networks. In addition, long device endurance cycles are likely needed to allow efficient implementation of logic tasks. Recent device research efforts have already shown promising results[50], and we believe a memristor-based reconfigurable computing system[51] can be an attractive alternative to scaling up the system functionality.

The third scaling-up factor is driven by the integration process. Successful system-level scaling largely depends on reliable memristor–CMOS integration. In general, the operation of any memristor system will still require some CMOS circuitry to provide the necessary interface and control operations, although the functions of the CMOS layer will be significantly reduced and aggressive CMOS scaling is no longer necessary. Efficient memristor–CMOS integration is thus key to achieving any system gains. Typical approaches based on chip-level integration or through-silicon vias will not be able to provide the required bandwidth between the memristor layer and the CMOS circuitry, and monolithic integration of memristor arrays directly on top of CMOS circuitry with very-high-density local interconnects is necessary and has indeed been shown experimentally to be feasible[52,53]. Specifically, memristor device fabrication typically requires a low thermal budget that makes the integration compatible with the existing CMOS substrate. The simple device structure also requires few additional masks, making the integration cost effective. Three-dimensional multi-layered memristor arrays can be fabricated either in a layer-by-layer stacked fashion, or by using a vertical device structure akin to vertical NAND memory technology[54]. Successful 3D integration of memristors with CMOS circuitry can thus significantly increase the system density beyond simple device scaling.

Scaling up to 3D fundamentally creates new opportunities for more cognitive architectures, both physically and conceptually. For example, inspired by the sheer sizes of neural circuits, early work in cognition explored how brains might represent concepts and their relationships as sparse vectors in a high-dimensional space[55]. These hypervectors can have a dimensionality as high as 10,000, a number justified in part by the connectivities in neural systems. Working in such a large dimensional space (that is also subject to randomness and sparsity) leads to cognitive operations (binding concepts such as a person's name and their gender) that can be accomplished through relatively simple operations such as multiplication, addition and permutation, forming an algebra over that space called hyperdimensional computing. The challenge in hyperdimensional computing is that these operations are nonetheless highly memory intensive, given both the dimensionality and the expected number of vectors (for example, the 100,000 words in the English language). Recent work has explored implementations of hyperdimensional computing with memristive arrays using the 3D vertical structure[56,57] to both generate the random vectors and perform the multiplication, addition and permutation operations in situ. The ability to implement such a large-scale system in hardware clearly depends on the capability to scale up in three dimensions.

Fundamentally, the operation of memristors is driven by the internal ion redistribution in response to external stimulation. This makes it possible to 'scale down' the device, possibly down to the single-atom level. Note here 'scaling down' means not only reducing the physical device size, but also the ability to control the inner operations of the device at atomic scales[58]. Such level of precise control

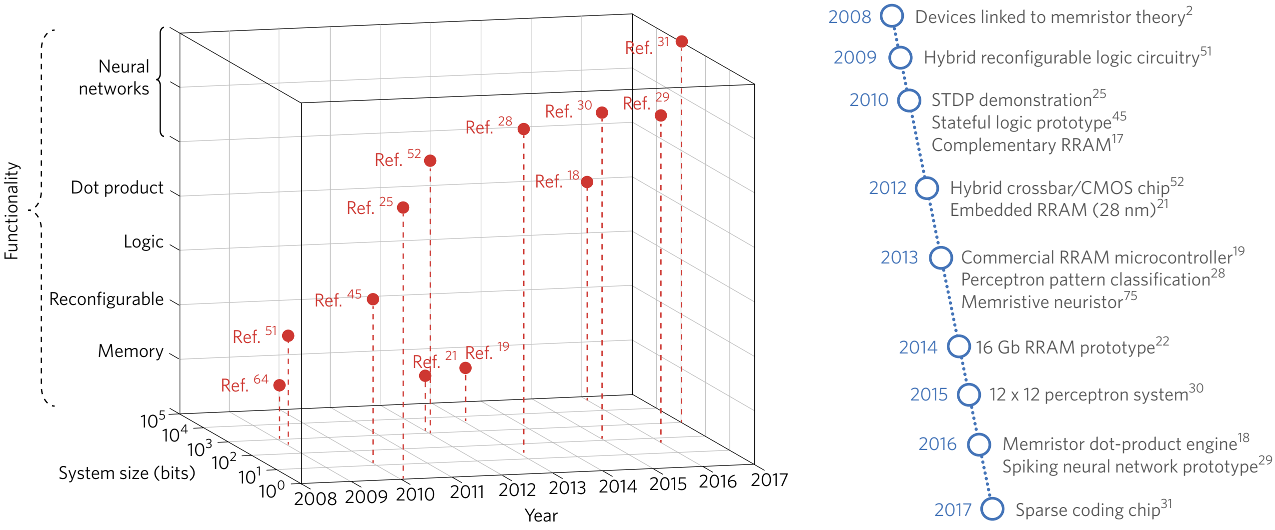

Fig. 3 | Rapid advances in memristor technology. Left: notable events in the past decade, showing the growth of memristor hardware system functionality and size. Right: timeline listing major memristor hardware developments. STDP, spike-timing-dependent plasticity.

will enable not only higher functional density, but also optimized device performance. For instance, device variability is due to the stochastic nature of the conductive filament formation process, and controlling the movement of individual ions can enforce predicable filament creation and minimize device variation. Previous studies have used graphene layers with controlled nanopores to control ion injection[59], metallic nanoparticles to enhance the local electric field[60] or aggressive scaling of the device to a size that allows only a single filament creation[61]. Understanding and control of the atomic processes can also help address the trade-offs that are often made in memristor devices. For example, lowering the set and reset current typically leads to a less-stable filament with a lower retention time due to spontaneous diffusion of the ions and atoms that leads to breakage in thin filaments. By confining the filament formation in atomically sized channels, it may be possible to improve the stability of the device even at very low programming current levels.

With increasing mastery and atomic control in memristors, another exponentially large space becomes available — the information per cell. We refer to this as 'scaling in', as the density of information available for storage and processing increases without increasing the amount of materials and chip area being utilized. Physical degrees of freedom are turned into informational and computational resources. A simple example is tuning the conductance range of a storage memristor. Binary conductance states (ON and OFF) give way to a range of intermediate conductance states that can encode more than 1 bit. Such multilevel behaviour has been exploited by a number of researchers[62–64], showing repeatable state control from 5 different conductance levels (over 2 bits) to over 64 levels (6 bits) for a single memristor. Physically, the continuum of conductances available can depend on parameters such as the density of free electrons, hopping sites, radius of the filamentary metallic channel, width of the tunnel barrier and so on. These physical parameters, in the memristor formalism, are referred to as the 'state variables' of the system and each state variable can offer a unique mechanism for control and dynamics tuning. An exponentially larger amount of information thus becomes available for every independent state variable that can be accessed in the system — for example, controlling the density of free electrons and the width of a tunnel barrier gives a combinatorially larger state space. Ultimately, the amount of information is limited by the total number of physical degrees of freedom in the system (proportional to the total number of atoms), but in practice, limitations will arise from the need to repeatedly control these state variables through, for example, electric field, current or temperature.

The role of chemistry and biological details

In bioinspired computing, one aims to mimic what is known about the brain and hopes that this will lead to a better computing system. However, how much biological detail is needed for a specific task is still an open question. For example, deep neural networks that only rely on the network topology but very little biological detail otherwise have been shown to be capable of performing tasks such as object classification in images with high accuracy after sufficient training[43]. However, recent developments have shown that even for deep neural networks, the more efficient training algorithms show striking resemblance to spiking-based learning rules observed in biology[65]. To a large extent, the debate over the role of biological details originates from two factors: the increased cost of implementing bio-like properties, and the lack of understanding (from neuroscience) of how these properties lead to practical functionalities. In this regard, the question may become easier to answer if one can faithfully mimic biological behaviours in hardware systems with little or no added cost — by using devices that natively possess biorealistic properties. Hardware systems based on such devices will offer new capabilities in artificial neural networks, and may even help accelerate the formulation and testing of hypotheses in neuroscience.

Recent findings at the device level show that it is possible to natively implement biorealistic properties in memristor devices without additional cost[12]. A representative example here is the calcium effect. The calcium concentration in the post-synaptic neuron increases following a spiking event of the pre-synaptic neuron, then decays within a timescale of tens of milliseconds. If the post-synaptic neuron also fires within this time frame, the calcium concentration can be enhanced above a threshold that triggers synaptic potentiation. The calcium concentration, and in turn the strength of the potentiation, depends on the relative timing of the pre- and post-neuron spikes and this mechanism has been argued as the possible underlying process behind the observed spike-timing-dependent plasticity and rate-dependent plasticity effects[66]. This type of behaviour has been recently observed in so-called second-order memristor devices[12,67] and diffusive memristor devices[68], where the rise and decay of one state variable (for example, local temperature) encodes the relative timing information and can subsequently modulate the change of a second state variable that represents the synaptic weight (for example, filament size). This level of biorealistic implementation at the device level can be extremely attractive in realizing bioinspired networks without increasing system cost.

Another interesting example is to examine the role of chemistry in biological systems, where synaptic weights are measured by the activities of receptors that can bind to neurotransmitters, where the binding process and the receptor activity are in turn driven by chemical reactions, for example, enzyme-enabled biocatalytic reactions[69]. From a device perspective, similar chemical reactions can help lower the energy required to operate the device and improve device reliability. For example, during resistive switching in a memristor, the device is converted from one stable state to another by overcoming an energy barrier between the two states[70]. The higher the energy barrier, the more stable the states are. However, a higher energy barrier means a larger bias voltage and, consequently, larger power is needed to program the device. By mimicking biology and using chemistry to assist the switching process, the effective energy barrier can be significantly lowered during switching, while a high-energy barrier can be maintained after releasing the 'gating' chemical to ensure device stability. This kind of chemical 'gating' effect can be obtained by using ions with low energy barrier (for example, Li ions) to drive the charge–discharge redox reactions in the conduction channel in a battery-like fashion. In this case, switching can occur at a very low voltage (for example, 5 mV), resulting in excellent power efficiency[71].

Beyond synaptic behaviours, memristive systems can be used to implement neuronal elements that ultimately receive, process and transmit information in bioinspired computing systems. Neurons are primarily characterized as accumulating charge (as inputs from other neurons), and, after crossing a threshold, generating an action potential. Models of the neuron dynamics can vary widely in the level of biological fidelity. However, a critical ingredient to replicate neuron behaviour is active gain, whereby small input signals can — under the right circumstances — generate heavily amplified and dynamical outputs. Thus, a solid-state implementation of a neuron must meet some basic dynamical properties[72].

To accurately describe the dynamical physics in memristors that can realize 'neuronal' properties, one important parameter is the local temperature in the device. Temperature strongly influences the electronic (transport) and ionic (mobility) properties and may, in turn, be strongly influenced by them as well. As a simple example, when applying an increasing voltage sweep to a memristor, the rising Joule heating and local temperature activates the electronic transport, which further increases the Joule heating in a strong positive feedback mode. For some material systems, such as VO_2 or NbO_2, this process leads to an observed negative differential resistance (NDR), generating a strong but volatile change in the conductance. In fact, many forms of NDR can be ultimately described as a positive feedback-driven effect based on internal temperature coupled to the electronic transport[73,74]. Consequently, owing to the inherent positive feedback, only a small amount of input signal is needed to generate a large effect, thus supplying the needed neuronal amplification alluded to earlier.

The neuristor[75] is such an NDR-based circuit element that realizes many of the spiking behaviours of biological neurons, including signal gain, and a refractory period between spikes. It can be composed of two NDR devices (for example, NbO_2) with parallel capacitors to form complementary Pearson–Anson oscillators. A small input signal triggers the thermal runaway process described above, which leads to a temporary increase and then decrease in conductance of the system, similar to the opening and closing of an ion channel in a biological system. This process propagates a spike signal that can be coupled to other neuristors through (non-volatile) synaptic memristors described earlier. Alternative approaches have also utilized the frequency of oscillations directly to measure the weighted sum of input synaptic connections and replace the role of integrate-and-fire neurons[76]. Thus, these approaches allow the full realization of a purely memristive neuromorphic architecture.

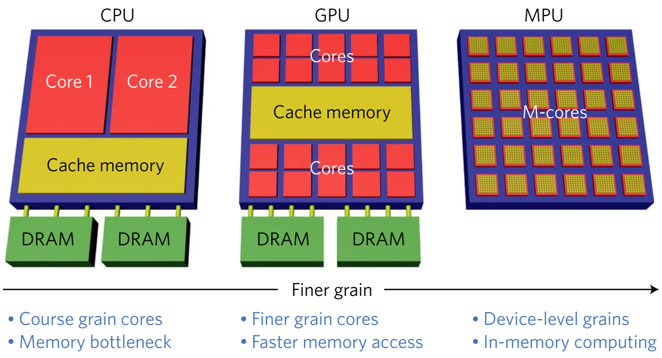

Fig. 4 | Possible evolution of the computing system. Starting from the conventional architecture with a separated processor and memory (central processing unit, CPU), graphics processing units (GPUs), with thousands of smaller cores and faster memory access, have become the workhorse of data-intensive computing tasks today. The proposed memory processing unit (MPU) architecture will continue this trend and will ultimately lead to full co-location of memory and logic at the smallest grain — the individual device level for efficient processing of a broad range of computing tasks.

In addition, recent work has shown that a single NbO_2 NDR element coupled to a parallel capacitor can undergo chaotic dynamics rather than purely periodic oscillations[77]. This deterministic chaos can be controlled through the input bias voltage, and was shown to derive from coupling to thermal fluctuations, again with positive feedback that leads to amplified effects. Moreover, it was shown that such a chaotic NDR element can be used to perform the thresholding function in a memristor-based Hopfield network. Such a network, where the weight matrix is implemented in a non-volatile memristor array, can solve combinatorial optimization problems such as the travelling salesman problem[77]. Such Hopfield networks are known to suffer from trapping in local minima, but the compact injection of chaotic dynamics can improve solution convergence, pointing the way towards a hardware accelerator for optimization problems. This can be related to similar concepts whereby stochasticity, rather than chaos, can be viewed as a computing resource exploited in biological systems[78]. Scaling up (including in 3D) of such a system is a currently unrealized opportunity to both explore the highly coupled dynamics that can emerge in such a dynamic network, and to better understand real biological networks.

Conclusions

Memristor-based architectures have shown great potential for developing future computing systems past the von Neumann and Moore's law era. Three possible implementations can be envisioned. In the short term, high-density, on-chip, non-volatile memories offered by memristors can significantly improve the performance of conventional von Neumann-based computing systems, and may find applications ranging from high-performance machine-learning systems to low-power embedded chips for the Internet of Things. Further advances in device technology and architecture developments may lead to large-scale implementation of memristor-based neuromorphic computing systems. Specifically, memristive crossbars provide a native solution to implement massively parallel and power-efficient vector-matrix operations that form the basis of neuromorphic operations. Moreover, carefully designed memristor devices can natively mimic the dynamics of their biological counterparts — synapses and neurons — and allow the network to develop complex emergent behaviours and possibly be used as model systems to test neuroscience hypotheses. Ultimately, we expect a memristor-based general-purpose, in-memory computing platform (Fig. 4). This efficient and reconfigurable computing platform, which we termed memory processing unit, can perform different tasks — data storage, arithmetic, logic and neuromorphic computing — using the same physical fabric that is programmable at the finest grain, the individual device level, without the need to move data outside the fabric. It can be argued that architectures such as the memristor-based memory processing unit is a natural evolution of the computing paradigm, following the same trend from central processing units to graphics processing units by moving towards finer-grained and highly parallel structures (Fig. 4).

We conclude by noting that biology has always served and will continue to serve as a great inspiration to develop methods for achieving lower-power and real-time learning systems. However, just as birds in nature may have inspired modern aeronautics technology, we eventually moved in new directions and capabilities for faster travel, larger carrying capacities and entirely different fuelling requirements. Similarly, in computing, modern application needs to go beyond those faced in nature, such as searching large databases, efficiently scheduling resources or solving highly coupled sets of differential equations. Interestingly, some of the observed characteristics in memristors may similarly provide 'beyond biology' opportunities in computing, taking advantage of the novel device dynamical behaviour and the network topology inspired by biology. In this regard, concepts such as the memory processing unit represent truly exciting opportunities down the road. To achieve these and other new computing systems of the future will require persistent and creative research that goes beyond any single discipline, and must include insights from neuroscience, physics, chemistry, computer science, and electrical and computer engineering, among others.

Received: 29 September 2017; Accepted: 22 November 2017;
Published online: 8 January 2018

References

1. Chua, L. O. & Kang, S. M. Memristive devices and systems. *Proc. IEEE* **64**, 209–223 (1976).
2. Strukov, D. B., Snider, G. S., Stewart, D. R. & Williams, R. S. The missing memristor found. *Nature* **453**, 80–83 (2008).
3. Lee, J. & Lu, W. D. On-demand reconfiguration of nanomaterials: when electronics meets ionics. *Adv. Mater*. https://doi.org/10.1002/adma.201702770 (2017).
4. Wong, H.-S. P. et al. Metal–oxide RRAM. *Proc. IEEE* **100**, 1951–1970 (2012).
5. Waser, R., Dittmann, R., Staikov, G. & Szot, K. Redox-based resistive switching memories — nanoionic mechanisms, prospects, and challenges. *Adv. Mater.* **21**, 2632–2663 (2009).
6. Govoreanu, B. et al. 10×10nm^2 Hf/HfOx crossbar resistive RAM with excellent performance, reliability and low-energy operation. In *2011 IEEE International Electron Devices Meeting (IEDM)* 31.6.1–31.6.4 (2011).
7. Yang, J. J., Strukov, D. B. & Stewart, D. Memristive devices for computing. *Nat. Nanotech* **8**, 13–24 (2013).
8. Torrezan, A. C., Strachan, J. P., Medeiros-Ribeiro, G. & Williams, R. S. Sub-nanosecond switching of a tantalum oxide memristor. *Nanotechnology* **22**, 485203 (2011).
9. Choi, B. J. et al. High-speed and low-energy nitride memristors. *Adv. Funct. Mater.* **26**, 5290–5296 (2016).
10. Kim, K.-H., Jo, S. H., Gaba, S. & Lu, W. D. Nanoscale resistive memory with intrinsic diode characteristics and long endurance. *Appl. Phys. Lett.* **96**, 053106 (2010).
11. Zhou, J. et al. Very low-programming-current RRAM with self-rectifying characteristics. *IEEE Electron Device Lett* **37**, 404–407 (2016).
12. Kim, S. et al. Experimental demonstration of a second-order memristor and its ability to biorealistically implement synaptic plasticity. *Nano Lett.* **15**, 2203–2211 (2015).
13. Jeong, Y., Kim, S. & Lu, W. Utilizing multiple state variables to improve the dynamic range of analog switching in a memristor. *Appl. Phys. Lett.* **107**, 173105 (2015).
14. Zidan, M. A. et al. Single-readout high-density memristor crossbar. *Sci. Rep.* **6**, 18863 (2016).
15. Yang, J. J. et al. Engineering nonlinearity into memristors for passive crossbar applications. *Appl. Phys. Lett.* **100**, 113501 (2012).
16. Zhou, J., Kim, K.-H. & Lu, W. D. Crossbar RRAM arrays: selector device requirements during read operation. *IEEE Trans. Electron Devices* **61**, 1369–1376 (2014).
17. Linn, E., Rosezin, R., Kügeler, C. & Waser, R. Complementary resistive switches for passive nanocrossbar memories. *Nat. Mater.* **9**, 403–406 (2010).
18. Hu, M. et al. Dot-product engine for neuromorphic computing: programming 1T1M crossbar to accelerate matrix-vector multiplication. In *2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC)* 1–6 (2016).
19. Hilson, G. IMEC, Panasonic Push Progress on ReRAM. https://www.eetimes.com/document.asp?doc_id=1327307 (2015).
20. Clarke, P. Crossbar ReRAM in Production at SMIC. https://www.eetimes.com/document.asp?doc_id=1331173 (2017).
21. Shen, W. C. et al. High-K metal gate contact RRAM (CRRAM) in pure 28nm CMOS logic process. In *2012 IEEE International Electron Devices Meeting (IEDM)* 31.6.1–31.6.4 (2012).
22. Fackenthal, R. et al. A 16Gb ReRAM with 200MB/s write and 1GB/s read in 27nm technology. In *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)* 338–339 (2014).
23. Yu, S. & Chen, P.-Y. Emerging memory technologies: recent trends and prospects. *IEEE Solid State Circuits Mag* **8**, 43–56 (2016).
24. Silver, D. et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484–489 (2016).
25. Jo, S. H. et al. Nanoscale memristor device as synapse in neuromorphic systems. *Nano Lett.* **10**, 1297–1301 (2010).
26. Neftci, E. O., Pedroni, B. U., Joshi, S., Al-Shedivat, M. & Cauwenberghs, G. Stochastic synapses enable efficient brain-inspired learning machines. *Front. Neurosci.* **10**, 241 (2016).
27. Yu, S. et al. Scaling-up resistive synaptic arrays for neuro-inspired architecture: challenges and prospect. In *2015 IEEE International Electron Devices Meeting (IEDM)* 17.3.1–17.3.4 (2015).
28. Alibart, F., Zamanidoost, E. & Strukov, D. B. Pattern classification by memristive crossbar circuits using ex situ and in situ training. *Nat. Commun.* **4**, 2072 (2013).
29. Milo, V. et al. Demonstration of hybrid CMOS/RRAM neural networks with spike time/rate-dependent plasticity. In *2016 IEEE International Electron Devices Meeting (IEDM)* 16.8.1–16.8.4 (2016).
30. Prezioso, M. et al. Training and operation of an integrated neuromorphic network based on metal-oxide memristors. *Nature* **521**, 61–64 (2015).
31. Sheridan, P. M. et al. Sparse coding with memristor networks. *Nat. Nanotech.* **12**, 784–789 (2017).
32. Choi, S., Shin, J. H., Lee, J., Sheridan, P. & Lu, W. D. Experimental demonstration of feature extraction and dimensionality reduction using memristor networks. *Nano Lett.* **17**, 3113–3118 (2017).
33. Burr, G. W. et al. Experimental demonstration and tolerancing of a large-scale neural network (165 000 synapses) using phase-change memory as the synaptic weight element. *IEEE Trans. Electron Devices* **62**, 3498–3507 (2015).
34. Linares-Barranco, B. & Serrano-Gotarredona, T. Memristance can explain spike-time dependent-plasticity in neural synapses. Preprint at http://precedings.nature.com/documents/3010/version/1 (2009).
35. Gupta, I. et al. Real-time encoding and compression of neuronal spikes by metal-oxide memristors. *Nat. Commun.* **7**, 12805 (2016).
36. Ambrogio, S. et al. Unsupervised learning by spike timing dependent plasticity in phase change memory (PCM) synapses. *Front. Neurosci.* **10**, 56 (2016).
37. Chua, L. O. & Yang, L. Cellular neural networks: theory. *IEEE Trans. Circuits Syst. I* **35**, 1257–1272 (1988).
38. Corinto, F., Ascoli, A., Kim, Y.-S. & Min, K.-S. in *Memristor Networks* (eds Adamatzky, A. & Chua, L.) 267–291 (Springer, New York, 2014).
39. Chi, P. et al. PRIME: a novel processing-in-memory architecture for neural network computation in ReRAM-based main memory. In *International Symposium on Computer Architecture (ISCA)* 27–39 (2016).
40. Sheri, A. M., Rafique, A., Pedrycz, W. & Jeon, M. Contrastive divergence for memristor-based restricted Boltzmann machine. *Eng. Appl. Artif. Intell.* **37**, 336–342 (2015).
41. Bojnordi, M. N. & Ipek, E. Memristive Boltzmann machine: a hardware accelerator for combinatorial optimization and deep learning. In *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)* 1–13 (2016).
42. Schuman, C. D. et al. A survey of neuromorphic computing and neural networks in hardware. Preprint at https://arxiv.org/abs/1705.06963 (2017).
43. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
44. Zidan, M. A. et al. Field-programmable crossbar array (FPCA) for reconfigurable computing. https://doi.org/10.1109/TMSCS.2017.2721160 (2017).
45. Borghetti, J. et al. 'Memristive' switches enable 'stateful' logic operations via material implication. *Nature* **464**, 873–876 (2010).
46. Yu, S. et al. Binary neural network with 16 Mb RRAM macro chip for classification and online training. In *2016 IEEE International Electron Devices Meeting (IEDM)* 16.2.1–16.2.4 (2016).

47. Kataeva, I., Merrikh-Bayat, F., Zamanidoost, E. & Strukov, D. Efficient training algorithms for neural networks based on memristive crossbar circuits. In *International Joint Conference on Neural Networks (IJCNN)* 1–8 (2015).
48. Liu, C., Hu, M., Strachan, J. P. & Li, H. H. Rescuing memristor-based neuromorphic design with high defects. In *2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC)* 1–6 (2017).
49. Shafiee, A. et al. ISAAC: a convolutional neural network accelerator with in-situ analog arithmetic in crossbars. In *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA)* 14–26 (2016).
50. International technology roadmap for semiconductors (ITRS); http://www.itrs2.net/itrs-reports.html
51. Borghetti, J. et al. A hybrid nanomemristor/transistor logic circuit capable of self-programming. *Proc. Natl Acad. Sci. USA* **106**, 1699–1703 (2009).
52. Kim, K.-H. et al. A functional hybrid memristor crossbar-array/CMOS system for data storage and neuromorphic applications. *Nano Lett.* **12**, 389–395 (2012).
53. Shulaker, M. M. et al. Three-dimensional integration of nanotechnologies for computing and data storage on a single chip. *Nature* **547**, 74–78 (2017).
54. Chen, H.-Y. et al. HfOx based vertical RRAM for cost-effective 3D cross-point architecture without cell selector. In *2012 IEEE International Electron Devices Meeting (IEDM)* 20.7.1–20.7.4 (2012).
55. Kanerva, P. Hyperdimensional computing: an introduction to computing in distributed representation with high-dimensional random vectors. *Cogn. Comput.* **1**, 139–159 (2009).
56. Li, H. et al. Hyperdimensional computing with 3D VRRAM in-memory kernels: device-architecture co-design for energy-efficient, error-resilient language recognition. In *2016 IEEE International Electron Devices Meeting (IEDM)* 16.1.1–16.1.4 (2016).
57. Li, H., Wu, T. F., Mitra, S. & Wong, H.-S. P. Resistive RAM-centric computing: design and modeling methodology. *IEEE Trans. Circuits Syst. I* **64**, 2263–2273 (2017).
58. Terabe, K., Hasegawa, T., Nakayama, T. & Aono, M. Quantized conductance atomic switch. *Nature* **433**, 47–50 (2005).
59. Lee, J., Du, C., Sun, K., Kioupakis, E. & Lu, W. D. Tuning ionic transport in memristive devices by graphene with engineered nanopores. *ACS Nano* **10**, 3571–3579 (2016).
60. Liu, Q. et al. Controllable growth of nanoscale conductive filaments in solid-electrolyte-based ReRAM by using a metal nanocrystal covered bottom electrode. *ACS Nano* **4**, 6162–6168 (2010).
61. Hou, Y. et al. Sub-10 nm low current resistive switching behavior in hafnium oxide stack. *Appl. Phys. Lett.* **108**, 123106 (2016).
62. Alibart, F., Gao, L., Hoskins, B. D. & Strukov, D. B. High precision tuning of state for memristive devices by adaptable variation-tolerant algorithm. *Nanotechnology* **23**, 075201 (2012).
63. Merced-Grafals, E. J., Dávila, N., Ge, N., Williams, R. S. & Strachan, J. P. Repeatable, accurate, and high speed multi-level programming of memristor 1T1R arrays for power efficient analog computing applications. *Nanotechnology* **27**, 365202 (2016).
64. Sheu, S.-S. et al. A 5ns fast write multi-level non-volatile 1 K bits RRAM memory with advance write scheme. In *2009 Symposium on VLSI Circuits* 82–83 (2009).
65. O'Connor, P. & Welling, M. Deep spiking networks. Preprint at https://arxiv.org/abs/1602.08323 (2016).
66. Shouval, H. Z., Bear, M. F. & Cooper, L. N. A unified model of NMDA receptor-dependent bidirectional synaptic plasticity. *Proc. Natl Acad. Sci. USA* **99**, 10831–10836 (2002).
67. Du, C., Ma, W., Chang, T., Sheridan, P. & Lu, W. D. Biorealistic implementation of synaptic functions with oxide memristors through internal ionic dynamics. *Adv. Funct. Mater.* **25**, 4290–4299 (2015).
68. Wang, Z. et al. Memristors with diffusive dynamics as synaptic emulators for neuromorphic computing. *Nat. Mater.* **16**, 101–108 (2017).
69. Martin, S. J., Grimwood, P. D. & Morris, R. G. M. Synaptic plasticity and memory: an evaluation of the hypothesis. *Annu. Rev. Neurosci.* **23**, 649–711 (2000).
70. Valov, I. & Lu, W. D. Nanoscale electrochemistry using dielectric thin films as solid electrolytes. *Nanoscale* **8**, 13828–13837 (2016).
71. Fuller, E. J. et al. Li-ion synaptic transistor for low power analog computing. *Adv. Mater.* **29**, 1604310 (2017).
72. Izhikevich, E. M. Simple model of spiking neurons. *IEEE Trans. Neural Netw. Learn. Syst* **14**, 1569–1572 (2003).
73. Funck, C. et al. Multidimensional simulation of threshold switching in NbO_2 based on an electric field triggered thermal runaway model. *Adv. Elect. Mater.* **2**, 1600169 (2016).
74. Gibson, G. et al. An accurate locally active memristor model for S-type negative differential resistance in NbOx. *Appl. Phys. Lett.* **108**, 023505 (2016).
75. Pickett, M. D., Medeiros-Ribeiro, G. & Williams, R. S. A scalable neuristor built with Mott memristors. *Nat. Mater.* **12**, 114–117 (2013).
76. Gao, L., Chen, P. Y. & Yu, S. NbOx based oscillation neuron for neuromorphic computing. *Appl. Phys. Lett.* **111**, 103503 (2017).
77. Kumar, S., Strachan, J. P. & Williams, R. S. Chaotic dynamics in nanoscale NbO_2 Mott memristors for analogue computing. *Nature* **548**, 318–321 (2017).
78. Maass, W. Noise as a resource for computation and learning in networks of spiking neurons. *Proc. IEEE* **102**, 860–880 (2014).

Acknowledgements

We acknowledge inspiring discussions with R. S. Williams, G. Astfalk, X. Zhu, W. Ma, F. Cai and Y. Jeong. This work was supported in part by the National Science Foundation (NSF) through grant CCF-1617315, by the Defense Advanced Research Program Agency (DARPA) through award HR0011-17-2-0018, and by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via contract number 2017-17013000002.

Author contributions

W.D.L conceived the project. All authors performed the project planning and comparative analysis. All authors carried out the discussions and the manuscript writing at all stages.

Competing interests

The authors declare no competing financial interests.

Additional information

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to J.P.S. or W.D.L.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.