Neuromorphic Circuits with Redox-Based Memristive Devices

Finn Zahari, Martin Ziegler, Pouya Doerwald, Christian Wenger, and Hermann Kohlstedt

Abstract This chapter addresses opportunities for memristive devices in the framework of neuromorphic computing. Memristive devices are two-terminal circuit elements, comprising resistance and memory functionality. This simple and likewise ingeniously concept allows beneficial applications in numerous neuromorphic circuits. However, the electrical characteristics as well as the materials and technological framework of memristive devices need an optimization for each specific application. The chapter starts with a short overview of basic principles of biological data processing followed by a taxonomy of different bio-inspired computing architectures, divided into time-dependent and time-independent concepts. Furthermore, the requirements on particular memristive device properties, such as I-V linearity, switching time, retention, number of states, time-dependency, and device variability, are discussed. The results of tangible examples of digital and analog memristive switching devices used in a deep neural network based on CMOS-integrated resistive random access memory devices (RRAMs) for chronic obstructive pulmonary disease (COPD) detection, in stochastic learning, in bio-inspired analog learning, and, finally, in oscillatory computing are presented and discussed.

F. Zahari (✉) · H. Kohlstedt
Nanoelectronics, Faculty of Engineering, Kiel University, 24143 Kiel, Germany
e-mail: fnz@tf.uni-kiel.de

H. Kohlstedt
e-mail: hko@tf.uni-kiel.de

M. Ziegler
Department of Electrical Engineering and Information Technology, Technische Universtität Ilmenau, 98693 Ilmenau, Germany
e-mail: martin.ziegler@tu-ilmenau.de

Institute of Micro and Nanotechnologies MacroNano, Technische Universtität Ilmenau, 98693 Ilmenau, Germany

P. Doerwald · C. Wenger
IHP Leibniz-Institut für innovative Mikroelektronik, Frankfurt/Oder, Germany
e-mail: wenger@ihp-microelectronics.com

M. Ziegler et al. (eds.), *Bio-Inspired Information Pathways*, Springer Series on Bio- and Neurosystems 16, https://doi.org/10.1007/978-3-031-36705-2_2

1 Introduction

While traditional von Neumann computing (binary and serial) continues to dominate the information and communication technology (ICT), recent groundbreaking innovations in alternative computing architectures and advanced electronic devices have become hard to ignore [1–6]. The mature silicon (Si)-based, complementary metal oxide semiconductor (CMOS) technology behind von Neumann computing led to tremendous progress in digital computing power over the last six decades. Moore's law, the prediction that the device integration density on a chip doubles every two to three years, and accompanied by scaling laws, served as secure guidelines for CMOS engineers [7, 8]. Today's Si-CMOS technology is characterized by impressive technical specifications. To mention but a view, the smallest feature size in advanced Si-CMOS transistor is in the order of 10 nm, several billion functional transistors are integrated into a single arithmetic logic unit (ALU), and a von Neumann computer is running at clock-frequencies of about 5 GHz [5]. Nonetheless, this success should not disguise the fact that Si-based von Neumann computing reaches its limits in the near future, and system performance progress has slowed down for a couple of years. For example, although a further shrinking of transistor dimension is still possible, limitations, such as quantum mechanical tunneling, will set fundamental constraints. The so-called memory gap appeared as another architecture-related show stopper. While the clock frequency increased with every new chip generation, the memory access time did not follow this trend. Therefore, the system performance is limited because the data handling between the ALU and memory presents a data bottleneck [9]. Aware of this stagnation, worldwide efforts by universities, research institutions, and industry focus on numerous novel computing architectures and advanced functional devices. Besides quantum computing, bio-inspired computing attracted considerable attention [10–15]. The term "bio-inspired" embraces various classes of computing architectures and hardware systems that take, to a more or less extent, biological information pathways as guidelines into account. In order to categorize such bio-inspired computing systems, basic principles of information pathways in nervous systems are summarized here. We want to emphasize, that the following introduction to the biological background describes the information processing in nervous systems in a very simplified way and represents only the tip of the iceberg. For more details on the intriguing biochemical and spatio-temporal mechanisms in this context, we refer the reader to the literature [16, 17]. A common way to explain information pathways in nervous systems is to distinguish the processing on the local neuron level from that of the entire system. Neurons are the information processing units in nervous systems. Neurons receive and process information in the form of action potentials. In Fig. 1a, the structure of a neuron, including the soma, dendrites, the axon, and connections to other neurons by synapses, is sketched [16]. An action potential (spike) is an abrupt transitory and transmitted change of the resting potential across the cell membrane. The amplitude of a spike is 100 mV, and its pulse duration is about 3.5 ms. Post-synaptic neurons receive action potentials (signals) from pre-synaptic neurons via dendrites and synapses. Those action potentials are

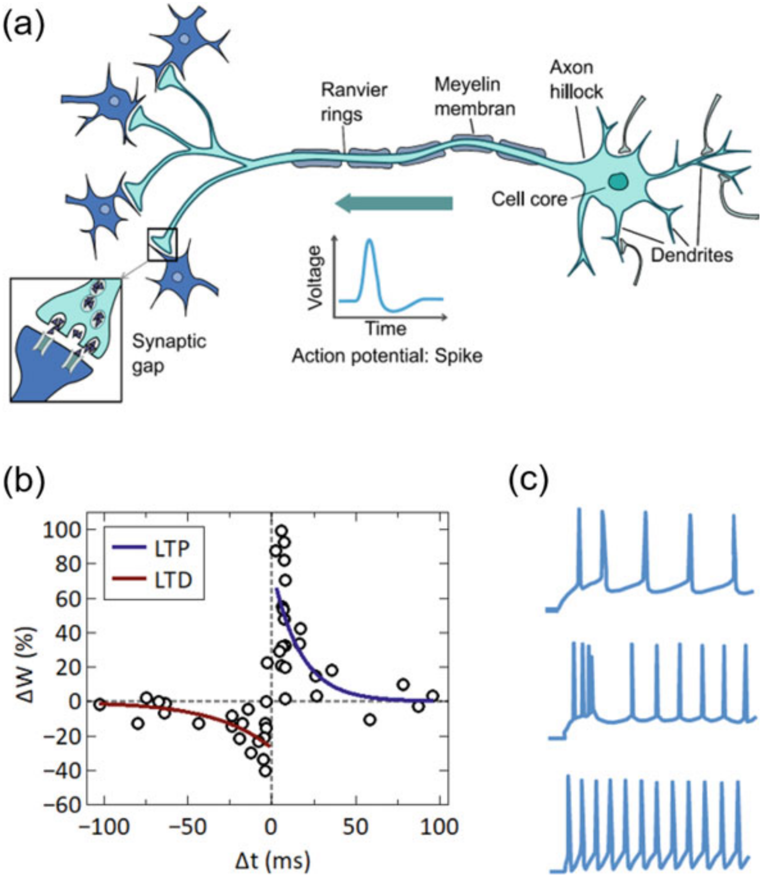

Fig. 1 **a** Blueprint of a neuron, including an enlarged sketch of a synapse and the illustration of a single action potential (i.e., a spike). Reprinted with permission of the corresponding author from [18]. **b** Change of synaptic weight (ΔW) in dependency on the timing ($\Delta t = t_{post} - t_{pre}$) of pre-synaptic spikes at time t_{pre} and post-synaptic spikes at time t_{post} illustrating spike-timing dependent plasticity (STDP). ΔW is measured 20-30 min after inducing the synaptic change with 60 correlated pre- and postsynaptic spikes with a frequency of 1 Hz at synapses between hippocampal glutamatergic neurons in culture. The data is fitted with exponential functions of the form $\Delta W = \mathrm{e}^{(-\Delta t/\tau)}$ for $\Delta t > 0$ (LTP) and $\Delta t < 0$ (LTD). Adapted from [19] [G.-q. Bi & M.-m. Poo, Annu. Rev. Neurosci. 24, 139–166 (2001)], with the permission of Annual Reviews. **c** Different irregular signal patterns, which occur in nervous systems [20]. ©2003 IEEE. Adapted, with permission, from [E. Izhikevich, IEEE Transactions on Neural Networks 14, 1569–1572 (2003)]

integrated within the post neuron's cell body. If the potential exceeds a threshold at the axon hillock, a new or a series of new spikes are generated at the axon hillock. Such a spike or spike trains are transmitted via the axon to subsequent neurons. The process is called leaky-integrate-and-fire (LIF), reflecting that the cell membrane is not a perfect insulator. The conduction velocity along the axon is between a few cm/s up to a few tenth m/s. We like to underline that the spike duration (3.5 ms) and the conduction velocity (e.g., 100 m/s) are six orders of magnitudes smaller than the pulse duration (1 ns) and signal transmission speed (roughly the speed of light in vacuum, $3 \cdot 10^8\ \mathrm{m/s}$) in modern processors, respectively. Facts, which indicate a different information processing of von Neumann computers and nervous systems. The mentioned interconnects are called synapses. Interestingly, the synaptic efficacies are not fixed and can change, which refers to the information coupling strength between the pre- and post-neuron. The efficacy can be increased (i.e., synaptic potentiation) or decreased (i.e., synaptic depression). This plastic behavior can last from milliseconds to minutes [called short-term potentiation (STP) or short-term depression (STD)] or from hours to years and up to the whole lifetime of animals [called long-term potentiation (LTP) or long-term depression (LTD)] [17]. The change of synaptic coupling strength depends on the common activity of the pre- and postsynaptic cells, e.g., in accordance to signal timing between the pre- and post-neuron. This mechanism is called spike-timing-dependet plasticity (STDP) [21]. STDP is closely related to Hebb's learning rule, which says that neurons that fire together wire together [22]. This rule describes the learning process in neural networks on the local cellular level by adjusting the synaptic efficacy dependent on the common activity of pre- and post-neurons. This further contributes to associative learning on the network level [23, 24]. Accordingly, STDP represents a local learning rule and is an essential process for establishing learning and memory in nervous systems [see Fig. 1b][19, 25]. In stark contrast to von Neumann computers, the spike generation in nervous systems is far from being regular. In dependency on the signal input of a neuron, a plethora of different firing rates (ranging from Poisson-like to bursting) are observed [26–28]. In Fig. 1c, examples of such irregular signal patterns are sketched [20]. Any signal transmission in nervous systems is accompanied by errors and noise [29]. For example, random potential fluctuations and the granular structure of neurotransmitters lead to a pronounced stochastic component in biological information pathways. Nevertheless, neural networks can store and retrieve information reliably. So noise is not a bug in biological systems, it is a feature [30, 31]. Nonetheless, although processes at the local neural level are highly stochastic, the entire nervous system exhibits rhythmic brain waves. In the human brain, they appear in the form of alpha, beta, and gamma waves [32]. Characteristic features, such as STDP [21, 25, 33], stochastic firing and bursting of neurons in the hundred Hz range, recurrent network structures, and aspects of oscillatory synchrony in larger neuronal ensembles [34–43] are essential ingredients in biological-based information processing. Moreover, factors related to the close interaction of a nervous system with its environment, i.e., external stimuli, are of crucial importance [44]. Therefore, neuronal design principles provide a model for neuromorphic systems, which are diametric to development strategies in present binary ICT, including precise GHz clock frequencies, near-light-speed signal transmission, and clearly separated logic and memory [45, 46]. In contrast to clock-driven von Neumann machines [9, 47], information processing in biological nervous systems is characterized by highly parallel, energy efficient, and adaptive architectures [10, 11, 14, 48].

Now we turn back to the field of neuromorphic circuits [10–15]. Bio-inspired computing aims to realize biological plausible information pathways (a few are mentioned above) in engineered systems. However, this goal immediately leads to numerous questions and challenges: Which of the manifold and intriguing information pathways observed in biology need to be mimicked by neuromorphic circuits to establish novel computing architecture with superior properties to conventional von Neumann systems? Could any biological process be simulated by a von Neumann computer to achieve similar performances as the biological model? Should novel neuromorphic circuits be made on today's mature Si-CMOS platform or might it be wise to open the material "tools box" apart from Si technology in order to facilitate the integration of novel devices and/or to incorporate self-assembly strategies similar to those observed in nature?

Might it be beneficial to design circuits working at biologically plausible time scales, i.e., with pulse durations of about a few ms and signal conduction velocities of about m/s, which lead to strong signal retardations? Finally, yet importantly, the consideration of stochasticity could be an additional design goal in accordance with its fundamental role in biology. These questions display only a small selection of possible approaches to mimic biological information pathways in engineered systems. This chapter focuses on neuromorphic circuits that take up those hallmarks of biological information processing that have been discussed above, i.e., STDP, stochasticity, oscillatory computing, and so on. The circuits compromise redox-based memristive devices as key components. Memristive devices consist of capacitor-like layer sequence, i.e., metal-memristive material-metal stacks. A universal property of the memristive devices is that the memristive state depends on previously induced charge flows, applied currents, or applied electric fields, thus storing any resulting resistance state. Memristive devices can be engineered to exhibit either binary switching or an analog resistance variation. Both device categories are subjects of this chapter and show beneficial features in pattern recognition and oscillatory computing. For details concerning resistive switching and the underlying physical-chemical mechanisms, we refer the reader to Chap. 3 and the overwhelming literature on the subject [49–52].

2 Requirements for Memristive Devices for Neuromorphic Computing

Memristive devices are being explored for many different types of neuromorphic computation schemes, where their non-volatility allows computation to be performed in memory [53]. In this respect, memristive devices allow overcoming the von Neumann bottleneck, where memory and computation are separated from each other

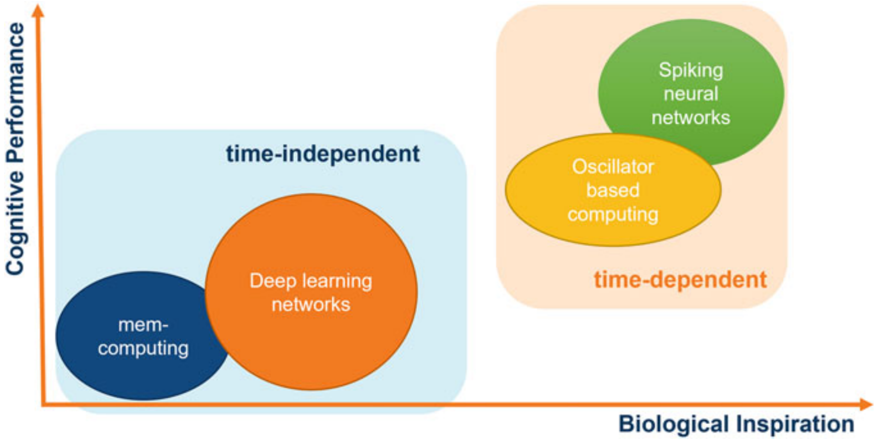

Fig. 2 Classification of different biologically inspired computing architectures within time-dependent and time-independent computational schemes: the more biological the computational scheme, the higher the expected cognitive performance

[9]. However, there are many different neuromorphic computing architectures, which place different requirements on memristive devices [54, 55]. Figure 2 shows a classification that relates different network architectures to their biological inspiration and cognitive performance. While time-independent computing schemes are most widely used, time-dependent computing schemes are more biologically realistic. However, their technical implementation is much more challenging. While oscillatory computing and spiking neural networks (SNNs) taking the temporal dimension of the signals into account [14, 56–58], mem-computing, and deep neural networks (DNNs) working in a time-independent way using synchronized signals (i.e., they are based on clock signals) [57, 58]. This section will show the differences in the requirements for the memristive devices. For this purpose, a short overview of the architectures will be given, and the requirements they impose on memristive devices will be elaborated. Concrete examples of memristive networks that use time-independent computational schemes are discussed in Sect. 3, while examples of time-dependent neuromorphic architectures are presented in Sect. 4.

Mem-Computing

Mem-computing was originally invented for non-bio-inspired applications which aim to implement new computing architectures tasks like stateful [59, 60] and non-stateful logic [61, 62], in-memory arithmetic operations [63], solving linear and partial differential equations [64–66], optimization [67–69], and signal processing [70, 71] (for a review, the reader is referred to [3, 55, 58]). In all of those applications, memristive devices are required which own a fast and low power switching, high cycling endurance (e.g., $> 10^{12}$ cycles for stateful logic) as well as a low device-to-device and cycle-to-cycle variability [55, 58]. However, the inherent randomness of memristive switching mechanisms are challenging for high-precision computing tasks and different techniques are needed to increase the overall precision of the devices [64, 65, 69].

Deep Neural Networks (DNNs)

DNNs [72] (including convolutional neural networks (CNN) [73]) are bio-inspired computing schemes that benefit from the in-memory computing architectures incorporating memristive devices. In these networks, a large number of artificial neurons arranged in layers connected in a feed-forward structure by adjustable weights. These analog weights are trained with the backpropagation algorithm, which implements the delta rule between each neural layer [72]. These networks are specialized in pattern recognition tasks and build the backbone of today's machine learning applications [3, 58]. However, several drawbacks come along with DNNs. The networks are usually set up in software running on traditional von Neuman architecture, i.e., mostly on general-purpose graphics processing units (GPUs) [74] or tensor processing units (TPUs) [75]. Since a huge amount of data is needed for training and many learning cycles are required for real applications, these networks consume a large amount of energy and space. In addition, the training is very time-consuming [3, 57]. In this context, memristive devices can provide a solution as their in-memory computing properties enable parallelization of processes, reducing power consumption and training time by orders of magnitude [76–79]. The significantly increased efficiency lies in implementing matrix-vector multiplications (MVMs) in hardware utilizing Ohm's law and Kirchoff's current law [54]. For this, however, strict requirements must be met by the memristive devices to be used as artificial synapses in DNNs, which are summarized in Fig. 3. A linear, gradual, and symmetric change in resistance is required for training [77, 80, 81]. For example, it has been shown that a 2% deviation from perfect symmetry increases the required number of analog states to train a DNN from 100 to 1000 [82]. In addition, endurance is essential since a lot of data is needed for training, combined with high energy efficiency and low latency to enable training directly on edge devices [83]. On the other hand, the training algorithm can compensate for device variability and yield to a certain degree [82]. Transferring pre-trained weights to memristive devices, furthermore, leads to less strict requirements the devices have to fulfill. In this respect, it has been shown that a resolution of four to eight bits is sufficient to compete with floating-point precision weights in the inference process [75, 84]. For this purpose, multiple binary devices can be combined to mimic the weights of a synapse [85, 86]. However, the device variability and yield are more critical for pre-trained networks, and endurance becomes less crucial [82]. Retention is also of particular relevance. While for training, short-term

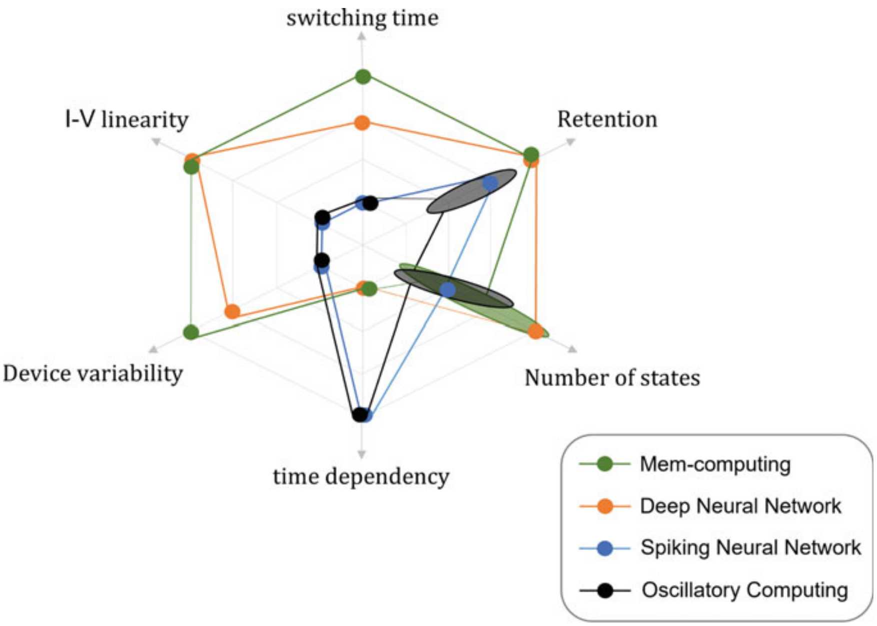

Fig. 3 Comparison of the requirements memristive devices have to fulfill to be suitable for different computational architectures

retention and stability are sufficient [78, 82, 87], for inference, long-term retention and stability must be given [78, 82, 87].

Spiking Neural Networks (SNNs)

In SNNs, the computation is based on synaptic connectivity and asynchronous, event-driven, and temporally precise signals [14, 15, 56–58, 88]. This enables such networks to adapt to changing environmental conditions and react accordingly. However, adequate local learning algorithms are required to exploit those advantages of SNNs over DNNs. These algorithms must satisfy the special needs of memristive devices and neuromorphic network structures in equal measure [89–97]. At the memristive device level, several criteria, and in some cases different requirements from DNNs, must be met in order to satisfy the bio-inspired learning rules designed for SNNs. Figure 3 summarizes these and compares them to the device requirements of DNNs. Since nonlinear time-dependent conductivity modulations are the basis of learning in SNNs [57, 58], a gradual but nonlinear resistance change in memristive devices can be beneficial [57, 97]. Furthermore, both the endurance and energy-efficient switching and read-out are crucial properties for memristive devices in SNNs [57, 83]. However, device-to-device and cycle-to-cycle variability [57] and yield are less critical since the effect of defective devices can be mitigated during learning [93]. Furthermore, the switching speed is expected to be less important [57, 88], especially if the networks run on a biologically relevant time scale with a spike duration of milliseconds and frequencies of a few Hz [33]. The performance of SNNs is often investigated with pattern recognition tasks with frame-based datasets [3, 98]. In this respect, typical benchmark datasets containing static images are MNIST [99], CIFAR [100], and ImageNet [101]. These datasets do not contain temporal information. Thus, they do not allow to show the full potential of the time-dependent SNNs [102] and an outperformance in comparison to DNNs has not been reported so far [58]. More suitable benchmarks for SNNs should contain real-world spatio-temporal data, e.g., collected with event-based sensors [98, 103]. Thus, instead of executing pattern recognition on static data, SNNs are expected to be superior in interacting with the real world in a dynamically changing environment by processing continuous but sparse input streams on an energy-efficient way [57, 98, 102]. More suitable benchmarks for those tasks are dealing, e.g., with hand and arm gesture detection (DvsGesture) [104], automated driving [105, 106], or robotics [107]. Moreover, generally applicable learning algorithms and network structures, which can cope with several different tasks, are within the focus of research [3, 15].

Oscillatory Computing

Nature uses time-coherent dynamics for information processing based on the formation of context-dependent, self-organized, and transient network structures. These enable us to react adequately to changing environmental conditions. Furthermore, these self-organized network structures are an important property for sensory integration [36, 108]. Even if the underlying mechanisms are only partially understood, the interaction between dynamics and topology has been identified as one of the essential building blocks of information processing in the brain in recent years [109]. In the current understanding, it is assumed that information is encoded into coherent states by temporally correlated neural activity patterns [110]. This concept offers, particularly, an elegant explanation for the binding problem - the question of the mechanism of sensory integration, which allows our brain to construct uniform perceptions from the multitude of sensory information. First evidence of these concepts could have been gathered from experiments with sensorimotor networks [111]. More recent studies have shown the universality of these concepts for the entire brain [110]. In this respect, memristive devices allow a new degree of freedom for the concept of neural synchrony: a local memory that supports a transient connectivity pattern [112–114]. The requirements for memristive devices needed for this have been less studied and are the content of Sect. 4.2. However, the inherent stochasticity of memristive devices has been shown to be helpful for this application [113].

3 Time-Independent Neural Networks

This section deals with two memristive networks that use time-independent computational schemes, both relying on a fully CMOS-integrated 4 kbit resistive random-access memory (RRAM) array [115–117]. The first network discussed in Sect. 3.1 is a mixed-signal-circuit implementation of a DNN for the detection of chronic obstructive pulmonary disease (COPD) [118]. Here, the devices are used to store the pre-trained weights of the DNN while the neurons are implemented in software. In that way, the possibility for on-chip recognition of saliva samples using in-memory techniques to detect COPD in a Point-of-Care application is shown. The second network introduced in Sect. 3.2 exploits the RRAM cells' inherent stochasticity to solve a pattern recognition task [119, 120]. The stochastic artificial neural network (StochANN) is able to learn a subset of the MNIST benchmark through the adaptation of synaptic weights directly in hardware (in a mixed-signal realization) through a supervised local stochastic learning rule. Additional simulations of StochANNs with a larger number of devices show the performance limits of such a network for the whole MNIST benchmark. The results are compared to state-of-the-art approaches using time-independent and time-dependent networks.

3.1 Deep Neural Network Implemented in CMOS-Integrated RRAM Arrays Used for Chronic Obstructive Pulmonary Disease Detection

In this section, a time-independent DNN trained for disease detection is introduced [118]. The three-layer network is composed of a binary, fully CMOS-integrated 4 kbit RRAM array [115–117] emulating the synaptic weights, while neurons are implemented in software. The network was trained entirely in software, and the weights were subsequently transferred to the RRAM array. In that way, memristive devices can be used for inference, while they do not have to fulfill the same requirements as needed for training (see Sect. 2). In the following, first, the disease to be detected and the relevant input parameters for the machine learning (ML) method are introduced. Afterward, the network implementation and its performance are described.

COPD, one of the most prevalent lung diseases worldwide, runs a perfidious course with an often long-lasting undiagnosed initial phase. Clinical treatment approaches for COPD result in repeated clinical visits and extended hospitalization for patients. This fact, apart from being an economic burden for healthcare infrastructures, drastically impacts patients' life quality. To address this issue, today's healthcare systems have encouraged the development of personalized solutions through which patients can receive appropriate medical assistance in an outpatient clinic or a home-care environment [121]. Recent advances in point-of-care medical devices have facilitated the early detection, prevention, and treatment of various diseases [122]. However, without analytical insight, collected data from medical sensors are merely raw data with low clinical value. For instance, in our previous work, a portable biosensor for managing COPD in home-care environments was presented [123]. The developed biosensor was capable of characterizing the viscosity of saliva samples for diagnostic purposes. However, saliva samples' viscosity properties are one parameter of various parameters required for COPD detection. As a result, upon viscosity measurements by the developed biosensor, a sophisticated diagnostic algorithm is required to detect COPD by concurrent consideration of all essential parameters related to a patient's personal and medical background. These demographic parameters include, but are not limited to, age, gender, weight, cytokine level, pathogen load, and the smoking background of subjects. Therefore, machine learning tools, or more specifically pattern recognition methods, could make the diagnostic procedure more efficient by converting collected data from medical sensors into meaningful clinical information. Moreover, machine learning can be used for identifying diagnostic links between symptoms and diseases that have been previously unknown and providing treatment plans and recommendations to healthcare specialists.

As a result, implementing ML tools is crucial for converting collected raw data from subjects into meaningful clinical-diagnostic information. Furthermore, advanced ML analytics could make the management of COPD in Point-of-Care applications more efficient. Nevertheless, drawbacks of cloud-based ML techniques for medical applications such as data safety, immerse energy consumption, and enormous computation requirements need to be addressed for this application. To address these challenges, CMOS-integrated RRAM arrays can be used for the hardware-based implementation of ML methods. Therefore, a memristive neuromorphic platform is presented in this work for on-chip recognition of saliva samples of COPD patients and healthy controls. Two groups of saliva samples, 160 for Healthy Controls (HC) and 79 for COPD patients, were collected in the frame of a joint research project at the Research Center Borstel, BioMaterialBank Nord (Borstel, Germany) [124, 125]. Patient materials were collected and anonymized prior to accessibility. The sampling procedure of the saliva samples was approved by the local ethics committee of the University of Luebeck under the approval number AZ-16-167. Figure 4 demonstrates a hierarchy chart, categorizing the collected saliva samples into extended subgroups with respect to their diagnosis, gender, and smoking status [124]. As shown in Fig. 5, analog values of these four attributes were converted into 23 binary bits [gender (1), smoking status (3), age (9), dielectric permittivity (10)]. Dielectric sensors could be used to characterize sputum samples collected from patients for early diagnosis of COPD. The CMOS-based dielectric sensor system used for the real-time monitoring of sputum samples is described in [126].

The neuromorphic hardware implementation [118] of the developed ANN model was performed with a 4-kbit array of CMOS-integrated RRAM devices based on amorphous HfO_2 developed by IHP [115–117]. The array consists of 64×64 memristive cells in a 1-Transistor-1-Resistor (1T-1R) configuration. The two distinct states, low resistance state (LRS) and high resistance state (HRS), were used for the implementation. The mean read-out currents are 30.8 mA and 3.2 mA at 0.2 V for LRS and HRS, respectively. For the deployment of the 10-level model, a mixed-signal

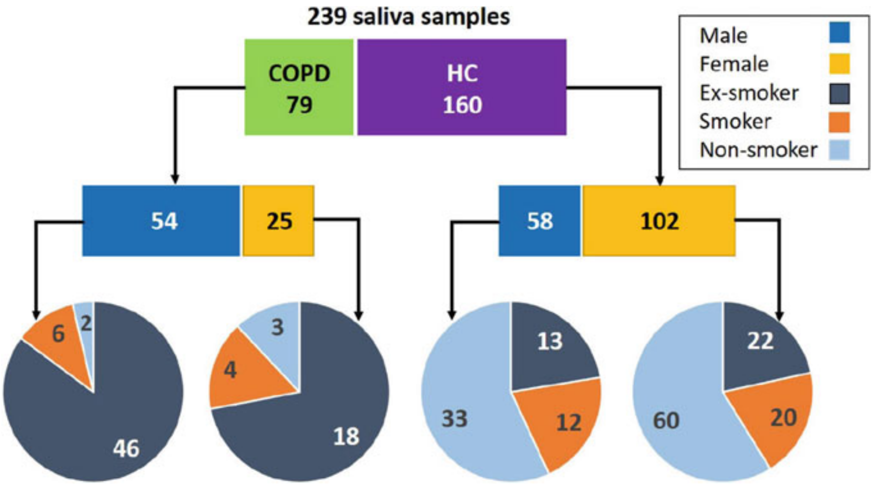

Fig. 4 Hierarchical categorization of collected saliva samples into extended subgroups with respect to their diagnosis, gender, and smoking status. Reprinted from [124] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

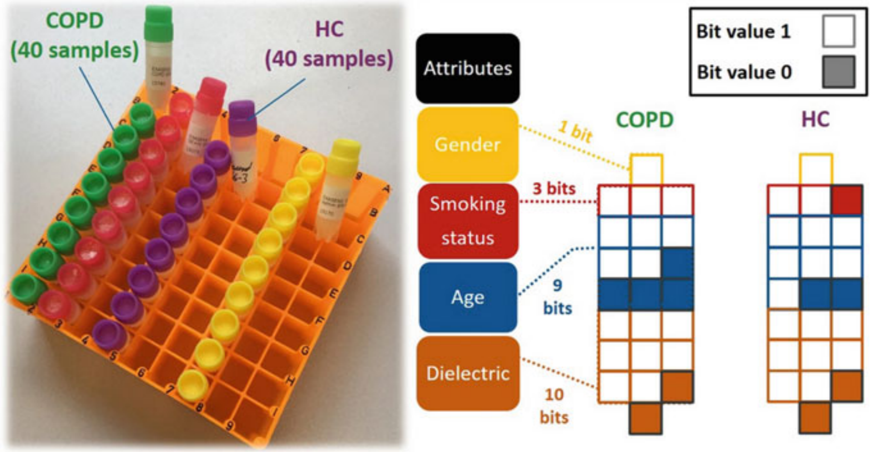

Fig. 5 Conversion of analog attributes of the dataset (gender, smoking status, age, and dielectric properties) into 23 binary bits. Reprinted from [118] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

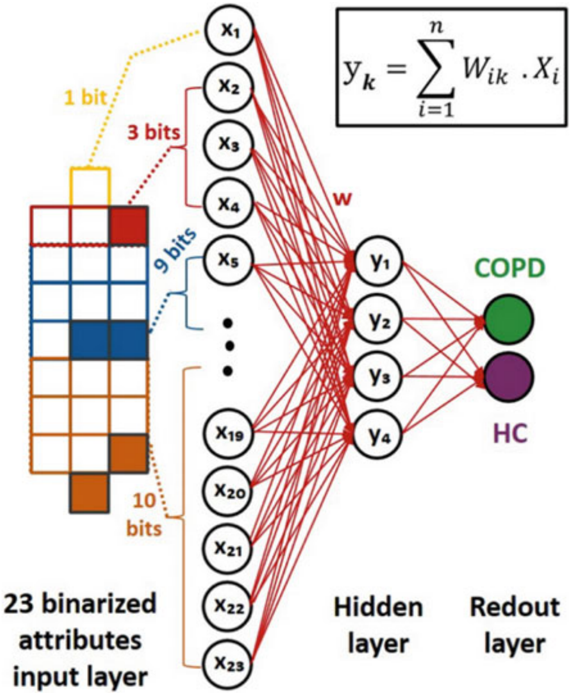

Fig. 6 ANN topology with one hidden layer for the classification of saliva samples of COPD patients and HC. Reprinted from [118] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

neuromorphic circuit with software-based neurons and hardware synapses was used [119, 120]. Considering the topology of the ANN model (see Fig. 6) with one hidden layer and one read-out layer with four and two neurons per layer, respectively, 106 parameters (i.e., synaptic weights and biases) were required for connecting the network layers. The resistance states of 1060 memristive devices on a single chip were set to the HRS or LRS, respective to the pre-trained weights. Every network parameter is represented by the combination of ten devices where five devices represent positive values and five devices represent negative values, respectively. The sum of ten read-out currents at 0.2 V represents the total value of one synaptic weight. After successfully implementing pre-trained weights on the hardware, the test subset of data was used to evaluate the performance of the neuromorphic model for the recognition of COPD and HC samples. In order to recognize the COPD samples with the mixed-signal approach, the 23 input bits of the test-subset data were applied to the simulated neurons within the input layer. The output neurons of every subarray are perceptrons with a sigmoidal activation function, which receive the sum of current values passing through the connected devices together with a specific bias value. These current values are normalized to the maximum value of the pre-trained analog network to guarantee that the sigmoid function is activated with a reasonable range of values. The output values of the third layer (read-out layer) perceptrons denote whether a test sample belongs to COPD or HC categories. This hardware realization agrees with the theory of neural networks that the weighted sum of inputs determines the value of a perceptron in the subsequent layer, as illustrated in Fig. 6.

In summary, the concept of on-chip recognition of saliva samples of COPD patients using a memristive neuromorphic platform was studied. A hardware-friendly artificial neural network model was developed and trained for classifying COPD and HC samples using real clinical data. Subsequently, a 10-level conversion of the trained classification model was transferred onto a memristive neuromorphic platform for the on-chip recognition. The memristive chip provided a remarkable accuracy of 89%, offering an alternative approach to cloud-based methods required for diagnosing COPD in Point-of-Care applications.

3.2 Stochastic Learning with Binary CMOS-Integrated RRAM Devices

The inherent stochastic nature of the filament formation and dissolution in RRAM devices is challenging for many applications, especially if a high numerical precision is needed (see Sect. 2). On the other hand, different approaches benefit from the randomness of resistive switching and exploit it explicitly for the technical emulation of biological information processing. Such networks include noise tolerant stochastic computing technologies [127], synchronization of oscillatory neurons to emulate neuronal coherence [113, 128] as described in Sect. 4.2, stochastic switching neurons [129, 130] and stochastic learning rules realized with single binary synapses [119, 120, 131–133], as well as compounds of several binary devices as one synapse [85, 129, 131, 133].

The stochastic learning algorithm [119, 120] described in this section utilizes the stochastic nature of binary fully CMOS-integrated 4 kbit RRAM arrays [115–117] in a 1-transistor-1-resistor (1T-1R) configuration, the same technology as used in Sect. 3.1, in a stochastic artificial neural network (StochANN) to learn the MNIST benchmark [99]. In that way, it is shown that the proposed StochANN is able to process analog information with binary memory cells. The devices are composed of a HfO_{2-x}/TiO_{2-y} bi-layer sandwiched between TiN electrodes. They can be switched between two distinct resistance levels, i.e., HRS and LRS. As an initial step, an electro-forming process is required. This is reliably done by the incremental step pulse with verify algorithm (ISPVA) [134]. The electrical device properties depend on the crystalline phase of the HfO_{2-x} [120, 135]. The switching probability for polycrystalline and amorphous HfO_{2-x} films is shown in Fig. 7a, b and c, d, respectively. The device-to-device (D2D) variability of 128 1T-1R devices is therefore determined by applying single voltage pulses in the set [Fig. 7a, c] and reset [Fig. 7b, d] regime, i.e., for the transition from HRS to LRS and vice versa. To obtain the measured data shown as dots in Fig. 7, resistance states were measured with a read-out voltage of 0.2 V after applying a positive or negative voltage pulse for set and reset transition, respectively. A current of 20 mA has to be exceeded to count

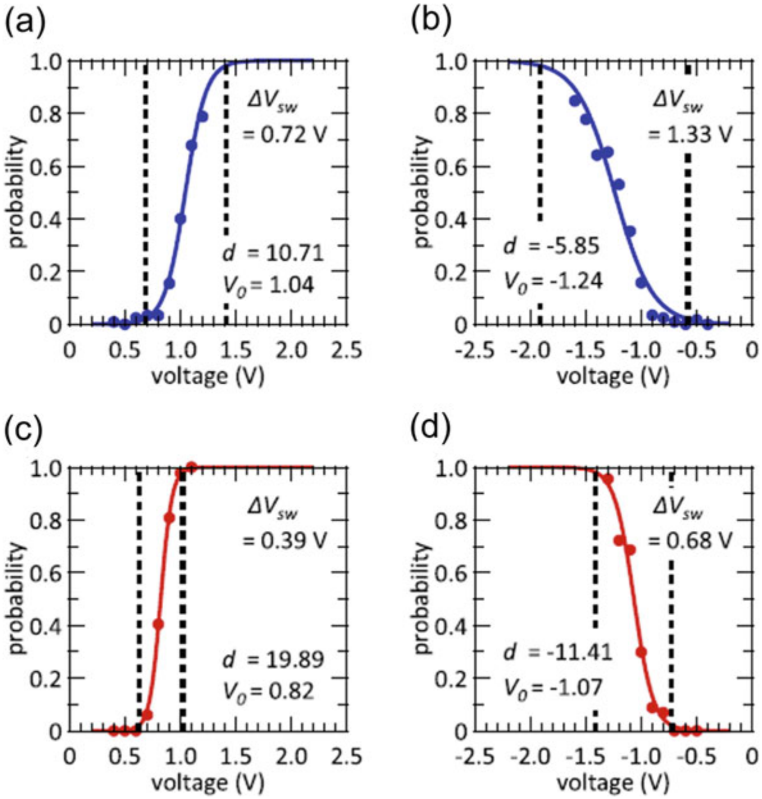

Fig. 7 Switching probability of the used RRAM devices dependent on the applied voltage amplitude of 10 ms pulses. Dots represent measured data, while the solid lines are fits with Eq. 1. The fit parameters d and V_0, as well as the switching window ΔV_{sw} are also depicted. In **a**, **b**, the set and reset behavior of the polycrystalline devices are shown, respectively, while in **c**, **d**, the set and reset behavior of the amorphous devices are depicted, respectively. For each technology, 128 devices were measured. Reprinted from [120] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

as an effective set operation, while the current has to be lower than 5 mA to count as a successful reset process. All pulses had a length of 10 ms. The cycle-to-cycle (C2C) variability shows no significant deviation from the D2D variability in similar devices [136]. Furthermore, the switching voltages determined here do not depend on the devices' position within the 4 kbit array. Thus, taking the D2D variability into account for designing the learning rule is reasonable. The switching probability dependence on an applied voltage pulse can be described by a Poisson distribution taking voltage amplitude and pulse width into account [113, 119]. The distribution function for N voltage pulses (neural activity level) with a voltage amplitude V can be expressed as [113, 120]

$$f_N = \frac{1}{1 + e^{-d(V - V_o)}}. \tag{1}$$

Here, V_0 denotes the voltage at which the probability f_N is equal to 0.5, and d is a measure of the distribution functions slope and, therefore, of the switching variability. The larger the absolute value of d, the smaller the switching window ΔV_{sw} in which a stochastic encoding of analog data is possible. The switching window is defined as the voltage interval in which the switching probability f_N is between 2 and 98 %. Fitting the measured D2D variability with Eq. 1 leads to the solid line in Fig. 7 as well as to the depicted d and ΔV_{sw} values. In summary, ΔV_{sw} is smaller for amorphous HfO_{2-x} than for polycrystalline HfO_{2-x} devices due to the grain boundaries' impact on the D2D variability and a more homogeneous defect distribution in amorphous hafnia films [135, 137]. Furthermore, ΔV_{sw} is smaller in the set transition compared to the reset transition for both technologies.

To emulate synaptic plasticity, the activity A of a neuron is encoded in voltage pulses with amplitudes V within the switching windows:

$$V = V_1 + A \cdot \Delta V, \tag{2}$$

with

$$A = \frac{N}{\Delta t}. \tag{3}$$

Here, N is the number of action potentials arriving at a neuron in the time interval Δt, and V_1 is the lower bound of the switching window. Exploiting the whole switching window to map analog activity, i.e., analog data, to the stochastic nature of the switching event is possible by a proper choice of ΔV. In the following, the StochANN utilizing this local learning rule is described, and the influence of the switching window size on the learning performance is shown.

The MNIST benchmark [99] of static visual patterns is used within this work. The learning data set contains 60,000 images of handwritten digits from 250 different writers. Each image consists of $28 \cdot 28$ greyscale pixels with 256-levels. Some representations of the ten included patterns (i.e., digits from zero to nine) are shown in Fig. 8. A test data set contains additional 10,000 images, which can be used to determine classification accuracy. For the StochANN [119, 120] described in this section, averaged images are used, as shown in Fig. 9. These are obtained by calculating the average greyscale values of 100 randomly chosen representations of each pattern. For learning, the pixel intensities of every image are mapped to the interval [0,1] by dividing the values of every pixel by the maximum pixel value of the respective image. Learning takes place in a supervised manner in a time-independent StochANN. The network topology is illustrated in Fig. 9. Each pixel has one corresponding input neuron connected to every of the output neurons (one for each pattern sketched here) in a two-layer feed-forward configuration similar to earlier approaches [89–93]. Each connection is made by a binary stochastic synapse. The input neurons map the pixel

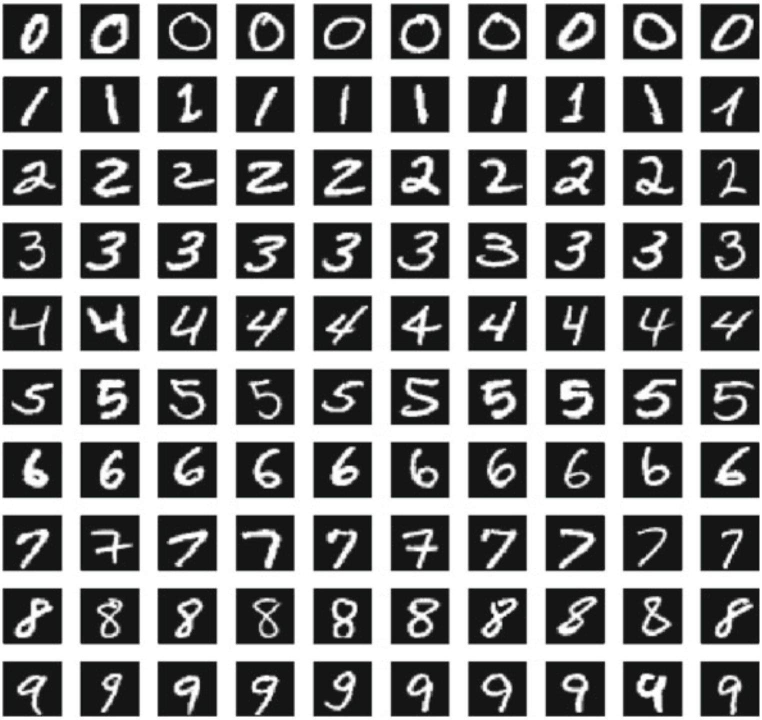

Fig. 8 Examples of MNIST images. Each row shows different representations of one pattern

intensities into voltage pulses, inducing a switching event with respective probability for the connections to the dedicated output neuron. In that way, the trained synaptic connections, i.e., memristive devices, form receptive fields of the output neurons. Either set or reset transitions of both technologies are used for learning. If the set transition is used for learning, reset pulses are applied to each synaptic device prior to execute the learning rule corresponding to a low probability p_{sat}. Accordingly, a low probability set pulse is used if the reset transition is exploited for learning. Thus, saturation effects are avoided. In that way, all training images can be used several times to train the network. It should be noted that the number of output neurons can vary for two reasons. First, if only a subset of the patterns is learned, fewer output neurons are necessary (one for each pattern). Second, each pattern can be learned by several output neurons to increase the network performance. The StochANN performance is evaluated experimentally with a mixed-signal circuit emulating the synapses in hardware using the fully CMOS-integrated RRAM arrays and neurons in software. Details about the circuit design can be found in [120]. Moreover, the network is simulated without taking device variabilities and imperfections into account. Here, the stochastic learning rule is simulated by generating a random number $r_{i,j}$ uniformly distributed over the interval (0, 1) for every pixel j of every learning image i and

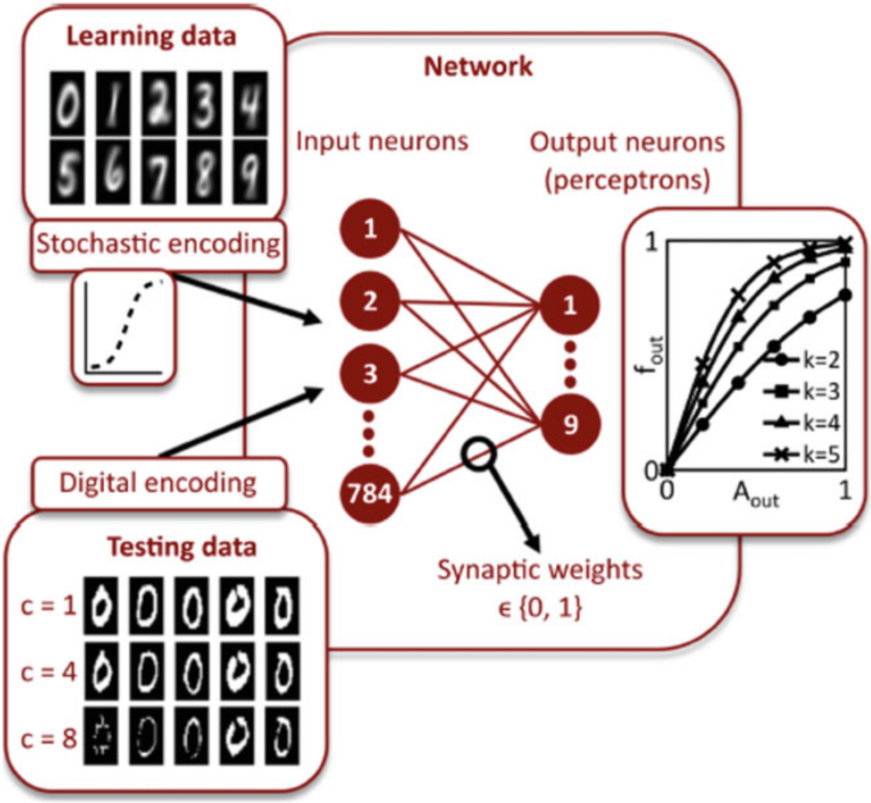

Fig. 9 Schematic visualization of the network structure (center) with 784 input neurons and 10 output neurons connected with memristive devices as binary synaptic weights. The learning data (left top) and the test data (left bottom) are sketched as well as the output neurons activation function (right). Reprinted from [120] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

the respective synaptic weight $w_{i,j}$ is set to 1 if the pixel intensity is larger than $r_{i,j}$. The simulations serve to determine the maximum possible network performance by omitting device imperfections. Furthermore, the number of available devices limits the experimentally realized network size. Thus, larger networks can be simulated to compare the stochastic learning algorithm to state-of-art networks. After learning, the classification accuracy can be evaluated by applying test images to the network. In experiments, 50 randomly chosen images of each pattern are used, while all 10,000 test images are used in simulations. Therefore, the pixel intensities are binarized. For each test image, a threshold value proportional to a global constant c and the mean pixel intensity of that image is determined. Pixel intensities, which do not exceed the threshold, are set to 0, and the others are set to 1. The larger c, the fewer pixels are active, as shown in Fig. 9. Every test image is shown once to the network by the input neurons, which induce a 0.2 V read-out pulse for pixel values of 1. The output neurons are modeled as perceptrons with an activation function

$$f_{out} = \frac{1 - e^{-k \cdot A_{out,i,m}}}{1 + e^{-k \cdot A_{out,i,m}}}. \tag{4}$$

Here, k is a positive constant that defines the slope, as shown in Fig. 9 and $A_{out,i,m}$ is the normalized activity of the input neurons for the test image i weighted by the synaptic connections $w_{j,m}$ of input neurons j to the output neuron m according to

$$A_{out,i,m} = \frac{1}{784} \cdot \sum_{j=1}^{784} p_{i,j,bin} \cdot w_{j,m}, \tag{5}$$

where $p_{i,j,bin}$ is the binarized intensity value of pixel j being part of image i. The weights $w_{j,m}$ are determined as logical 1 if the read current exceeds 10 mA and the set transition is used for learning. If the reset transition is used, $w_{j,m}$ is assigned a logical 0 if the current is larger than 10 mA. Thus, Eq. 5 is valid for both cases. The output neuron, which receptive field re-samples the test image best, has the highest activation $A_{out,i,m}$, and associates the test image to the pattern it learned. If several output neurons are used to learn the same pattern, the sums of all activation functions belonging together are evaluated. A classification accuracy, named recognition rate in the following, is determined by calculating the percentage of correctly assigned test images.

Two MNIST subsets were used first to compare experimental results with simulations. One subset contains the digits “0”, “1” and “9”, while the second subset consists of “0”, “3” and “8”. Thus, the patterns differ more from one another in the first set compared to the second set, where the patterns have more pixels in common. In total, $3 \cdot 784 = 2,352$ individual synaptic connections are needed. This number of functional devices is selected from each type (polycrystalline or amorphous devices) of the 4 kbit chips, and they are randomly assigned to the output neurons. The parameters $c = 4$ (binarization of test images), $k = 5$ (slope of output neurons activation function in Eq. 4) and $p_{sat} = 35\%$ (for to avoid saturation of weights) were optimized in simulations. Here, recognition rates (mean value and standard deviation) of 84.5% (±4.6%) for the subset {0, 3, 8}, and 87.0% (±4.8%) for the subset {0, 1, 9} were determined with 100 simulation runs for each subset with five learning epochs each. It should be emphasized here that the algorithm converges within these five training epochs. Combining the results of both subsets, a recognition rate of 85.7% (±4.9%) is obtained. This is plotted in Fig. 10a as dashed line (mean value) and grey area (standard deviation). Receptive fields trained experimentally are shown in Fig. 11 for both device technologies, both subsets, and both state transitions used for learning. It is obvious that both types of devices and both transitions can be used within the network to learn the respective patterns in hardware. Furthermore, the switching windows size ΔV_{sw} affects the learned patterns. In particular, the smallest switching window corresponding to the set transition of the amorphous

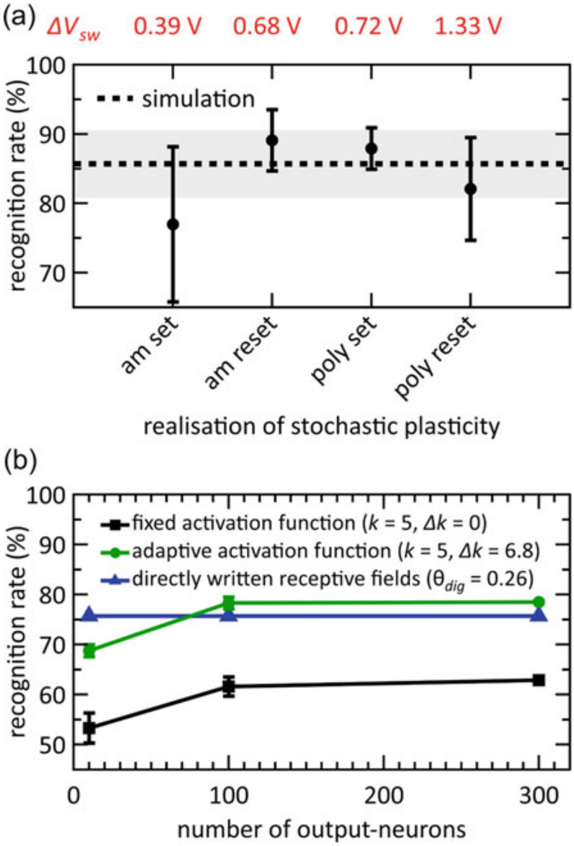

Fig. 10 Recognition rates of the StochANN. **a** Combined experimental results for MNIST subsets {0, 1, 9} and {0, 3, 8}. Mean values and standard deviations of five experimental runs for each pattern (i.e., ten runs in total for each data point) are shown as black dots while standard deviations are given as error bars. Simulation results (100 runs for each subset) are given as dashed line (mean value) and gray area (standard deviation). The abbreviations “poly” and “am” denote the polycrystalline and the amorphous HfO_{2-x}-based devices, and “set” and “reset” denote the transition used to emulate stochastic plasticity. **b** simulation results for the whole MNIST dataset are shown (mean values and standard deviation of five runs with five learning epochs each). These results are achieved with a fixed activation function (black squares) and an adaptive activation function (green circles) of the output neurons. The recognition rates with directly written receptive fields are also shown for comparison (blue triangles). Reprinted from [120] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

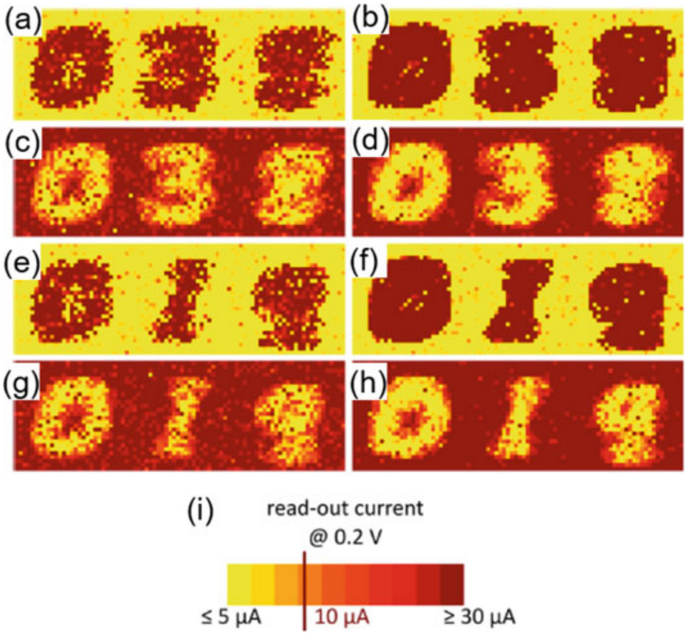

Fig. 11 Receptive fields of learned patterns in hardware. The read-out currents of the RRAM devices measured with 0.2 V are shown. In **a**–**d**, the used patterns are {0, 3, 8} learned with the set transition of polycrystalline devices **a**, the set transition of amorphous devices **b**, the reset transition of polycrystalline devices **c**, and the reset transition of amorphous devices **d**. In **e**–**h**, the used patterns are {0, 1, 9} learned with the set transition of polycrystalline devices **e**, the set transition of amorphous devices **f**, the reset transition of polycrystalline devices **g**, and the reset transition of amorphous devices **h**. Read-out currents are encoded in the pixel color as specified in **i**. Reprinted from [120] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

devices leads to receptive fields, which are more challenging to differentiate visually [Fig. 11b, f] compared to the largest switching window belonging to the reset transition of the polycrystalline devices [Fig. 11c, g]. However, the determined recognition rates show deviations from the simplified assumption that an increased switching window leads to better classification accuracy. In Fig. 10a, the recognition rates for both technologies and both state transitions are show. Each data point denotes combined mean values and standard deviations for five experimental runs of each subset (i.e., ten experimental runs in total for each data point). On the x-axis, “am” denotes amorphous and “poly” denotes polycrystalline HfO_{2-x} devices. Furthermore, ΔV_{sw} is given in Fig. 10a. In summary, the medium-sized switching windows corresponding to the reset transition of amorphous HfO_{2-x} devices and the set transition of polycrystalline HfO_{2-x} devices show performances within the error margin of the simulations. Thus, the simulations accurately reproduce the experimental results. A larger switching window for the reset transition of the polycrystalline HfO_{2-x} devices and a narrower switching window for the set transition of the amorphous HfO_{2-x}

devices, however, result in worse recognition performance. Thus, evidence is shown that device variability has an impact on network performance. In particular, the size of the switching window must not be too small to optimize the stochastic synapses in the proposed StochANN. The reason for the worse accuracy obtained with the largest ΔV_{sw} has to be evaluated in the future.

The network performance is tested with all ten patterns in simulations as well. All results were obtained with five learning epochs in five simulation runs. The recognition rates are summarized in Fig. 10b. Black dots show the StochANN performance for a fixed slope $k = 5$ of the output neurons' activation function. Recognition rates were determined as 53.3% (±3.0%) for ten output neurons (i.e., one for each pattern) 61.6% (±1.9%) for 100 output neurons and 62.9% (±0.7%) for 300 output neurons. An increase in learning epochs or the number of output neurons does not lead to any improvement. Furthermore, these results are compared to patterns written directly into the synaptic states without any learning algorithm involved. For this purpose, the input patterns were binarized using a fixed threshold Θ_{bin}. The synaptic weights were set to 1 if the corresponding pixel intensity of the input images were larger than the threshold. As shown by the blue triangles in Fig. 10b, a 75.7% recognition rate was obtained for a fixed Θ_{bin} of 0.26, which was optimized in simulations. No standard deviation can be denoted since no stochasticity is involved but only one deterministic prototype of each pattern is stored. Thus, binarizing the input images and writing them directly into the receptive fields improves the performance compared to the stochastic learning rule. However, a thorough optimization of Θ_{bin} for the specific dataset has to be performed, which becomes more tedious as the number of different input patterns increases. Moreover, an adaptive slope of the output neurons' activation function in combination with the stochastic learning rule leads to even higher recognition rates [green squares in Fig. 10b]. Here, the slope k is adapted with

$$k_m = k_0 - \Delta k \cdot \frac{\sum_{j=1}^{784} w_{j,m}}{784}, \tag{6}$$

where k_0 is the base value, and Δk is a positive constant weighted by the total strength of the synaptic connections. Thus, the slope is steeper for neurons that have learned patterns with less active pixels, leading to a stronger activation of those neurons for less input strength, as can be seen in Fig. 9. The slope adaptation only depends on the final weights stored in the synaptic connection. No adaptation during learning is necessary. The adaptive slope has similar functionality to variable threshold values for the output neurons reported for other pattern recognition networks [89, 90, 92, 93, 138, 139]. Here, the adaptive thresholds are essential to obtain a high recognition performance by emulating homeostasis. With $k_0 = 5$ and $\Delta k = 6.8$, a classification accuracy of 68.8% (±1.2%), 78.3% (±1.2%) and 78.5% (±0.2%) were achieved for 10, 100, and 300 output neurons, respectively. This shows that the stochastic learning rule slightly outperforms the directly written receptive fields for more than 100 neurons in the output layer. This can be explained by the fact that only one prototype of each pattern exists for directly written patterns, while variations of the prototypes exist in the learned receptive fields.

Using more complex time-independent networks performing supervised learning leads to recognition rates $> 98.5\%$ for SNNs [140, 141] and 99.87% for a thoroughly optimized CNN [142] and thus comparable to human performance estimated to be approx. 99.8% [143]. Using unsupervised learning in time-dependent neural networks is reported to achieve 93.5% with 300 output neurons [90] as well as 91.9% and 95.0% with 1600 and 6400 output neurons and the same amount of inhibitory neurons [138]. The latter approach was extended in Ref. [95] to a so-called lattice map (LM)-SNN leading to an accuracy of 94.07% for 1600 excitatory and inhibitory neurons, respectively. For a broad overview of different time-dependent and time-independent networks using supervised or unsupervised learning methods, the reader is referred to the overwhelming literature [98, 138, 142, 144–146]. All results named so far were obtained in simulations where only in [90] memristive devices were modeled to be used as synaptic connections. A fully hardware-implemented CNN based on multilevel RRAM devices can achieve recognition rates of 96.2% [147]. Here, a five-layer network is trained in software in a supervised manner, and the weights are subsequently transferred to eight 128×16 1T-1R arrays using two devices as one synapse to obtain positive and negative weights. Moreover, re-training of the last feature extraction layer was done in hardware. Another approach, in which two analog RRAM devices are used as one hardware synapse together with software neurons, reaches an accuracy of 91.7% for a re-scaled MNIST dataset of $8 \cdot 8$ pixel size. Here, a three-layer network using one array of 128×64 1T-1R devices can be utilized for learning directly in hardware using a supervised learning scheme [148]. Simulations of an extended network show a recognition rate of 97.3% taking device variability into account. A neuromorphic processor implementing a multilayer SNN with static random access memory (SRAM) allowing on-line supervised learning reaches a recognition accuracy of 97.83% [149]. Furthermore, hardware acceleration of DNN inference with pre-trained weights transferred to PCM devices is reported to lead to a recognition rate of 98.3% [150]. Another integrated circuit utilizes memristive devices as synaptic connections with the possibility of on-line learning has also been published [151].

The StochANN performance shown here is promising for such a simple network structure but has to be improved to compete with other reported networks. One big drawback of the proposed concept is that only the averaged pattern can be learned. Transfer the supervised learning rule into an unsupervised learning approach can potentially help to extract more prototypes of each pattern [89, 90, 93] without the need for supervised learning by computing and gradually improving an error function, as done in classical backpropagation algorithms [77, 80, 81]. Furthermore, using several binary devices as one synapse can help to improve the network performance [85, 129, 131, 133]. The advantage of the proposed network is that learning can be done directly in hardware with a mature technology using fully CMOS-integrated RRAM devices as synapses.

4 Time-Dependent Neural Networks

In this section, examples of time-dependent memristive networks are covered. In Sect. 4.1, an SNN based on analog memristive devices performing bio-inspired learning and pattern recognition is presented [93]. Simulations reproducing real device behavior on learning the MNIST benchmark are provided while the impact of device variability and yield is investigated. A mixed-signal circuit implementation using real crossbar-integrated double barrier memristive devices (DBMDs) [152] to learn basal patterns experimentally is furthermore shown [94]. In Sect. 4.2, examples for oscillator computing with memristive devices are provided. According to Fig. 2, these networks show the highest degree of biological inspiration and, therefore, the highest amount of cognitive performance is expected. The oscillator-based computing scheme shown below emulates perception by transient synchronization of memristively coupled oscillators. Thereby, it establishes a certain analogy to biology to solve the binding problem [113]. The influence of the switching dynamics of two types of memristive devices on the synchronization of oscillators is, furthermore, investigated, and device requirements for oscillatory computing are deduced.

4.1 Bio-Inspired Learning with Analog Memristive Devices

In this section, a time-dependent neural network utilizing analog memristive devices to emulate bio-inspired learning for a pattern recognition task is presented. The network performance is investigated by simulations incorporating real device behavior [93] and by the realization of a mixed-signal circuit using real crossbar-integrated devices [94]. Here, LTP and LTD are induced by replicating the Hebbian learning rule described in Sect. 1 for unsupervised bio-inspired learning. Hebbian learning was already realized a decade ago with single memristive devices by emulating STDP [153, 154] as well as LTP and LTD [155].

The used devices are so-called double barrier memristive devices (DBMDs) with the layer sequence Au/Nb_xO_y(2.5nm)/Al_2O_3(1.3 nm)/Nb [152] which are explained in detail in Chap. 3. Here, memristive switching is reported to take place by field-driven oxygen ion movement within Nb_xO_y, modulating the effective Schottky barrier height and the effective tunneling width of the Al_2O_3 [152, 156]. Thus, a homogeneous interface-based switching leading to a gradual resistance change is performed. The amount of resistance change depends on the applied voltage amplitude and time. A mathematical description of experimentally determined switching data is given by the memristive plasticity model of Ziegler et al. [24]. This model is compatible with advanced biophysical plasticity models that can fit experimental data on STDP, while it is also suitable to describe plasticity emulation with memristive devices. In that way, a behavioral model is obtained, which can be used for network-level simulations to explore how the modeled devices can be utilized to emulate Hebbian plasticity in trainable neuromorphic networks. Therefore, the degree of conductance change, i.e., the change of synaptic weight ω, is linked to the applied voltage pulses. The weight change is given by [24]

$$\frac{d\omega}{dt} = \beta\,(\omega)\,\omega(t)\left(1 - \frac{1}{\omega_{max}}\omega(t)\right). \tag{7}$$

Here, β is the weight-dependent learning rate, and ω_{max} is the maximum achievable weight. The switching dynamics of memristive devices are expressed in β, which depends not only on the electrical stimuli but also on the present conductance state. Thus β depends on the switching mechanism of the memristive device and can lead to various learning behaviors [24, 157]. The learning rate can be different for potentiation β_p and depression β_d. Furthermore, the synaptic weight ω represents the conductance G of a memristive device. Since the conductance change usually depends on the voltage pulse amplitude ΔV as well as the width Δt and the number of pulses n, the learning rates β_p and β_d are also modeled to be dependent on these parameters [24]:

$$\beta_p\,(G, n, \Delta t, \Delta V) = k_p\alpha(\Delta V)\lambda(\Delta t)(1 - \gamma G(n-1)) \tag{8}$$

$$\beta_d\,(G, n, \Delta t, \Delta V) = -k_d\alpha(\Delta V)\lambda(\Delta t)\gamma G(n-1), \tag{9}$$

where k_p, k_d, and γ are positive constants, while α and λ account for the non-linearity of the memristive devices' switching process.

Figure 12 shows the plasticity measurements (conductance vs. pulse number) of DBMDs investigated with voltage pulses of different amplitude and widths on 86 single devices. Black dots denote the average data from 86 individual devices, while error bars denote the standard deviation. Red solid lines show a replication with the introduced plasticity model. Model parameters are given in the respective original paper [93]. The gradual conductance modulation is clearly visible in Fig. 12a. Here, 1000 equivalent positive voltage pulses inducing potentiation and subsequent 1000 equivalent negative voltage pulses inducing depression were applied to the devices, as illustrated in the inset. A pulse duration of $\Delta t = 1$ ms was chosen together with $\Delta V = 3.9$ V and $\Delta V = -2.5$ V for potentiation and depression, respectively. Device conductance was read out by applying a voltage of 0.48 V, i.e., well below the threshold to change the device state [152], after every 100 potentiation or depression pulses. The data are depicted relative to the average maximum conductance $G_{max} =$ 100 nS of all devices after 1000 potentiation pulses. To determine the variations of the device conductance in dependency on the pulse amplitude and width, potentiation pulses with ΔV between 2.4 V and 3.7 V and a fixed width of 1 ms [Fig. 12b] and Δt ranging from 1 ms to 30 ms and a fixed amplitude of 3.9 V [Fig. 12c] were used. The data points in both figures show device conductance after 1000 pulses. The impact of depression pulses is shown in Fig. 12d for 1000 voltage pulses of 30 ms length, and ΔV between -1.4 V and -2.6 V applied to previously fully potentiated devices. An asymmetry between positive and negative voltages is obvious. Moreover, the device conductance is nearly unaffected for positive voltages of 2.4 V and below

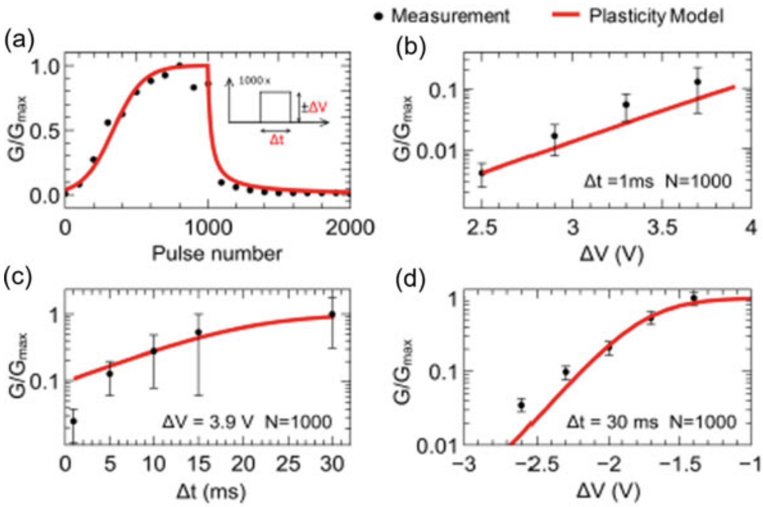

Fig. 12 Plasticity measurements of DBMDs: Mean values from 86 individual devices are given as black dots, while error bars denote the standard deviation. Conductances are normalized with the average maximum conductance G_{max} achieved with 1000 potentiation pulses of $\Delta V = 3.9$ V amplitude and $\Delta t = 1$ ms width. Fits with the plasticity model are given as red solid lines. **a** Typical conductance modulation with 1000 potentiation pulses ($\Delta V = 3.9$ V, $\Delta t = 1$ ms) and 1000 depression pulses ($\Delta V =$ -2.5 V, $\Delta t = 1$ ms) is shown. Each data point depicts the normalized conductance measured after 100 pulses. The conductance change after 1000 potentiation pulses with varying pulse amplitudes and widths is shown in **b** and **c**, respectively. The conductance change with 1000 depression pulses of different amplitude is shown in **d**. Reprinted from [93] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

as well as for negative voltages with an absolute value of 1.4 V or below. By using potentiation pulses with $\Delta V = 3.9$ V, however, the conductance can be increased by two orders of magnitude, which can be fully turned back with $\Delta V = -2.6$ V. A thorough analysis of the data is given in [93].

The measured data incorporated in the plasticity model can now be used to simulate a neuromorphic network capable of learning visual patterns by adjusting the synaptic weights emulated by DBMDs. The MNIST dataset [99] of handwritten digits, which is introduced in Sect. 3.2, shall be learned. The network operates similarly as networks reported in other works [89–92]. The two-layer feedforward network is schematically shown in Fig. 13 [93]. Here, each input neuron (blue circles) is connected to every output neuron (red circles) by DBMDs (symbols of memristive devices) arranged in a crossbar array. Every input neuron stochastically encodes the intensity of one pixel into voltages pulses [91–93]. Therefore, the pixel intensities are normalized to the interval $(-1, 1)$. The absolute values of the normalized intensities denote the probability of an input voltage pulse generation, while the sign stands for the voltage polarity. In that way, every input neuron either generates no spike or voltages pulses of +0.6 V or -0.6 V amplitude, both not affecting the device conductance by themselves. The currents flowing to the output neurons depend not only on

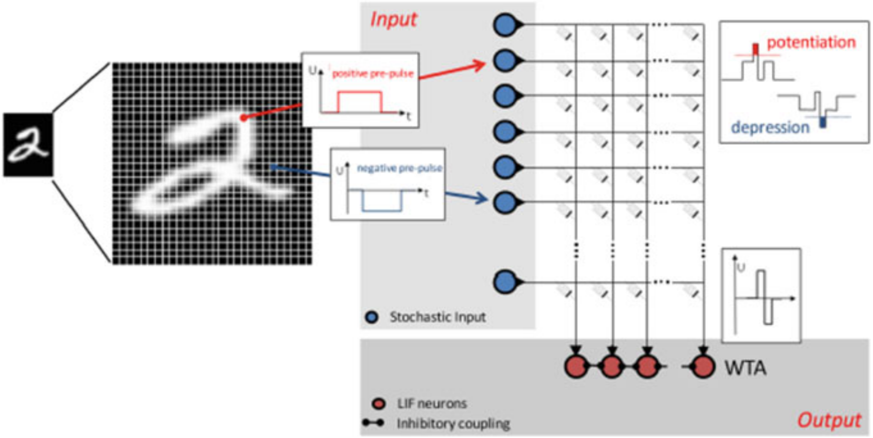

Fig. 13 Schematic of the simulated neural network. Reprinted from [93] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

the voltage pulse amplitudes but also on the conductance of the devices. The leaky integrate-and-fire (LIF) output neurons [33, 158] integrate the incoming stimuli until a certain threshold is reached [89]. When an output neuron reaches its threshold, a voltage pulse consisting of a positive part with $\Delta V = 2.9$ V and a negative part with $\Delta V = -2.3$ V. This post-synaptic pulse overlaps with the pre-synaptic pulses and changes the conductance of memristive devices connected to the respective output neuron. If a positive pre-synaptic pulse superimposes with the post-synaptic pulse, a potentiation takes place for the positive part ($V_{sum} = 3.5$ V) while the negative part does not affect the memristive state significantly ($V_{sum} = -1.7$ V). Vice versa, a net depression takes place when a negative input pulse overlaps with an output pulse ($V_{sum} = 2.3$ V and $V_{sum} = -2.9$ V, respectively). If no input pulse occurs, the post-synaptic pulse's voltage amplitudes alone do not significantly impact device conductance. Thus, input pixels with a strong intensity lead to positive input pulses, which, superimposed with induced output pulses, increase the device conductance. Pixels with low intensity induce negative input pulses, which lead to decreased conductance if an output spike simultaneously occurs. Thus, prototypes of the patterns are stored in the resistance states of the memristive devices. All devices connected to an output neuron are building the receptive field of this specific neuron. In that way, unsupervised associative learning based on local Hebbian plasticity is realized. Essential for network performance is, furthermore, an inhibitory coupling network for the output neurons implementing a winner-takes-it-all (WTA) mechanism, in which the first spiking neuron resets the integration of all other neurons [89]. Moreover, an adapting individual threshold is implemented for the output neurons to allow that all output neurons participate equivalently in learning. This mimics homeostasis in biological systems [89].

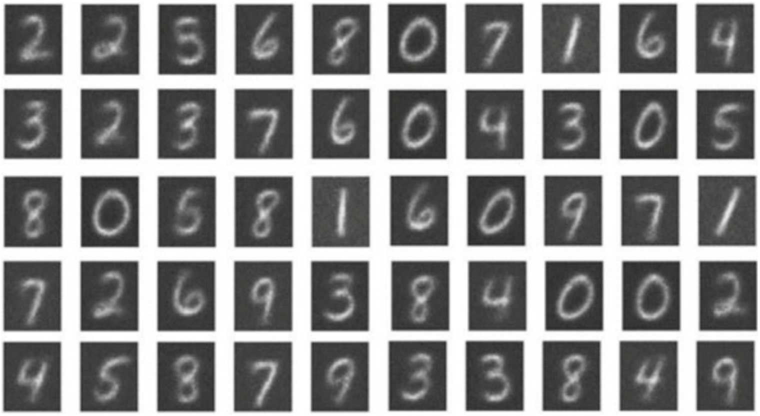

Fig. 14 Obtained receptive fields after unsupervised learning with 50 output neurons. In the gray-scale used to visualize device conductance white indicates maximum device conductance (strong synaptic weight), while black represents minimum conductance values (weak synaptic weight) of the memristive devices. Reprinted from [93] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

After learning, the network can be used to classify unknown images. Therefore, the output neuron, which receptive field matches the input image best, creates an output pulse. To assign every output neuron to its learned pattern, a small amount of pre-classified images is applied to the network and it is evaluated for which pattern the output neurons get activated. Afterward, the network performance can be evaluated by applying all 10,000 images from the MNIST test dataset to the network and calculating recognition accuracy by determining the percentage of correctly assigned patterns. Therefore, only the pre-synaptic pulses to encode the images are used, and the spiking events of the output neurons are tracked while post-synaptic pulse generation is suppressed to stop changing the device states. For 10, 20, 50, and 100 output neurons, recognition rates of 65%, 70%, 77%, and 82%, respectively, were determined. These rates are in good agreement with similar networks [89–92]. A typical set of learned receptive fields obtained in a simulation with 50 output neurons is shown in Fig. 14. It can be seen that the implemented network using the Hebbian learning scheme is able to learn different prototypes of all ten patterns (digits from zero to nine). The obtained performance is significantly lower than those from other spiking networks, as described in Sect. 3.2. However, network requirements for using memristive devices in SNNs are examined in this work, while improving pattern recognition computing schemes was not intended.

The results presented so far were generated without taking device variability into account. Now, the impact of different degrees of D2D and C2C variability is investigated [93]. Therefore, networks with ten output neurons were trained with three iterations of the whole MNIST learning dataset. Results are shown in Fig. 15(a,b), in which every data point was determined as the average of three simulation runs. The findings are then compared in Fig. 15c to the measured variability of real

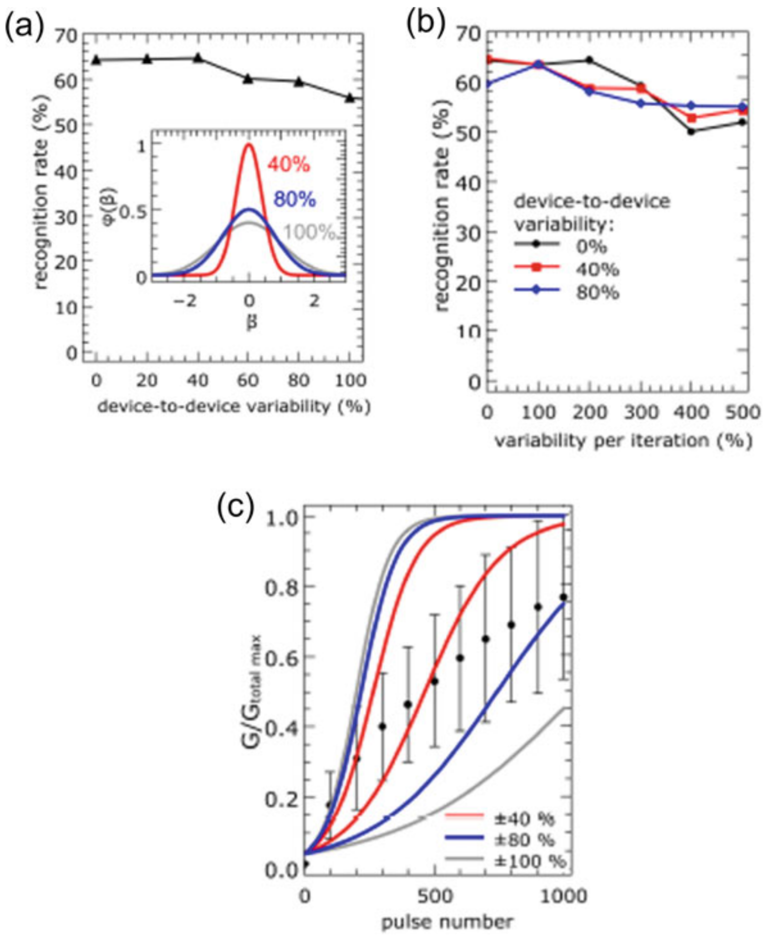

Fig. 15 Impact of device variability, i.e., variability of local learning rate of individual devices, on network performance. Device-to-device (D2D) and cycle-to-cycle (C2C) variability are depicted as standard deviation of Gaussian distributions. Each data point represents the mean value of three simulation runs. **a** D2D variability **b** C2C variability for a D2D variability of 0 % (black dots), 40 % (red squares) and 80 % (blue diamonds) **c** Experimentally determined D2D variation for potentiation with $\Delta V = 3.9$ V and $\Delta t = 1$ ms. Each data point shows the normalized mean conductance of 86 devices after 100 potentiation pulses. The red, blue, and gray lines indicate the range of learning rates with, 40%, 80%, and 100% D2D variability, respectively. Reprinted from [93] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

DBMDs. To model D2D variability, the learning rates of all devices were varied with Gaussian distributions [see inset of Fig. 15a] initially in every simulation run. As depicted in Fig. 15a, a standard deviation of up to 50% does not affect the recognition rate, while a further increase in variability slightly influences the performance. The C2C variability was modeled by varying the individual devices' learning rate for every applied image with a Gaussian distribution. Figure 15b (black dots) shows that a C2C variability of up to 200% does not affect the network performance. A combination of D2D and C2C variability, the most realistic scenario, is given in Fig. 15b with red squares and blue diamonds. Robust performance is achieved for a C2C variability of 100% combined with a D2D variability of 80%. Furthermore, the numerical investigation provides evidence that the most crucial performance losses result from a constant D2D variability since this effect does not average out in many learning iterations like it is the case for C2C variability. To estimate if the D2D variability of DBMDs does allow to use them as artificial synapses in the investigated network, the measured D2D variability is compared to the theoretically obtained boundaries. Figure 15c shows experimentally recorded device conductance of 86 individual devices after every 100th potentiation pulse with amplitudes of 3.9 V and 1 ms normalized by the highest recorded conductance $G_{max,total}$. Furthermore, the solid lines in Fig. 15c indicate the variation range of learning rates with 40%, 80%, and 100% D2D variability. Thus, the experimentally obtained D2D variability lies within the required variation interval. Moreover, in the original paper [93], the experimentally obtained yield, i.e., the percentage of functional devices, is shown to be approx. 98%, which does not influence the network performance significantly. In conclusion, evidence is provided that DBMDs are attractive candidates to be used as artificial synapses in neuromorphic circuits. In particular, the gradual conductance change under voltage pulsing, as well as the variability and yield of such real devices are believed to be suitable for the investigated network and learning rule.

The possibility of using DBMDs as artificial synapses has also been shown experimentally [94]. Here, a two-layer network, like described above, has been implemented in a mixed-signal-circuit. The synaptic connections were emulated with a real crossbar array containing $16 \cdot 16 = 256$ devices. Due to the high I-V non-linearity and the diode-like character of the DBMDs, no additional selector devices are needed to avoid the sneak path problem (see Sect. 2 and Chap. 3). The I-V characteristics of crossbar integrated devices were proven to be similar to single devices [94], as it is also shown in Chap. 3. The neurons were emulated in software. During learning, the conductances measured with a pre-synaptic pulse of 0.9 V amplitude were used to compute the current flowing to the output neurons. If a post-synaptic spike is triggered, potentiation and depression pulses with $\Delta V = 3.6$ V and $\Delta t = 100$ ms as well as $\Delta V = -1.1$ V and $\Delta t = 300$ ms, respectively, were used. In that way, the overlapping of pre- and post-synaptic pulses was not realized, but the potentiation and depression took place according to the Hebbian learning scheme explained above. This deviation from simulations was needed because the simple circuit design did not allow for real parallel data processing [94]. However, the work aimed to provide a proof of principle that DBMDs can be used in a crossbar configuration without the need for additional selector devices as artificial synapses in neuromorphic networks.

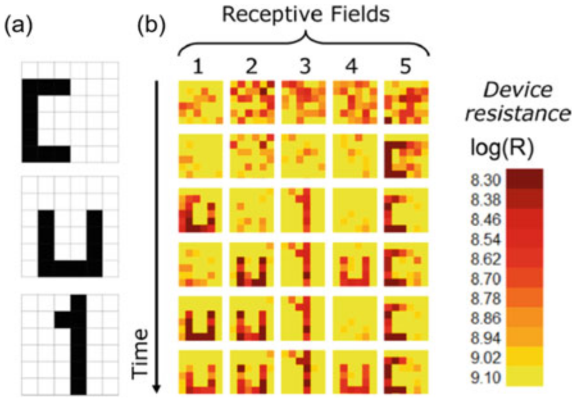

Fig. 16 Experimental results **a** Used training data **b** Obtained receptive fields during unsupervised learning with five output neurons. The pixel color indicates the memristive devices' resistance values (synaptic weights). Reprinted from [94] (licensed under CC BY 4.0, https://creativecommons.org/licenses/by/4.0/)

Figure 16a shows the simple $6 \cdot 6$ pixel pattern that the network shall learn. Figure 16b shows the developing receptive fields for five output neurons. Every pixel represents the resistance of one memristive device emulating the synaptic connection between one input and one output neuron. Initially, the resistances encoded in the pixel color are randomly distributed. After using 22,000 learning images in total, all three patterns are learned. Thus, the realization of unsupervised bio-inspired learning was possible with real DBMDs arranged in a crossbar structure thanks to the high I-V non-linearity and the diode-like character, as well as due to no required initial electro-forming step. As shown in the simulations above, the presented system can, in principle, cope with more complex tasks. However, a much larger amount of memristive cells is necessary for that. Due to the high resistances of DBMDs (even in LRS) and the several orders of magnitude lower wiring resistance ($\approx 100\ \Omega$ for an individual wire with the size $1100 \cdot 40 \cdot 0.5\ \text{mm}^3$) in the present crossbar array, larger arrays are believed to work as well.

4.2 *Oscillatory Computing*

Biological information processing relies heavily on nonlinear dynamics [36, 109]. This enables the integration of the multitude of information in an enormous and massively parallel network of neurons divided into functionally specialized regions such as the visual cortex, auditory cortex, or dorsolateral prefrontal cortex. Each of these regions participates as a context-dependent, self-organized, and transient subnetwork [36, 108]. Even if the underlying mechanisms are only partially understood, the interaction between dynamics and topology has been identified as one of the essential building blocks of information processing in the brain in recent years [109]. In the current understanding, it is assumed that information is encoded into coherent states by temporally correlated neural activity patterns [110]. This concept offers, particularly, an elegant explanation for the binding problem - the question of the mechanism of sensory integration, which allows our brain to construct uniform perceptions from the multitude of sensory information. First, evidence of these concepts could have been gathered from experiments with sensorimotor networks [111]. More recent studies have shown the universality of these concepts for the entire brain [110]. Recently, oscillators coupled by memristive devices have been shown to emulate this kind of information processing to some extent [112, 113].

In this section, we report on oscillator computing with memristive devices. We show how the dynamics of oscillator networks, coupled by memristive devices, is affected by the resistance of these devices. Therefore, important requirements for memristive devices are discussed as well as applications with the possibility to open up new pathways towards the construction of cognitive electronics.

4.2.1 Oscillator Computing with Memristive Connectivity

From studies of the thalamocortical system, Hoppensteadt and Izhikevich proposed a computational scheme based on oscillators with different frequencies that are weakly coupled to an externally changed medium, causing dynamic connectivity [159]. In their model, information is encoded in the oscillators' phase and/or frequency synchrony. The weak coupling, thereby, allows a dynamic change of their connectivity patterns depending on an external signal. A similar approach follows the idea of memristive coupled oscillator structures [112, 113], which will be explained in the following.

Figure 17 shows the model of two memristively coupled oscillators. Both oscillators are initially oscillating in their own frequencies f_i and f_j [Fig. 17a]. As long as their coupling is weak, they are not affecting each other. However, if the coupling strength between the oscillators increases, they start to interfere and synchronize in frequency and phase for sufficient high coupling strengths [Fig. 17b]. However, if the coupling strength is decreased thereafter, the oscillators will desynchronize again due to their different frequencies. This model can be realized with two self-sustained van der Pol oscillators with resistive coupling, as shown in Fig. 17c [112], i.e., via the conductance g_m of memristive devices. While in the following the model of memristive coupled oscillators will be discussed in the framework of van der Pol oscillators, any other type of oscillator may also be suitable [160]. The oscillator system shown in Fig. 17 can be described by the following set of second-order dimensionless nonlinear equations:

$$\frac{d^2y_1}{dt^2} = -\alpha\left(1 - y_1^2\right)\frac{dy_1}{dt} - \beta\frac{(y_1 + \gamma_1)^2}{\gamma_1} = g_m(x,t)\left(\frac{dy_2}{dt} - \frac{dy_1}{dt}\right) \tag{10}$$

$$\frac{d^2y_2}{dt^2} = -\alpha\left(1 - y_2^2\right)\frac{dy_2}{dt} - \beta\frac{(y_2 + \gamma_1)^2}{\gamma_1} = g_m(x,t)\left(\frac{dy_1}{dt} - \frac{dy_2}{dt}\right) \tag{11}$$

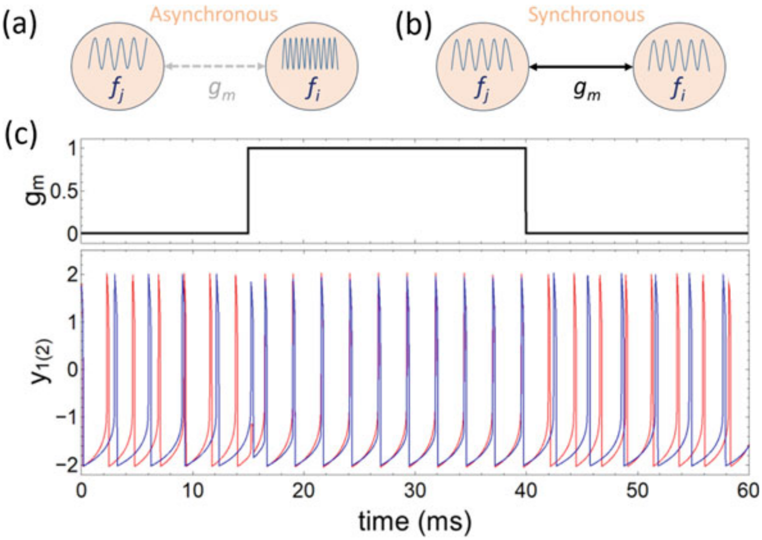

Fig. 17 **a**, **b** Concept of memristively coupled oscillators. **c** Depending on the coupling strength g_m (conductance of the memristive device) a frequency and phase synchronization of the oscillators occurs, as seen from the time course of the oscillator voltage y_1 (y_2) in the lower graph

Here, β, γ , and α are positive constants that define the uncoupled oscillators' damping, non-linearity, and frequency behaviors, respectively. Furthermore, g_m is the mutual coupling, representing the conductance of the memristive device. The memristive device can be modeled via

$$I = g_m(x,t) \cdot V \quad \text{with} \quad \frac{dx}{dt} = f(x, V, t). \tag{12}$$

Here, x is the memristive state variable, V the voltage across the memristive device (for example, $V = y_1 - y_2$), and f a dynamic function describing the voltage-driven atomic reconfiguration in the particular material system [160]. The device conductance influences the coupling strength via $g_m(x,t) = x \cdot G_{on} + (1-x) \cdot G_{off}$, where G_{on} and G_{off} are the maximum and minimum conductance of the device, respectively. The obtained result is shown in Fig. 17c. Here, the change of the coupling strength [upper graph in Fig. 17c] synchronizes the initially asynchronous oscillators. Thus, memristively coupled oscillators allow emulating the initially described essential principle of biological information processing: synchronization (information encoding into coherent states by correlated neural activity) and memory (change of connectivity).

4.2.2 Memristive Devices for Oscillator Computing

Using suitable memristive devices is an important point for realizing the previously presented model in hardware. The typical characteristics of memristive devices can be roughly divided into two classes. The first class of devices shows an abrupt resistance jump at a specific voltage (type 1), while the second class exhibits a gradual resistance change under an applied voltage (type 2). In Fig. 18, representatives of the two classes of memristive devices are compared. The first device (type 1) has the layer sequence Al/TiO$_x$/Ag and is an electro-chemical metallization cell (ECM), while the second device (type 2) is composed of the layer sequence TiN/TiO$_x$/HfO$_x$/Au, and its resistance change is functionally based on a valence-change mechanism (VCM). For the latter, a bi-layer oxide structure has been used, which is known to stabilize the resistance switching mechanism [161, 162]. Figure 18b shows measurements for the two types of devices. While for type 1 cell an abrupt jump in the resistance is observed, and the investigated type 2 device shows a more gradual transition of the conductance under voltage cycling. To analyze the consequences of the device type on the dynamics of coupled oscillators, two scenarios were simulated: (i) an abrupt and fast change of the coupling strength, when a threshold voltage is reached (behavior type 1) and (ii) a gradual change of the coupling strength, when the threshold value $V_{th,p}$ is exceeded (grey line in Fig. 18). Furthermore, the bi-directional switching behavior was taken into account such that g_m is reduced again when the voltage falls below a negative threshold value $V_{th,n}$. The obtained results are shown in Fig. 18c. It can be seen that the binary switching devices show a much faster synchronization.

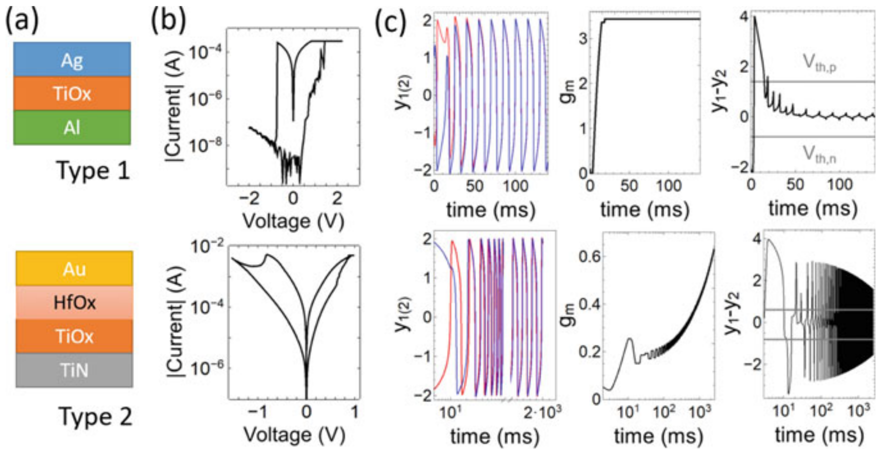

Fig. 18 **a** Typical memristive devices with archetypal switching characteristics. **b** I-V characteristics of an Al/TiO$_x$/Ag cell (type1) and TiN/TiO$_x$/HfO$_x$/Au (type2). **c** Simulation of the dynamics of the two coupled oscillators (y_1 and y_2) via a memristive device (conductance g_m) with an abrupt change in resistance (red) and a gradual change (blue). Here $y_1 - y_2$ represents the voltage across the memristive device. The threshold values for set and reset of the devices are shown in gray

4.2.3 Experimental Realization and Networks

A hardware implementation of memristively coupled oscillators is shown in Fig. 19. Here, van der Pol oscillators are realized via a programmable unijunction transistor (PUT) based circuit, as described in detail in Ref. [112]. In this circuit, the two oscillators are coupled via a memristive device so that their voltage oscillations (at points u_j and u_i) are smaller than the thresholds of the devices. Thus, the voltage oscillations cannot change the resistance value of the memristive device [see Fig. 19a]. The ECM cell shown in Fig. 18a was used as the memristive device, which changes its conductance g_m depending on an external voltage signal u_{learn}. As shown in Fig. 19b, the oscillators can be synchronized with this circuit depending on g_m. For the example shown in Fig. 19b, a voltage sequence of seven voltage pulses of 2.2 V amplitude was applied to the memristive device. This yields a conductance change of the device.

The transition to a simple artificial neural network is shown in Fig. 19c and d. The four oscillators of the first layer are connected to the two oscillators a and b of the second layer. Initially, all six oscillators of the network oscillate in their own frequency, as indicated in the contour plot by the different colors [Fig. 19c]. By applying a voltage sequence u_{learn}, the connectivity matrix g_m between the oscillators changes, which strengthens the coupling between oscillators 1, 2, and a as well as oscillators 3, 4, and b. As a result, the system oscillates with only two frequencies, as shown in Fig. 19d (different colors in contour plot). This computing scheme

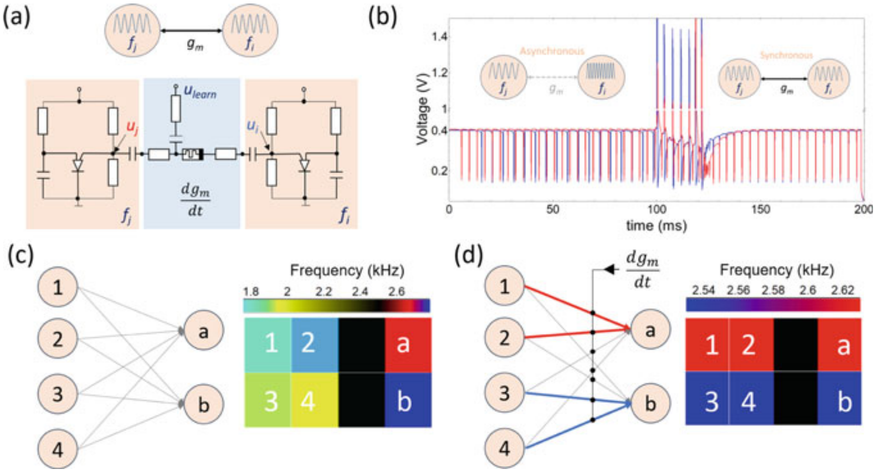

Fig. 19 **a** Experimental realization of memristively coupled van der Pol oscillators. **b** Voltage courses u_j and u_i are labeled in the circuit diagram in **a**. The voltage train u_{learn} (starting at 100 ms) decreases the resistance of the memristive device and the previously asynchronous oscillators synchronize. The ECM cell shown in Fig. 18a was used as memristive device. **c** Network of 6 van der Pol oscillators. By changing the coupling matrix g_m as a consequence of altered external conditions, the oscillators are synchronized in two frequencies (red and blue) **d**

was used in Ref. [113] to emulate perception. Thereby, u_{learn} can be identified as a level of perception that strengthens or weakens different connections depending on a temporal change of perception. For that, the inherent stochasticity of the memristive devices was used. In this respect, the number of voltage pulses defines the probability that the memristive device changes its resistance state. Thus, the number of voltage pulses of u_{learn} simulates the level of perception. Therefore, a certain analogy to biology can be established in the solution of the binding problem [113].

Acknowledgements Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - Project-ID 434434223 - SFB 1461. Moreover, the financial support by the DFG via the Research Unit 2093 is acknowledged. We thank Nora Kohlstedt for preparing Fig. 1(a)., i.e. the sketch of a neuron.

References

1. Yang, J.J., Strukov, D.B., Stewart, D.R.: Memristive devices for computing. Nat. Nanotechnol. **8**, 13–24 (2013)
2. Schuman, C.D., Potok, T.E., Patton, R.M., et al.: A survey of neuromorphic computing and neural networks in hardware (2017). arXiv:1705.06963 [cs]. arXiv: 1705.06963 [cs]
3. Burr, G.W., Shelby, R.M., Sebastian, A., et al.: Neuromorphic computing using non-volatile memory. Adv. Phys.: X **2**, 89–124 (2017)
4. Merolla, P.A., Arthur, J.V., Alvarez-Icaza, R., et al.: A million spiking-neuron integrated circuit with a scalable communication network and interface. Science **345**, 668–673 (2014)
5. Radamson, H.H., Zhu, H., Wu, Z., et al.: State of the art and future perspectives in advanced CMOS technology. Nanomaterials **10**, 1555 (2020)
6. Frenkel, C., Bol, D., Indiveri, G. Bottom-up and top-down neural processing systems design: neuromorphic intelligence as the convergence of natural and artificial intelligence (2021). arXiv: 2106.01288 [cs]
7. Moore, G.E.: Cramming more components onto integrated circuits. Electronics **38**, 4 (1965)
8. Dennard, R., Gaensslen, F., Yu, H.-N., et al.: Design of ion-implanted MOSFET's with very small physical dimensions. IEEE J. Solid-State Circuits **9**, 256–268 (1974)
9. Backus, J.: Can Programming be liberated from the von neumann style? A functional style and its algebra of programs. Commun. ACM **21**, 613–641 (1978)
10. Mead, C., Ismail, M.: Analog VLSI Implementation of Neural Systems First. Springer, Boston (1989)978-1-4613-1639-8
11. Mead, C.: Neuromorphic electronic systems. Proc. IEEE **78**, 1629–1636 (1990)
12. Chicca, E., Stefanini, F., Bartolozzi, C., et al.: Neuromorphic electronic circuits for building autonomous cognitive systems. Proc. IEEE **102**, 1367–1388 (2014)
13. Indiveri, G., Liu, S.-C.: Memory and information processing in neuromorphic systems. Proc. IEEE **103**, 1379–1397 (2015)
14. Kendall, J.D., Kumar, S.: The building blocks of a brain-inspired computer. Appl. Phys. Rev. **7**, 011305 (2020)
15. Christensen, D.V., Dittmann, R., Linares-Barranco, B., et al.: Roadmap on neuromorphic computing and engineering (2022). arXiv:2105.05956 [cond-mat]. arXiv: 2105.05956 [cond-mat] (2021)
16. Kandel, E.R.: Principles of Neural Science, 5th edn. McGraw-Hill, New York (2013)978-0-07-139011-8
17. Luo, L.: Principles of Neurobiology. Garland Science, New York (2015)978-0-8153-4492-6

18. Birkoben, T., Kohlstedt, H., Matter mind matter (2022). arXiv:2204.12774 [cond-mat, q-bio]. arXiv: 2204.12774 [cond-mat, q-bio]
19. Bi, G.-Q., Poo, M.-M.: Synaptic modification by correlated activity: Hebb's Postulate Revisited. Annu. Rev. Neurosci. **24**, 139–166 (2001)
20. Izhikevich, E.: Simple model of spiking neurons. IEEE Trans. Neural Netw. **14**, 1569–1572 (2003)
21. Markram, H., Gerstner, W., Sjöström, P.J.: Spike-timing-dependent plasticity: a comprehensive overview. Front. Synapt. Neurosci. **4**, 2 (2012)
22. Hebb, D.O.: The Organization of Behavior: A Neuropsychological Theory. Psychology Press, New York (2002)978-1-4106-1240-3
23. Gerstner, W.: To Appear in: From Neuron to Cognition via Computational Neuroscience, M. Arbib, J. Bonaiuto (eds.), vol. 21. MIT Press, Cambridge (2011)
24. Ziegler, M., Riggert, C., Hansen, M., et al.: Memristive Hebbian plasticity model: device requirements for the emulation of Hebbian plasticity based on memristive devices. IEEE Trans. Biomed. Circuits Syst. **9**, 197–206 (2015)
25. Bi, G.-Q., Poo, M.-M.: Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. J. Neurosci. **18**, 10464–10472 (1998)
26. He, B.J.: Scale-free brain activity: past, present, and future. Trends Cogn. Sci. **18**, 480–487 (2014)
27. Izhikevich, E.M.: Dynamical Systems in Neuroscience: The Geometry of Excitability and Bursting. First MIT Press paperback edition. MIT Press, Cambridge (2010). ISBN:978-0-262-51420-0
28. Schroeter, M.S., Charlesworth, P., Kitzbichler, M.G., et al.: Emergence of rich-club topology and coordinated dynamics in development of hippocampal functional networks in vitro. J. Neurosci. **35**, 5459–5470 (2015)
29. Stein, R.B., Gossen, E.R., Jones, K.E.: Neuronal variability: noise or part of the signal? Nat. Rev. Neurosci. **6**, 389–397 (2005)
30. Smetters, D.K., Zador, A.: Synaptic transmission: noisy synapses and noisy neurons. Curr. Biol. **6**, 1217–1218 (1996)
31. Zhang, C., Zhang, D. Stepanyants, A.: Noise in neurons and synapses enables reliable associative memory storage in local cortical circuits. eNeuro **8** (2021)
32. Buzsáki, G., Draguhn, A.: Neuronal oscillations in cortical networks. Science **304**, 1926–1929 (2004)
33. Gerstner, W., Kistler, W.M.: Spiking Neuron Models: Single Neurons, Populations. Plasticity. Cambridge University Press, Cambridge (2002)978-0-521-89079-3
34. Ernst, U., Pawelzik, K., Geisel, T.: Synchronization induced by temporal delays in pulse-coupled oscillators. Phys. Rev. Lett. **74**, 1570–1573 (1995)
35. Hoppensteadt, F.C., Izhikevich, E.M.: Weakly Connected Neural Networks Applied Mathematical Sciences, vol. 126. Springer, New York (1997)978-0-387-94948-2
36. Buzsáki, G.: Rhythms of the Brain. Oxford University Press, New York (2006)978-0-19-530106-9
37. Arenas, A., Díaz-Guilera, A., Kurths, J., et al.: Synchronization in complex networks. Phys. Rep. **469**, 93–153 (2008)
38. Galizia, C.G., Lledo, P.-M. (eds.): Neurosciences: From Molecule to Behavior: A University Textbook. Springer Spektrum, Heidelberg (2013)978-3-642-10768-9
39. Gerstner, W., Kistler, W.M., Naud, R., et al.: Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition. Cambridge University Press, Cambridge (2014)978-1-107-06083-8
40. Amil, P., Cabeza, C., Masoller, C., et al.: Organization and identification of solutions in the time-delayed mackey-glass model. Chaos **25**, 043112 (2015)
41. Strogatz, S.H.: Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering, 2nd edn. Westview Press, a Member of the Perseus Books Group, Boulder (2015). ISBN: 978-0-8133- 4910-7

42. Watts, D.J., Strogatz, S.H.: Collective dynamics of ‘Small-World’ networks. Nature **393**, 440–442 (1998)
43. Uhlhaas, P., Pipa, G., Lima, B., et al.: Neural synchrony in cortical networks: history, concept and current status. Front. Integrat. Neurosc. **3** (2009)
44. Beer, R.D.: Dynamical approaches to cognitive science. Trends Cogn. Sci. **4**, 91–99 (2000)
45. Höfflinger, B.: CHIPS 2020 VOL. 2: New Vistas in Nanoelectronics. Springer International Publishing, Cham (2016). ISBN: 978-3-319-22093-2
46. Veendrick, H.: Nanometer Cmos Ics. Springer Science+Business Media, LLC, New York (2017)978-3-319-47595-0
47. Turing, A.M.I.: Computing Machinery and Intelligence. Mind **LIX**, 433–460 (1950)
48. Rueckert, U.: CHIPS 2020 VOL. 2: New Vistas in Nanoelectronics, Höfflinger, B., (ed.), pp. 249–274. Springer International Publishing, Cham (2016). ISBN: 978-3-319-22093-2
49. Ielmini, D., Waser, R.: Resistive Switching: From Fundamentals of Nanoionic Redox Processes to Memristive Device Applications. Wiley-VCH Verlag GmbH Co. KGaA, Weinheim (2016)978-3-527-33417-9
50. Lanza, M., Waser, R., Ielmini, D., et al.: Standards for the characterization of endurance in resistive switching devices. ACS Nano **15**, 17214–17231 (2021)
51. Bian, H., Goh, Y.Y., Liu, Y., et al.: Stimuli-responsive memristive materials for artificial synapses and neuromorphic computing. Adv. Mater. **33**, 2006469 (2021)
52. Wang, R., Yang, J.-Q., Mao, J.-Y., et al.: Recent advances of volatile memristors: devices, mechanisms, and applications. Adv. Intell. Syst. **2**, 2000055 (2020)
53. Di Ventra, M., Pershin, Y.V.: The parallel approach. Nat. Phys. **9**, 200–202 (2013)
54. Truong, S.N., Min, K.-S.: New memristor-based crossbar array architecture with 50- multiplication of analog neuromorphic computing. J. Semicond. Technol. Sci. **14**, 356–363 (2014)
55. Ielmini, D., Wong, H.-S.P.: In-memory computing with resistive switching devices. Nat. Electron. **1**, 333–343 (2018)
56. Maass, W.: Networks of spiking neurons: the third generation of neural network models. Neural Netw. **10**, 1659–1671 (1997)
57. Ziegler, M., Wenger, C., Chicca, E., et al.: Tutorial: concepts for closely mimicking biological learning with memristive devices: principles to emulate cellular forms of learning. J. Appl. Phys. **124**, 152003 (2018)
58. Sebastian, A., Le Gallo, M., Khaddam-Aljameh, R., et al.: Memory devices and applications for in-memory computing. Nat. Nanotechnol. **15**, 529–544 (2020)
59. Borghetti, J., Snider, G.S., Kuekes, P.J., et al.: Memristive’ switches enable ‘stateful. logic operations via material implication. Nature **464**, 873–876 (2010)
60. Xu, N., Fang, L., Kim, K.M., et al.: Time-efficient stateful dual-bit-memristor logic. physica status solidi (RRL) – Rapid Res. Lett. **13**, 1900033 (2019)
61. Li, S., Xu, C., Zou, Q., et al.: Pinatubo: A Processing-in-Memory Architecture for Bulk Bitwise Operations in Emerging Non-Volatile Memoriesin 2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC), pp. 1–6 (2016)
62. Maan, A.K., Jayadevi, D.A., James, A.P.: A survey of memristive threshold logic circuits. IEEE Trans. Neural Netw. Learn. Syst. **28**, 1734–1746 (2017)
63. Wright, C.D., Liu, Y., Kohary, K.I., et al.: Arithmetic and biologically-inspired computing using phase-change materials. Adv. Mater. **23**, 3408–3413 (2011)
64. Le Gallo, M., Sebastian, A., Mathis, R., et al.: Mixed-precision in-memory computing. Nat. Electron **1**, 246–253 (2018)
65. Zidan, M.A., Jeong, Y., Lee, J., et al.: A general memristor-based partial differential equation solver. Nat. Electron **1**, 411–420 (2018)
66. Sun, Z., Pedretti, G., Ambrosi, E., et al.: Solving matrix equations in one step with cross-point resistive arrays. PNAS **116**, 4123–4128 (2019)
67. Pershin, Y.V., Di Ventra, M.: Solving mazes with memristors: a massively parallel approach. Phys. Rev. E **84**, 046703 (2011)
68. Mostafa, H., Müller, L.K., Indiveri, G.: An event-based architecture for solving constraint satisfaction problems. Nat. Commun. **6**, 8941 (2015)

69. 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), pp. 1–13 (2016)
70. Li, C., Hu, M., Li, Y., et al.: Analogue signal and image processing with large memristor crossbars. Nat. Electron. **1**, 52–59 (2018)
71. Le Gallo, M., Sebastian, A., Cherubini, G., et al.: Compressed sensing with approximate message passing using in-memory computing. IEEE Trans. Electron Devices **65**, 4304–4312 (2018)
72. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**, 436–444 (2015)
73. Rawat, W., Wang, Z.: Deep convolutional neural networks for image classification: a comprehensive review. Neural Comput. **29**, 2352–2449 (2017)
74. LeCun, Y.: 1.1 Deep Learning Hardware: Past, Present, and Future in 2019 IEEE International Solid- State Circuits Conference - (ISSCC), pp. 12–19 (2019)
75. Jouppi, N.P., Young, C., Patil, N., et al.: In-datacenter performance analysis of a tensor processing unit. In: Proceedings of the 44th Annual International Symposium on Computer Architecture, pp. 1–12. Association for Computing Machinery, New York (2017). ISBN: 978-1-4503-4892-8
76. 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)
77. Burr, G.W., Narayanan, P., Shelby, R.M., et al.: Large-scale neural networks implemented with non-volatile memory as the synaptic weight element: comparative performance analysis (accuracy, speed, and power). In: 2015 IEEE International Electron Devices Meeting (IEDM), pp. 4.4.1–4.4.4 (2015)
78. Ambrogio, S., Narayanan, P., Tsai, H., et al.: Equivalent-accuracy accelerated neural-network training using analogue memory. Nature **558**, 60–67 (2018)
79. Gokmen, T. Vlasov, Y.: Acceleration of deep neural network training with resistive cross-point devices: design considerations. Front. Neurosci. **10** (2016)
80. Burr, G.W., Shelby, R.M., Sidler, S., 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 Devic. **62**, 3498–3507 (2015)
81. Yu, S.: Neuro-inspired computing with emerging nonvolatile memorys. Proc. IEEE **106**, 260–285 (2018)
82. Cartier, E.A., Kim, W., Gong, N., et al.: Reliability Challenges with Materials for Analog Computing in 2019 IEEE International Reliability Physics Symposium (IRPS), pp. 1–10 (2019)
83. Milo, V., Malavena, G., Monzio Compagnoni, C., et al.: Memristive and CMOS devices for neuromorphic computing. Materials **13**, 166 (2020)
84. Jia, Z., Maggioni, M., Smith, J., et al. Dissecting the NVidia Turing T4 GPU via microbenchmarking (2019). arXiv:1903.07486 [cs]. arXiv: 1903.07486 [cs]
85. Bill, J., Legenstein, R.: A compound memristive synapse model for statistical learning through STDP in spiking neural networks. Front. Neurosci. **8**, 412 (2014)
86. Valentian, A., Rummens, F., Vianello, E., et al.: Fully Integrated Spiking Neural Network with Analog Neurons and RRAM Synapses in 2019 IEEE International Electron Devices Meeting (IEDM), pp. 14.3.1–14.3.4 (2019)
87. Agarwal, S., Jacobs Gedrim, R.B., Hsia, A.H., et al.: Achieving ideal accuracies in analog neuromorphic computing using periodic carry. In: 2017 Symposium on VLSI Technology, pp. T174–T175 (2017)
88. Chicca, E., Indiveri, G.: A recipe for creating ideal hybrid memristive-CMOS neuromorphic processing systems. Appl. Phys. Lett. **116**, 120501 (2020)
89. Querlioz, D., Bichler, O. Gamrat, C. Simulation of a memristor-based spiking neural network immune to device variations In: The 2011 International Joint Conference on Neural Networks, pp. 1775–1781 (2011)
90. Querlioz, D., Bichler, O., Dollfus, P., et al.: Immunity to device variations in a spiking neural network with memristive nanodevices. IEEE Trans. Nanotechnol. **12**, 288–295 (2013)

91. Sheridan, P., Ma, W., Lu, W.: Pattern Recognition with Memristor Networks in 2014 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1078–1081 (2014)
92. Zahari, F., Hansen, M., Mussenbrock, T., et al.: Pattern recognition with TiOx-based memristive devices. Materials **2**, 203–216 (2015)
93. Hansen, M., Zahari, F., Ziegler, M., et al. Double-barrier memristive devices for unsupervised learning and pattern recognition. Front. Neurosci. **11** (2017)
94. Hansen, M., Zahari, F., Kohlstedt, H., et al.: Unsupervised Hebbian learning experimentally realized with analogue memristive crossbar arrays. Sci. Rep. **8**, 8914 (2018)
95. Hazan, H., Saunders, D., Sanghavi, D. T., et al.: Unsupervised Learning with Self-Organizing Spiking Neural Networks in 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–6 (2018)
96. Brivio, S., Conti, D., Nair, M.V., et al.: Extended memory lifetime in spiking neural networks employing memristive synapses with nonlinear conductance dynamics. Nanotechnology **30**, 015102 (2018)
97. Brivio, S., Ly, D.R.B., Vianello, E., et al.: Non-linear memristive synaptic dynamics for efficient unsupervised learning in spiking neural networks. Front. Neurosci. **15**, 27 (2021)
98. Pfeiffer, M., Pfeil, T.: Deep learning with spiking neurons: opportunities and challenges. Front. Neurosci. **12**, 774 (2018)
99. Lecun, Y., Bottou, L., Bengio, Y., et al.: Gradient-based learning applied to document recognition. Proc. IEEE **86**, 2278–2324 (1998)
100. Krizhevsky, A.: Learning multiple layers of features from tiny images. Technical report, University of Toronto, p. 60 (2009)
101. Russakovsky, O., Deng, J., Su, H., et al.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. **115**, 211–252 (2015)
102. Roy, K., Jaiswal, A., Panda, P.: Towards spike-based machine intelligence with neuromorphic computing. Nature **575**, 607–617 (2019)
103. Dora, S., Kasabov, N.: Spiking neural networks for computational intelligence: an overview. Big Data Cognit. Comput. **5**, 67 (2021)
104. Amir, A., Taba, B., Berg, D., et al.: A Low Power, Fully Event-Based Gesture Recognition System in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7388–7397 (2017)
105. Binas, J., Neil, D., Liu, S.-C., et al.: DDD17: End-to-end DAVIS driving dataset (2017) . arXiv:1711.01458 [cs]. arXiv: 1711.01458 [cs]
106. Sironi, A., Brambilla, M., Bourdis, N., et al.: HATS: histograms of averaged time surfaces for robust event-based object classification (2018). arXiv:1803.07913 [cs]. arXiv: 1803.07913 [cs]
107. Mueggler, E., Rebecq, H., Gallego, G., et al.: The event-camera dataset and simulator: event-based data for pose estimation, visual odometry, and SLAM. Int. J. Robot. Res. **36**, 142–149 (2017)
108. Chialvo, D.R.: Emergent complex neural dynamics. Nat. Phys. **6**, 744–750 (2010)
109. Engel, A.K., Fries, P., Singer, W.: Dynamic predictions: oscillations and synchrony in top-down processing. Nat. Rev. Neurosci. **2**, 704–716 (2001)
110. Fries, P.: A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. Trends Cogn. Sci. **9**, 474–480 (2005)
111. Gray, C.M., König, P., Engel, A.K., et al.: Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. Nature **338**, 334–337 (1989)
112. Ignatov, M., Hansen, M., Ziegler, M., et al.: Synchronization of two memristively coupled van der pol oscillators. Appl. Phys. Lett. **108**, 084105 (2016)
113. Ignatov, M., Ziegler, M., Hansen, M., et al.: Memristive stochastic plasticity enables mimicking of neural synchrony: memristive circuit emulates an optical illusion. Sci. Adv. **3**, e1700849 (2017)
114. Torrejon, J., Riou, M., Araujo, F.A., et al.: Neuromorphic Computing with nanoscale spintronic oscillators. Nature **547**, 428–431 (2017)

115. Walczyk, D., Walczyk, C., Schroeder, T., et al.: Resistive switching characteristics of CMOS embedded HfO2-based 1T1R cells. In: Microelectronic Engineering. Proceedings of the 17th Biennial International Insulating Films on Semiconductor Conference, vol. 88, pp. 1133–1135 (2011)
116. Kim, S., Ishii, M., Lewis, S., et al.: NVM Neuromorphic Core with 64k-Cell (256-by-256) Phase Change Memory Synaptic Array with on-Chip Neuron Circuits for Continuous in-Situ Learning in 2015 IEEE International Electron Devices Meeting (IEDM), pp. 17.1.1–17.1.4 (2015)
117. Mahadevaiah, M.K., Perez, E., Wenger, C. et al.: Reliability of CMOS Integrated Memristive HfO2 Arrays with Respect to Neuromorphic Computing in 2019 IEEE International Reliability Physics Symposium (IRPS), pp. 1–4 (2019)
118. Zarrin, P.S., Zahari, F., Mahadevaiah, M.K., et al.: Neuromorphic on-chip recognition of saliva samples of COPD and healthy controls using memristive devices. Sci. Rep. **10**, 19742 (2020)
119. Wenger, C., Zahari, F., Mahadevaiah, M.K., et al.: Inherent stochastic learning in CMOS-integrated HfO2 arrays for neuromorphic computing. IEEE Electron Device Lett. **40**, 639–642 (2019)
120. Zahari, F., Pérez, E., Mahadevaiah, M.K., et al.: Analogue pattern recognition with stochastic switching binary CMOS-integrated memristive devices. Sci. Rep. **10**, 14450 (2020)
121. Dong, T., Santos, S., Yang, Z., et al.: Sputum and salivary protein biomarkers and point-of-care biosensors for the management of COPD. Analyst **145**, 1583–1604 (2020)
122. Baker, S.B., Xiang, W., Atkinson, I.: Internet of Things for Smart Healthcare: Technologies, Challenges, and Opportunities. IEEE Access **5**, 26521–26544 (2017)
123. Guha, S., Jamal, F. I. Wenger, C.: A review on passive and integrated near-field microwave biosensors. Biosensors (Basel)**7**, E42 (2017)
124. Zarrin, P.S., Roeckendorf, N., Wenger, C.: In-vitro classification of saliva samples of COPD patients and healthy controls using machine learning tools. IEEE Access **8**, 168053–168060 (2020)
125. Zarrin, P.S., Roeckendorf, N.: Database containing saliva samples of COPD patients and healthy controls. https://archive.ics.uci.edu/ml/datasets/Exasens
126. Zarrin, P., Ibne Jamal, F., Roeckendorf, N., et al.: Development of a portable dielectric biosensor for rapid detection of viscosity variations and its in vitro evaluations using saliva samples of copd patients and healthy control. Healthcare **7**, 11 (2019)
127. Yang, Y., Lu, W.: Nanoscale resistive switching devices: mechanisms and modeling. Nanoscale **5**, 10076–10092 (2013)
128. Fries, P.: A mechanism for cognitive dynamics: neuronal communication through neuronal coherence. Trends Cogn. Sci. **9**, 474–480 (2005)
129. Hu, M., Wang, Y., Qiu, Q., et al.: The Stochastic Modeling of TiO2 Memristor and Its Usage in Neuromorphic System Design in 2014 19th Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 831–836 (2014)
130. Tuma, T., Pantazi, A., Gallo, M.L., et al.: Stochastic phase-change neurons. Nat. Nanotech **11**, 693–699 (2016)
131. Gaba, S., Sheridan, P., Zhou, J., et al.: Stochastic memristive devices for computing and neuromorphic applications. Nanoscale **5**, 5872–5878 (2013)
132. Suri, M., Querlioz, D., Bichler, O., et al.: Bio-inspired stochastic computing using binary CBRAM synapses. IEEE Trans. Electron Devic. **60**, 2402–2409 (2013)
133. Payvand, M., Nair, M.V., Müller, L.K., et al.: A neuromorphic systems approach to in-memory computing with non-ideal memristive devices: from mitigation to exploitation. Faraday Discuss. **213**, 487–510 (2019)
134. Grossi, A., Zambelli, C., Olivo, P., et al.: Electrical characterization and modeling of pulse-based forming techniques in RRAM arrays. Solid-State Electron. **115**, 17–25 (2016)
135. Milo, V., Zambelli, C., Olivo, P., et al.: Multilevel HfO2-based RRAM devices for low-power neuromorphic networks. APL Mater. **7**, 081120 (2019)
136. Fantini, A., Goux, L., Degraeve, R., et al.: Intrinsic Switching Variability in HfO2 RRAM in 2013 5th IEEE International Memory Workshop, pp. 30– 33 (2013)

137. Grossi, A., Perez, E., Zambelli, C., et al.: Impact of the precursor chemistry and process conditions on the cell-to-cell variability in 1T–1R based HfO 2 RRAM devices. Sci. Rep. **8**, 1–11 (2018)
138. Diehl, P. U. Cook, M.: Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Front. Comput. Neurosci. **9** (2015)
139. Yousefzadeh, A., Stromatias, E., Soto, M., et al.: On practical issues for stochastic STDP hardware with 1-Bit synaptic weights. Front. Neurosci. **12** (2018)
140. Lee, J. H., Delbruck, T. Pfeiffer, M.: Training deep spiking neural networks using backpropagation. Front. Neurosci. **10** (2016)
141. Wu, Y., Deng, L., Li, G., et al.: Spatio-temporal backpropagation for training high-performance spiking neural networks. Front. Neurosci. **12** (2018)
142. Ahlawat, S., Choudhary, A., Nayyar, A., et al.: Improved handwritten digit recognition using convolutional neural networks (CNN). Sensors **20**, 3344 (2020)
143. LeCun, Y., Jackel, L.D., Denker, J.S., et al.: Learning algorithms for classification: a comparison on handwritten digit recognition, 16 (1995)
144. Kheradpisheh, S.R., Ganjtabesh, M., Thorpe, S.J., et al.: STDP-based spiking deep convolutional neural networks for object recognition. Neural Netw. **99**, 56–67 (2018)
145. Zhang, T., Cheng, X., Jia, S., et al.: Self-backpropagation of synaptic modifications elevates the efficiency of spiking and artificial neural networks. Sci. Adv. **7**, eabh0146 (2021)
146. Martin, E., Ernoult, M., Laydevant, J., et al.: EqSpike: spike-driven equilibrium propagation for neuromorphic implementations. iScience **24**, 102222 (2021)
147. Yao, P., Wu, H., Gao, B., et al.: Fully hardware-implemented memristor convolutional neural network. Nature **577**, 641–646 (2020)
148. Li, C., Belkin, D., Li, Y., et al.: Efficient and self-adaptive in-situ learning in multilayer memristor neural networks. Nat. Commun. **9**, 1–8 (2018)
149. Park, J., Lee, J., Jeon, D.: A 65-Nm neuromorphic image classification processor with energy-efficient training through direct spike-only feedback. IEEE J. Solid-State Circuits **55**, 108–119 (2020)
150. Khaddam-Aljameh, R., Stanisavljevic, M., Mas, J.F., et al.: HERMES Core - A 14nm CMOS and PCM-based In-Memory Compute Core Using an Array of 300ps/LSB Linearized CCO-based ADCs and Local Digital Processing in 2021 Symposium on VLSI Technology, pp. 1–2 (2021)
151. Ishii, M., Kim, S., Lewis, S., et al.: On-Chip Trainable 1.4M 6T2R PCM Synaptic Array with 1.6K Stochastic LIF Neurons for Spiking RBM in 2019 IEEE International Electron Devices Meeting (IEDM)), pp. 14.2.1–14.2.4 (2019
152. Hansen, M., Ziegler, M., Kolberg, L., et al.: A double barrier memristive device. Sci. Rep. **5**, 13753 (2015)
153. Jo, S.H., Chang, T., Ebong, I., et al.: Nanoscale memristor device as synapse in neuromorphic systems. Nano Lett. **10**, 1297–1301 (2010)
154. Linares-Barranco, B., Serrano-Gotarredona, T., Camuñas-Mesa, L., et al.: On spike-timing-dependent-plasticity, memristive devices, and building a self-learning visual cortex. Front. Neurosci. **5**, 26 (2011)
155. Ohno, T., Hasegawa, T., Tsuruoka, T., et al.: Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. Nat. Mater. **10**, 591–595 (2011)
156. Dirkmann, S., Hansen, M., Ziegler, M., et al.: The role of ion transport phenomena in memristive double barrier devices. Sci. Rep. **6**, srep35686 (2016)
157. Querlioz, D., Bichler, O., Vincent, A.F., et al.: Bioinspired programming of memory devices for implementing an inference engine. Proc. IEEE **103**, 1398–1416 (2015)
158. Gerstner, W.: Time structure of the activity in neural network models. Phys. Rev. E **51**, 738–758 (1995)
159. Hoppensteadt, F.C., Izhikevich, E.M.: Oscillatory neurocomputers with dynamic connectivity. Phys. Rev. Lett. **82**, 2983–2986 (1999)
160. Pikovsky, A., Rosenblum, M., Kurths, J.: Synchronization: A Universal Concept in Nonlinear Sciences. ISBN: 978-0-521-53352-2. Cambridge University Press, Cambridge (2001)

161. Stathopoulos, S., Khiat, A., Trapatseli, M., et al.: Multibit memory operation of metal-oxide bi-layer memristors. Sci. Rep. **7**, 17532 (2017)

162. Park, S., Klett, S., Ivanov, T., et al.: Engineering method for tailoring electrical characteristics in TiN/TiOx/ HfOx/Au bi-layer oxide memristive devices **3**, 16 (2021)