Neuromorphic computing with multi-memristive synapses

Irem Boybat,1,2,3 Manuel Le Gallo,1 Nandakumar S. R.,1,3 Timoleon Moraitis,1 Thomas Parnell,1 Tomas Tuma,1 Bipin Rajendran,3 Yusuf Leblebici,2 Abu Sebastian,1 and Evangelos Eleftheriou1

1) IBM Research – Zurich, 8803 Rüschlikon, Switzerland.
2) EPFL, 1015 Lausanne, Switzerland.
3) New Jersey Institute of Technology, Newark, NJ 07102, USA.

Brain-inspired neuromorphic computing has recently emerged as a promising avenue towards building the next generation of intelligent computing systems. It has been proposed that memristive devices, which exhibit history-dependent conductivity modulation, could be used to efficiently represent the strength of synaptic connections between the neuronal nodes in neural networks. However, precise modulation of the device conductance over a wide dynamic range, necessary to maintain high network accuracy, is proving to be a challenging task, primarily due to the physical mechanisms that underlie the operation of these devices. To address this challenge, we present a generic multi-memristive synaptic architecture with an efficient global clock-based arbitration scheme. We show that this concept is applicable to both artificial neural networks (ANN) and spiking neural networks (SNN). Experimental results involving over a million phase change memory devices are presented where an SNN with multi-memristive synapses is used for unsupervised learning of temporal correlations between event-based data streams. The work presented opens a pathway for realizing large-scale neural networks using memristive technology, and represents a significant step towards the realization of energy-efficient neuromorphic computing systems.

The human brain with less than 20 Watts of power consumption offers a processing capability that exceeds the petaflops mark and thus outperforms state-of-the-art supercomputers by several orders of magnitude in terms of energy efficiency and volume. Building ultra-low power cognitive computing systems inspired by the operating principles of the brain is a promising avenue towards achieving such efficiency. Recently, deep learning has revolutionized the field of machine learning by providing human-like performance in areas such as computer vision, speech recognition, or complex strategic games. However, the current hardware implementations of deep neural networks are still far from competing with biological neural systems in terms of energy consumption and real-time information-processing capabilities.

One of the reasons for this inefficiency is that most of the neural networks are implemented on computing systems based on the conventional von Neumann architecture, that is, the processing unit and the memory are physically separated. There are a few attempts to build custom neuromorphic hardware that is optimized to implement neural algorithms. However, as these custom systems are typically based on conventional silicon CMOS circuitry, the area efficiency of such hardware implementations remains relatively low, especially for systems that learn in-situ and require non-volatile synaptic elements. Recently, a new class of emerging nanoscale devices has shown promise for realizing the synaptic dynamics in a compact and power-efficient manner. These memristive devices store information in their resistance/conductance states and exhibit conductivity modulation based on the programming history. The central idea in building cognitive hardware based on memristive devices is to store the synaptic weights as the conductance states of such devices.

The two essential red synaptic attributes that need to be emulated by memristive devices are the synaptic efficacy and plasticity. Synaptic efficacy refers to the generation of a synaptic output based on the incoming neuronal activation. In the case of artificial neural networks (ANN), the synaptic output is obtained by multiplying the real-valued neuronal activation with the synaptic weight. In the case of a spiking neural network (SNN), the synaptic output is generated when the presynaptic neuron fires and typically is a signal that is proportional to the synaptic conductance. Using memristive devices, synaptic efficacy can be realized using Ohm’s law by measuring the current that flows through the device when an appropriate voltage signal is applied. Synaptic plasticity, in contrast, is the ability of the synapse to change its weight, typically during the execution of a learning algorithm. An increase in the synaptic weight is referred to as potentiation, and a decrease in the synaptic weight is referred to as depression. In the case of an ANN, the weights are usually changed based on the backpropagation algorithm, whereas in an SNN, local learning rules such as spike-timing-dependent plasticity (STDP) or supervised learning algorithms such as NormAD could be used. The implementation of synaptic plasticity in memristive devices is achieved by the application of appropriate electrical pulses that change the conductance of these devices through various physical mechanisms, such as ionic drift, diffusion, ferroelectric effects, spintronic effects, and phase transitions.

Demonstrations that combine memristive synapses with digital or analog CMOS neuronal circuitry are indicative of the potential to realize highly efficient neuromorphic systems. However, to incorporate such devices into large-scale neuromorphic systems without compromising on the network performance, significant improvements in the memristive device characteristics are needed. Some of the device characteristics that limit the system performance include the limited conductance range, asymmetric conductance response (differences in the manner in which the conductance changes between potentiation and depression), the stochasticity associated with conductance changes, variability between devices, and reliability issues across a large array of devices.

a) Electronic mail: ibo@zurich.ibm.com
b) Electronic mail: ase@zurich.ibm.com
Clearly, advances in materials science and device technology could play a key role in addressing some of these challenges, but equally important are innovations in synaptic architectures. One example is the differential synaptic architecture in which two memristive devices are used in a differential configuration such that one device is used for potentiation and the other for depression. This was proposed for synapses implemented using phase change memory (PCM) devices which exhibit strong asymmetry in their conductance response. However, this approach suffers from issues related to device mismatch within the differential pair of devices as well as the need to refresh the device conductance frequently to avoid conductance saturation.\(^{33}\) In another approach proposed recently,\(^{37}\) several binary memristive devices are programmed and read in parallel to implement a synaptic element with multi-level weight storage capability. In this work, the probabilistic switching exhibited by certain types of memristive devices, such as metal-oxide resistive RAM, is exploited to implement this scheme. However, it may be challenging to achieve fine-tuned probabilistic switching across a large number of devices reliably, required in an experimental realization of this approach. Alternatively, pseudo-random number generators could be used to implement this probabilistic update scheme with deterministic memristive devices,\(^{38}\) albeit with the associated costs of increased circuit complexity.

In this article, we propose a multi-memristive synaptic architecture that overcomes the main drawbacks of the above-mentioned schemes, and experimentally demonstrate an implementation using nanoscale PCM devices. First, we present the concept of multi-memristive synapses. Next, we illustrate the challenges posed by memristive devices for neuromorphic computing by studying the operating characteristics of PCM. Using comprehensive models of the devices, we demonstrate the potential of the multi-memristive synaptic concept in realizing ANNs and SNNs for the exemplary benchmark task of handwritten digit classification. Finally, we present a large-scale experimental implementation of an SNN with multi-memristive synapses using more than one million PCM devices to detect temporal correlations in event-based data streams.

I. RESULTS

A. The multi-memristive synapse

The concept of the multi-memristive synapse is illustrated schematically in Fig. 1. In such a synapse, the synaptic weight is represented by the combined conductance of \(N\) devices. For the realization of synaptic efficacy, an input voltage corresponding to the neuronal activation is applied to all constituent devices. The sum of the individual device currents forms the net synaptic output. For the implementation of synaptic plasticity, only one device is selected and programmed at a time. This selection is done with a clock-based arbitration scheme where a global selection clock with \(N\) steps points to one of the devices. After the weight update, the clock is incremented by a fixed increment rate. Having an increment rate co-prime with the clock length \(N\) guarantees that all devices within each synapse eventually get selected and will receive a comparable number of updates provided there is a sufficiently large number of updates. Moreover, if a single selection clock is used for all synapses of a neural network, \(N\) can be chosen to be co-prime with the total number of synapses in the network to avoid updating the same device within each synapse repeatedly. In addition to the global selection clock, additional independent clocks, such as a potentiation frequency clock or a depression frequency clock, could be incorporated to control the frequency of potentiation/depression events further (see Fig. 1).

The constituent devices of the multi-memristive synapse can be arranged either in a differential or in a non-differential architecture. In the non-differential architecture, each synapse consists of \(N\) devices, and one device is selected and potentiated/depressed to achieve synaptic plasticity. In the differential architecture, two sets of devices are present and the synaptic conductance is calculated as \(G_{\text{syn}} = G_+ - G_-\), where \(G_+\) is the total conductance of the set representing the potentiation of the synapse and \(G_-\) is the total conductance of the set representing the depression of the synapse. Each set consists of \(N/2\) devices. When the synapse has to be potentiated, one device from the group representing \(G_+\) is selected and potentiated, and when the synapse has to be depressed, one device from the group representing \(G_-\) is selected and potentiated.

B. Multi-memristive synapses based on PCM devices

In this section, we will demonstrate the concept of multi-memristive synapses using nanoscale PCM devices. A PCM device consists of a thin film of phase change material sandwiched between two metal electrodes (Fig. 2(a)).\(^{39}\) The phase change material can be in a high-conductance crystalline phase or in a low-conductance amorphous phase. In an as-fabricated device, the material is typically in the crystalline phase. When a current pulse of sufficiently high amplitude (referred to as the depression pulse) is applied, a significant portion of the phase change material melts owing to Joule heating. If the pulse is interrupted abruptly, the molten material quenches into the amorphous phase as a result of the glass transition. To increase the conductance of the device, a current pulse (referred to as potentiation pulse) is applied such that the temperature reached via Joule heating is above the crystallization temperature but below the melting point, resulting in the recrystallization of part of the amorphous region.\(^{40}\) The extent of crystallization depends on the amplitude and duration of the potentiation pulse as well as on the number of such pulses. By progressively crystallizing the amorphous region with the application of potentiation pulses, a continuum of conductance levels can be realized.

In this section, we will demonstrate the concept of multi-memristive synapses using nanoscale PCM devices. A PCM device consists of a thin film of phase change material sandwiched between two metal electrodes (Fig. 2(a)).\(^{39}\) The phase change material can be in a high-conductance crystalline phase or in a low-conductance amorphous phase. In an as-fabricated device, the material is typically in the crystalline phase. When a current pulse of sufficiently high amplitude (referred to as the depression pulse) is applied, a significant portion of the phase change material melts owing to Joule heating. If the pulse is interrupted abruptly, the molten material quenches into the amorphous phase as a result of the glass transition. To increase the conductance of the device, a current pulse (referred to as potentiation pulse) is applied such that the temperature reached via Joule heating is above the crystallization temperature but below the melting point, resulting in the recrystallization of part of the amorphous region.\(^{40}\) The extent of crystallization depends on the amplitude and duration of the potentiation pulse as well as on the number of such pulses. By progressively crystallizing the amorphous region with the application of potentiation pulses, a continuum of conductance levels can be realized.
First, we present experimental characterization of single-device PCM-based synapses based on doped Ge$_2$Sb$_2$Te$_5$ (GST) and integrated into a prototype chip in 90 nm CMOS technology [33] (see the methods section for more details). Figure 2(b) shows the evolution of the mean device conductance as a function of the number of potentiation pulses applied. A total of 10,000 devices were used for the characterization and the programming pulse amplitude $I_{\text{prog}}$ was varied from 60 $\mu$A to 120 $\mu$A. It can be seen that the mean conductance value increases as a function of the number of potentiation pulses. The dynamic range of conductance response is limited as the change in the mean conductance decreases and eventually saturates with increasing number of potentiation pulses. Figure 2(c) shows the mean cumulative change in conductance as a function of the number of pulses for different values of $I_{\text{prog}}$. A well-defined nonlinear monotonic relationship exists between cumulative conductance change and the number of potentiation pulses. In addition, there is a granularity that is determined by how small a conductance change can be induced by applying a single potentiation pulse. Large conductance change granularity as well as nonlinear conductance response, both observed in the PCM characterization performed here, have been shown to degrade the performance of neural networks with memristive synapses [33,36,41]. Moreover, when a conductance decrease is desired, a single depression pulse in PCM has an all-or-nothing effect that fully depresses the device conductance to (almost) 0 $\mu$S. Such a strongly asymmetric conductance response is undesirable in memristive device-based implementations of neural networks [41] and this is a significant challenge for PCM-based synapses.

There is also significant intra- and inter-device variability associated with the conductance response in PCM devices. The intra-device variability in PCM is attributed to the differences in atomic configurations associated with the amorphous phase-change material created during the melt-quench process [41]. To investigate the intra-device variability, we measured the conductance change on the same PCM device induced by a single potentiation pulse of amplitude $I_{\text{prog}} = 100 $ $\mu$A over 1,000 trials (Fig. 2[d], top panel). To obtain more reliable statistics, the same measurement is done on 1,000 different devices and the values of the mean and standard deviation of conductance changes averaged over the 1,000 devices are 0.84 $\mu$S and 0.799 $\mu$S, respectively. To quantify the inter-device variability, we monitor the conductance change across the 1,000 devices induced by a single potentiation pulse (Fig. 2[d], bottom panel). We then repeat the same experiment over 1,000 trials and the values of the mean and standard deviation averaged over the 1,000 trials are 0.84 $\mu$S and 0.809 $\mu$S, respectively. These experiments show that the standard deviation of the conductance change due to intra-device variability is almost as large as that due to the inter-device variability. The rather surprising finding is that the randomness in the conductance change is to a large extent intrinsic to the physical characteristic of the device implies that improvements in array-level variability will not necessarily be effective in reducing the randomness.

The characterization work presented so far highlights the challenges associated with synaptic realizations using PCM devices and these can be generalized to other memristive technologies. The limited dynamic range, the asymmetric and nonlinear conductance response, the granularity and the randomness associated with conductance changes all pose challenges for realizing neural networks using memristive synapses. We now show how our concept of multi-memristive synapses can help in addressing some of those challenges. Experimental characterizations of multi-memristive synapses comprising 1, 3, and 7 PCM devices per synapse are shown in Fig. 3(a). The conductance change is averaged over 1,000 synapses. One selection clock with an increment rate of one arbitrates the device selection. As the total conductance is the sum of the individual conductance values, the dynamic range scales linearly with the number of devices per synapse. Alternatively, for a learning algorithm requiring a fixed dynamic range, multi-memristive synapses can improve the effective conductance change granularity. In addition, the mean cumulative conductance change is linear over an extended range of potentiation pulses compared to a single device. With multiple devices, we can also partially mitigate the challenge of asymmetric conductance response. At any instance, only one device is depressed, which implies that the effective synaptic conductance decreases gradually in several steps instead of the abrupt decrease observed in a single device. Moreover, using the depression frequency clock, the cumulative conductance change for potentiation and depression can be made approximately symmetric by adjusting the frequency of depression events. Finally, Fig. 3(b) shows that both the mean and the variance of the conductance change scale linearly with increasing number of devices per synapse, leading to an overall increase in weight update resolution.

C. Applications of multi-memristive synapses in neural networks

The multi-memristive synapse architecture is applicable to a wide range of neural networks and learning schemes and is particularly suitable for neural networks in which an increased precision and reliability of weight updates is necessary to obtain higher performance [43]. To study the impact of multi-memristive synapses in the context of ANNs and SNNs, we developed a simplified device model that captures the finite conductance range as well as the granularity and randomness associated with the conductance changes of PCM device [33]. The model parameters are to a large extent based on the PCM characterization work presented in the preceding section. In the model, each device is assumed to have a limited conductance range between 0 $\mu$S and 10 $\mu$S. When applying a potentiation pulse, the conductance change is assumed to be a Gaussian random number with mean and standard deviation equal to 0.5 $\mu$S. The conductance changes associated with two different potentiation pulses are assumed to be independent. When applying a single depression pulse, the conductance of the device is set to zero in a deterministic manner. In the differential architecture, the conductance values are refreshed only when 90% of the maximum conductance of either $G_+$ or $G_-$ is reached.
First, we present results that show the performance of an ANN with multi-memristive synapses. The feedforward fully-connected network is trained with the backpropagation algorithm to perform a classification task on the MNIST data set of handwritten digits (33, 46) (see Fig. 4(a) and the methods section). The ideal classification performance of this network, assuming double-precision floating-point accuracy for the network parameters, is 97.8%. The synaptic weights are represented using the conductance values of a multi-memristive synapse model based on the device model described earlier. One selection clock is used for all the synapses of the network and the weight updates are done sequentially through all the synapses in the same order at every pass. The weight changes are only applied if they can be rounded to the minimal effective synaptic granularity, which is determined by the number of devices in the synapse. A large weight change is possible through the application of multiple potentiation pulses to the same device. As shown in Fig. 4(a), the classification accuracy improves with increasing number of devices per synapse. Using the differential architecture, we can achieve test accuracies exceeding 95.8%. Remarkably, accuracies exceeding 93% are possible even with the non-differential architecture. This performance is achieved with the help of the depression clock that compensates for the device asymmetry. Note that, a non-differential architecture would have a reduced implementation complexity compared to its differential counterpart because the refresh operation, which requires reading and reprogramming G+ and G-, can be completely avoided. Note that when a single device is used to represent a synapse, the classification accuracy is below 45%, which clearly illustrates the efficacy of the proposed architecture.

In a second investigation, we study an SNN with multi-memristive synapses to perform the same task of MNIST digit classification, but with unsupervised learning (34) (see Fig. 4(b) and the methods section). The weight updates are performed using an STDP rule: the synapse is potentiated whenever a presynaptic spike appears prior to a postsynaptic spike and depressed whenever a presynaptic spike appears after the postsynaptic spike. The amount of weight increase (decrease) within the potentiation (depression) window is constant and is independent of the timing difference between the spikes. The classification performance of the network trained with this rule using double-precision floating point accuracy for the network parameters is 74.7%. Simulation studies were performed with the same model used for the ANN. A single selection clock is used to arbitrate all synapses along with a potentiation frequency clock to reduce the effective potentiation events and a depression clock in the non-differential architecture to improve the asymmetric conductance response. The network can classify more than 70% of the digits correctly for $N > 11$ with both differential and non-differential architectures, whereas the network that uses only a single device per synapse has a classification accuracy below 20%.

In both cases, we see that the multi-memristive synapse significantly outperforms a synapse based on a single device clearly illustrating the effectiveness of the proposed architecture. Moreover, the fact that the non-differential architecture achieves a comparable performance as the differential architecture is promising for synaptic realizations using highly asymmetric devices. A non-differential architecture would have a reduced implementation complexity compared to its differential counterpart because the refresh operation, which requires reading and reprogramming G+ and G-, can be completely avoided.

### D. Experimental results: Spike-based correlation detection based on multi-memristive synapses

Next, we present an experimental demonstration of the multi-memristive synapse architecture in an SNN that detects temporal correlations in event-based data streams. The SNN comprises a neuron interfaced to plastic synapses, with each one receiving an event-based data stream as presynaptic input spikes (33, 46) (see Fig. 5(a) and the methods section for more details). A subset of the data streams are mutually temporally correlated while the rest are uncorrelated. When the input streams are applied, postsynaptic currents are generated at the synapses which received a spike. These resulting postsynaptic outputs are accumulated at the neuron. When this neuronal membrane potential exceeds a threshold, the output neuron fires generating a post-synaptic spike. The synaptic weights are updated using an STDP rule; synapses which have a postsynaptic spike occurring after a presynaptic spike within a time window get potentiated, whereas the synapses which receive a postsynaptic spike prior to the presynaptic spike get depressed. As it is more likely that the temporally correlated inputs eventually govern the neuronal firing events, the conductance of synapses receiving correlated inputs is expected to increase, whereas that of synapses whose input are uncorrelated is expected to decrease. Hence, the final steady-state distribution of the weights should display a separation between synapses receiving correlated and uncorrelated inputs.

In our experiment, multi-memristive synapses with PCM devices are used to store the synaptic weights. The network comprises 1,000 synapses, of which only 100 receive temporally correlated inputs with a correlation coefficient of 0.75. The difficulty in detecting whether an input is correlated or not increases both with decreasing $c$ and decreasing number of correlated inputs. Hence, detecting only 10% correlated inputs with $c < 1$ is a fairly difficult task and requires precise synaptic weight changes for the network to be trained effectively (50). Each synapse comprises $N$ PCM devices organized in a non-differential architecture. The sum of the conductance values associated with the constituent devices is mapped to a synaptic weight. The selection clock is incremented by a unitary step after each weight update. During the weight update at a synapse, a single potentiation pulse or a single depression pulse is applied to one of the devices the selection clock points to. A depression frequency clock of length two is incremented by a unitary step after each weight update. The selection clock is used for all the synapses of the network and the weight updates are done sequentially through all the synapses in the same order at every pass. The weight changes are only applied if they can be rounded to the minimal effective synaptic granularity, which is determined by the number of devices in the synapse. A large weight change is possible through the application of multiple potentiation pulses to the same device. As shown in Fig. 5(a), the classification accuracy improves with increasing number of devices per synapse. Using the differential architecture, we can achieve test accuracies exceeding 95.8%. Remarkably, accuracies exceeding 93% are possible even with the non-differential architecture. This performance is achieved with the help of the depression clock that compensates for the device asymmetry. Note that, a non-differential architecture would have a reduced implementation complexity compared to its differential counterpart because the refresh operation, which requires reading and reprogramming G+ and G-, can be completely avoided. Note that when a single device is used to represent a synapse, the classification accuracy is below 45%, which clearly illustrates the efficacy of the proposed architecture.

In a second investigation, we study an SNN with multi-memristive synapses to perform the same task of MNIST digit classification, but with unsupervised learning (34) (see Fig. 4(b) and the methods section). The weight updates are performed using an STDP rule: the synapse is potentiated whenever a presynaptic spike appears prior to a postsynaptic spike and depressed whenever a presynaptic spike appears after the postsynaptic spike. The amount of weight increase (decrease) within the potentiation (depression) window is constant and is independent of the timing difference between the spikes. The classification performance of the network trained with this rule using double-precision floating point accuracy for the network parameters is 74.7%. Simulation studies were performed with the same model used for the ANN. A single selection clock is used to arbitrate all synapses along with a potentiation frequency clock to reduce the effective potentiation events and a depression clock in the non-differential architecture to improve the asymmetric conductance response. The network can classify more than 70% of the digits correctly for $N > 11$ with both differential and non-differential architectures, whereas the network that uses only a single device per synapse has a classification accuracy below 20%.

In both cases, we see that the multi-memristive synapse significantly outperforms a synapse based on a single device clearly illustrating the effectiveness of the proposed architecture. Moreover, the fact that the non-differential architecture achieves a comparable performance as the differential architecture is promising for synaptic realizations using highly asymmetric devices. A non-differential architecture would have a reduced implementation complexity compared to its differential counterpart because the refresh operation, which requires reading and reprogramming G+ and G-, can be completely avoided.

### D. Experimental results: Spike-based correlation detection based on multi-memristive synapses

Next, we present an experimental demonstration of the multi-memristive synapse architecture in an SNN that detects temporal correlations in event-based data streams. The SNN comprises a neuron interfaced to plastic synapses, with each one receiving an event-based data stream as presynaptic input spikes (33, 46) (see Fig. 5(a) and the methods section for more details). A subset of the data streams are mutually temporally correlated while the rest are uncorrelated. When the input streams are applied, postsynaptic currents are generated at the synapses which received a spike. These resulting postsynaptic outputs are accumulated at the neuron. When this neuronal membrane potential exceeds a threshold, the output neuron fires generating a post-synaptic spike. The synaptic weights are updated using an STDP rule; synapses which have a postsynaptic spike occurring after a presynaptic spike within a time window get potentiated, whereas the synapses which receive a postsynaptic spike prior to the presynaptic spike get depressed. As it is more likely that the temporally correlated inputs eventually govern the neuronal firing events, the conductance of synapses receiving correlated inputs is expected to increase, whereas that of synapses whose input are uncorrelated is expected to decrease. Hence, the final steady-state distribution of the weights should display a separation between synapses receiving correlated and uncorrelated inputs.

In our experiment, multi-memristive synapses with PCM devices are used to store the synaptic weights. The network comprises 1,000 synapses, of which only 100 receive temporally correlated inputs with a correlation coefficient of 0.75. The difficulty in detecting whether an input is correlated or not increases both with decreasing $c$ and decreasing number of correlated inputs. Hence, detecting only 10% correlated inputs with $c < 1$ is a fairly difficult task and requires precise synaptic weight changes for the network to be trained effectively (50). Each synapse comprises $N$ PCM devices organized in a non-differential architecture. The sum of the conductance values associated with the constituent devices is mapped to a synaptic weight. The selection clock is incremented by a unitary step after each weight update. During the weight update at a synapse, a single potentiation pulse or a single depression pulse is applied to one of the devices the selection clock points to. A depression frequency clock of length two is incremented by a unitary step after each weight update. The selection clock is used for all the synapses of the network and the weight updates are done sequentially through all the synapses in the same order at every pass. The weight changes are only applied if they can be rounded to the minimal effective synaptic granularity, which is determined by the number of devices in the synapse. A large weight change is possible through the application of multiple potentiation pulses to the same device. As shown in Fig. 5(a), the classification accuracy improves with increasing number of devices per synapse. Using the differential architecture, we can achieve test accuracies exceeding 95.8%. Remarkably, accuracies exceeding 93% are possible even with the non-differential architecture. This performance is achieved with the help of the depression clock that compensates for the device asymmetry. Note that, a non-differential architecture would have a reduced implementation complexity compared to its differential counterpart because the refresh operation, which requires reading and reprogramming G+ and G-, can be completely avoided. Note that when a single device is used to represent a synapse, the classification accuracy is below 45%, which clearly illustrates the efficacy of the proposed architecture.
devices. This holds true even for lower values of correlation coefficient. With $N = 1$, there are strong abrupt fluctuations in the evolution of the conductance values because of the abrupt depression events as shown in Fig. 5(c). With $N = 7$, a more gradual potentiation and depression behavior is observed. At the end of the experiment, for $N = 7$ the synapses receiving correlated and uncorrelated inputs can be perfectly separated. In contrast, the weights of correlated inputs display a wider weight distribution and there are numerous misclassified weights for $N = 1$.

The multi-memristive synapse architecture is also scalable to larger network sizes. To demonstrate this, we repeated the above correlation experiment with 144,000 input streams, and with 7 PCM devices per synapse, resulting in more than one million PCM devices in the network. As before, 10% of the synapses received correlated inputs with a correlation coefficient of 0.75. The synapses are arranged in a non-differential architecture, and a depression frequency clock of length 2 is used. As shown in Fig. 5(d), well-separated synaptic distributions have been achieved in the network at the end of the experiment.

II. DISCUSSION

We presented a hardware synapse architecture in which multiple memristive devices are used to efficiently implement in-situ learning and synaptic communication in neural networks. This concept bears similarities to several aspects of neural connectivity in biology, as biological neural connections also comprise multiple sub-units. For instance, in the central nervous system, a presynaptic neuron may form multiple synaptic terminals (so-called boutons) to connect to a single postsynaptic neuron. Moreover, each biological synapse contains a plurality of presynaptic release sites and postsynaptic ion channels. Furthermore, our implementation of plasticity in the composite hardware synapse through changes in the individual memristors is analogous to the fact that each of the synaptic connections between a pair of biological neurons is individually plastic, as are the individual ion channels of a synaptic connection. The involvement of progressively larger numbers of memristive devices during the potentiation of the hardware synapse is analogous to the development of new synaptic connections during the late phase of biological long-term potentiation, as well as to the development of new ion channels in a potentiated synapse.

A significant advantage of the proposed multi-memristive synapse is its crossbar compatibility. Crossbar arrays built with memristive devices can offer a dense hardware platform as a large number of devices can be organized in a small chip area. By placing the devices that constitute a single synapse along the bit lines of a crossbar, it is possible to sum up the currents using Kirchhoff’s law and obtain the total synaptic current without the need for any additional circuitry. Another advantage of the multi-memristive synapse concept is that the associated area/power penalty is not significant. The neuron circuitry of the crossbar array, which typically consumes a larger area than the crossbar array, remains the same except for the additional circuitry needed for arbitration. For example, assuming a neuron area of $10^9 F^2$ (the area of a neuron from TrueNorth chip), without time-division multiplexing and a crossbar array of 1000x1000 synapses, the total neuron area would be $10^3 \times 10^9 F^2$. A comparable total synapse area would be obtained assuming 1,000F$^2$ per synapse, which should be sufficient to realize a multi-memristive synapse with $N \leq 10$. Moreover, the power dissipation associated with programming the memristive devices is orders of magnitude higher than that required for the read operation. As in our synaptic concept, only one device is programmed at each instance of synaptic update, the power dissipation due to programming remains identical to when a single device per synapse is used. Finally, because even a single global clock can be used for arbitrating a whole array, the additional area/power overhead is expected to be minimal.

The proposed architecture also offers several advantages in terms of reliability. Even if one of the devices of a synapse fails, the other devices will continue to perform synaptic efficacy and plasticity. In addition, each device within a synapse gets programmed less frequently than if a single device would be used, which effectively increases the overall lifetime of a multi-memristive synapse compared to a single device implementation. The potentiation and depression frequency clocks reduce the effective number of programming operations of a synapse, further improving endurance-related issues.

The device selection in the multi-memristive synapse is performed based on the arbitration module alone, and without any knowledge of the conductance values of the individual devices, thus there is a non-zero probability that a potentiation (depression) pulse will not result in an actual potentiation (depression) of the synapse. This would effectively translate into a weight-dependent plasticity whereby the probability to potentiate reduces with increasing synaptic weight and the probability to depress reduces with decreasing synaptic weight. This attribute could affect the overall performance of a neural network. For example, weight-dependent plasticity has been shown to impact the classification accuracy negatively in an ANN. In contrast, a study suggests that it can stabilize an SNN intended to detect temporal correlations.

In summary, we propose a novel synaptic architecture comprising multiple memristive devices with non-ideal characteristics to efficiently implement learning in neural networks. This architecture is shown to overcome several significant challenges that are characteristic to nanoscale memristive devices proposed for synaptic implementation, such as the asymmetric conductance response, limitations in resolution and dynamic range, as well as device-level variability. The architecture is applicable to a wide range of neural networks and memristive technologies and is crossbar-compatible. The high potential of the concept is demonstrated experimentally in a large-scale SNN performing unsupervised learning. The proposed architecture and its experimental demonstration hence are a significant step towards the realization of highly efficient, large-scale neural networks based on memristive devices with typical experimentally observed non-ideal characteristics.
III. METHODS

A. Experimental platform

The experimental hardware platform is built around a prototype PCM chip with 3 million devices with a 4-bank inter-leaved architecture. The mushroom-type PCM devices are based on doped Ge$_2$Sb$_2$Te$_5$ (GST) and were integrated into the prototype chip in 90 nm CMOS technology. The radius of the bottom electrode is approximately 20 nm, and the thickness of the phase change material is approximately 100 nm. A thin oxide n-type field effect transistor (FET) enables access to each PCM device. The chip also integrates the circuitry for addressing, an 8-bit on-chip analog-to-digital converter (ADC) for readout, and voltage- or current-mode programming. An analog-front-end (AFE) board is connected to the chip and accommodates digital-to-analog converters (DACs) and ADCs, discrete electronics, such as power supplies, voltage and current reference sources. An FPGA board with embedded processor and Ethernet connection implements the overall system control and data management.

B. PCM characterization

For the experiment of Fig. 2(b), measurements are done on 10,000 devices. All devices are initialized to approximately 0.1 $\mu$S with an iterative procedure. In the experiment, 20 potentiation pulses with a duration of 50 ns and varying amplitudes are applied. After each potentiation pulse, the devices are read 50 times repeatedly in approximately 5 sec intervals. The reported device conductance for a potentiation pulse is the average conductance obtained by the 50 consecutive read operations. This method is used to minimize the impact of drift and read noise. At the end of the experiment, approximately 300 devices were excluded for Fig. 2(b) because they had an initial conductance of less than 0.1 $\mu$S and a final conductance after 20 potentiation pulses of more than 30 $\mu$S.

In the measurements for Fig. 2(c), 10,000 devices are used. The data is obtained after initializing the device conductances to 5 $\mu$S by an iterative procedure. Next, potentiation (depression) pulses of varying amplitude are applied. Duration of the potentiation pulses are 50 ns. Every potentiation (depression) pulse is followed by 50 read operations that are done approximately 5 sec apart. The device conductance is averaged for the 50 read operations.

In the experiments of Fig. 2(d), 1,000 devices used. All devices are initialized to approximately 0.1 $\mu$S with an iterative procedure. This is followed by 4 potentiation pulses of amplitude $I_{\text{prog}}=100 \mu$A and width 50 ns. After the latter 2 potentiation pulses, devices are read 20 times with each read approximately 1.5 sec apart. The device conductances for 20 read operations are averaged. The difference between the averaged conductances for the 3rd and 4th potentiation pulses is defined as the conductance change. This experimental sequence is repeated on the same devices for 1,000 times so that 1,000 conductance changes are measured for each device.

For the experiments of Fig. 3 measurements are done on 1,000, 3,000, and 7,000 devices for $N = 1, 3$ and 7 respectively. Device conductances are initialized to 5 $\mu$S by an iterative procedure. Next, for potentiation, programming pulses of amplitude $100 \mu$A and width 50 ns are applied. For depression, programming pulses of amplitude $450 \mu$A are applied. After each potentiation (depression) pulse, device conductances are read 50 times and averaged.

C. Simulation of neural networks

The ANN contains 784 input neurons, 250 hidden layer neurons, and 10 output neurons. In addition, there is one bias neuron at the input layer and one bias neuron at the hidden layer. For training, all 60,000 images from the MNIST training set are used in the order they appear in the database over 10 epochs. All 10,000 images from the MNIST test set are shown for testing. The input images from the MNIST set are greyscale pixels with values ranging from 0 to 255 and have a size of 28 times 28. Each input layer neurons receives input from one image pixel, and the input is the pixel intensity scaled by 255 in double-precision floating point. The neurons of the hidden and the output layers are sigmoid neurons. Synapses are multi-memristive, and each input layer neurons receives input from one image pixel, and the input is the pixel intensity scaled by 255 in double-precision floating point. The neurons of the hidden and the output layers are sigmoid neurons. Synapses are multi-memristive, and each synapse comprises $N$ devices. The total synaptic weight $w$ is in the interval $[-1, 1]$. The devices in a synapse are arranged using either a non-differential or a differential architecture. In the non-differential architecture, the weight of each device $w_n$ is in the interval $[-1/N, 1/N]$ and initialized randomly with a uniform distribution in the interval $[-1/N, 1/N]$. The total synaptic weight is calculated as $\sum_{n=1}^{N} w_n$. The random seed is fixed for all simulations. In the differential architecture, $N$ devices are arranged in two sets, where $\frac{N}{2}$ devices represent $G_+$ and $\frac{N}{2}$ devices represent $G_-$. The weight of each device $w_{n,+/-n,-}$ for $n = 1, 2, ..., \frac{N}{2}$ is in the range between $[0, \frac{1}{N}]$. The weight of each device is initialized randomly with a uniform distribution in the interval $[0, \frac{1}{N}]$. The total synaptic weight is $(\sum_{n=1}^{N} w_{n,+}) - (\sum_{n=1}^{N} w_{n,-})$. For double floating-point precision, the synaptic weights are initialized with a uniform distribution in the interval $[-0.5, 0.5]$. The weight updates $\Delta w$ are done sequentially to synapses and the selection clock is incremented by one after each weight update. If $\Delta w > 0$, the synapse will undergo potentiation. In both architectures, each potentiation pulse induces a weight change of size $\epsilon = \frac{0.1}{N}$ and the number of potentiation pulses to be applied are calculated by rounding $\frac{\Delta w}{\epsilon}$. Then, for each potentiation pulse, an independent Gaussian random number with mean and standard deviation of...
\( \varepsilon \) is added. This weight change is applied to the device to which the selection clock points. If \( \Delta w < 0 \), the synapse will undergo a depression. In the differential architecture, a potentiation pulse is applied to a device from the set representing \( G_+ \) using the above-mentioned methodology. In the non-differential architecture, a depression pulse is applied to one of the devices pointed at by the selection clock if \( \Delta w < \varepsilon \). The weight of the device drops to its minimum value. For \( N > 1 \), we used a depression frequency clock of length 5. No depression frequency clock is used for \( N = 1 \). In the differential architecture, after the weight change has been applied for potentiation and depression, synapses are checked for the refresh operation. If there is a synapse which has \( w_{i, n} > 0.9 \) or \( w_{i, n} < 0.9 \), a refresh is done on that synapse. \( w \) is recorded and all devices in the synapse are set to 0. The programming will be done to devices of the set \( w_{i, n} \) if \( w > 0 \) or to devices of the set \( w_{i, n} \) if \( w < 0 \). The number of potentiation pulses to each device in the set is calculated by rounding \( \frac{\Delta w}{\varepsilon} \). Each device in the set receives an identical number of pulses. For each device in the set to be potentiated, one independent Gaussian random number with mean and standard deviation of \( \varepsilon \) is calculated for each of the potentiation pulses. The learning rate is 0.4 for all simulations.

The SNN comprises 784 input neurons and 50 output neurons. These synapses are multi-memristive and each synapse is composed of \( N \) memristive devices. The network is trained with all 60,000 images from the MNIST set over 3 epochs and tested with all 10,000 test images from the set. The simulation time step is 5 ms. Each input neuron receives input from one pixel of the input image. Each input image is presented for 350 ms, and the information regarding the intensity of each pixel is in the form of spikes. We create the input spikes using a Poisson distribution, where independent Bernoulli trials are conducted to determine whether there is a spike at a time step. A spike rate is calculated as \( \text{pixel intensity} \times 20 \text{ Hz} \). A spike is generated if (spike rate \( \times 5 \text{ ms} > x \)), where \( x \) is a uniformly distributed random number between 0 and 1. The input spikes create a current with the shape of a delta function at the corresponding synapse. The magnitude of this current is equal to the weight of the synapse. The synaptic weights \( w \in [0, 1] \), are learned with an SDTP rule\(^{[1]} \). The synapses are arranged in a non-differential or a differential architecture. In the non-differential architecture, \( w_n \) is in the interval \( [0, \frac{1}{N}] \). The weight of each device is initialized randomly with a uniform distribution in the interval \( \left[ \frac{2}{3N}, \frac{3}{3N} \right] \). The total synaptic weight is calculated as \( \sum_{n=1}^{N} w_n \). In the differential architecture, \( N \) devices are arranged in two sets. The weight of each device \( w_{n, n} \) for \( n = 1, 2, ..., \frac{N}{2} \) is in the range \( [0, \frac{1}{N}] \). The weight of each device is initialized randomly with a uniform distribution in the interval \( \left[ \frac{2}{3N}, \frac{1}{3N} \right] \). The total synaptic weight is \( (\sum_{n} w_{n+}) - (\sum_{n} w_{n-}) + 0.5 \). For double-floating-point precision, the synaptic weights are initialized with a uniform distribution in the interval \([-0.25, 0.75] \). At each simulation time step, the synaptic currents are summed at the output neurons and accumulated using a state variable \( X \). The output neurons are of the leaky integrate-and-fire type and they have a leak constant of \( \tau = 200 \text{ ms} \). Each output neuron has a spiking threshold. This spiking threshold is set initially to 0.08 (note that the sum of the currents is normalized by the number of input neurons) and is altered by homeostasis during training. An output neuron spikes when \( X \) exceeds the neuron threshold. Only one output neuron is allowed to spike at a single time step, and if the state variables of several neurons exceed their threshold, then the neuron whose state variable exceeds its threshold the most is the winner. The state variables of all other neurons are set to 0 if there is a spiking output neuron. If there is a postsynaptic spike, the synapses that received a presynaptic spike in the past 30 ms are potentiated. If there is a presynaptic spike the synapses that had a postsynaptic spike in the past 1.05 s are depressed. The weight change amount is constant for potentiation \( \Delta w_+ \) and depression \( \Delta w_- \), following a rectangular STDP rule. In the double-precision floating-point accuracy, \( \Delta w_+ = 0.005 \) and \( \Delta w_- = 0.0025 \). For simulations with the device model, \( \Delta w_+ = 0.01 \) and \( \Delta w_- = 0.006 \). The weight updates are done using the scheme described above. For the non-differential architecture, a depression pulse is applied when \( \Delta W < 0 \). The depression clock length is set to the floor of \( \frac{1}{N} \Delta W_+ \) for \( N > 1 \). In both the non-differential and the differential architectures, a potentiation frequency clock of length 2 is used. After 1000th input image, upon presentation of every two images, the output neuron thresholds are adjusted through homeostasis. The threshold increase for every output neuron is calculated as \( 0.0005 \times (A - T) \), where \( A \) is the activity of the neuron and \( T \) is the target firing rate. \( A \) is calculated as \( \frac{5 \text{ count}}{\text{count max}} \), where \( S \) is the sum of the neuron’s firing event in the past 100 examples. We define the \( T \) as \( \frac{5 \text{ count}}{\text{count max}} \), where 50 is the number of output neurons in the network. After training, the synaptic weights and the neuron thresholds are kept constant. To quantify how well the training is, we show all 60,000 images to the network, and the neuron that spikes the most often during the presentation of an image for 350 ms is recorded. The neuron is mapped to the class, i.e., to one of the 10 digits, for which it spiked the most. This mapping is then used to detect the classification accuracy when the test set is presented.

### D. Correlation detection experiment

The network for correlation detection comprises 1,000 plastic synapses connected to an output neuron. Each synapse is multi-memristive and consists of \( N \) devices. The synaptic weights \( w \in [0, 1] \) are learned with an SDTP rule\(^{[6]} \). Because of the hardware latency, we will use normalized units to describe time in the experiment. The experiment time steps are of size \( T_i = 0.1 \). Each synapse receives a stream of spikes, and the spikes have the shape of the delta function. 100 of the input spike stream are correlated. The correlated and the uncorrelated spike stream have equal rates of \( r_{\text{cor}} = r_{\text{uncor}} = 1 \). The correlated inputs share the outcome of a Bernoulli trial. This Bernoulli trial is described as \( B = x > 1 - r_{\text{cor}} \times T_i \), where \( x \) is a uniformly distributed random number. By using this event, the input spikes for the correlated streams are generated as \( B \times (r_{\text{cor}} \times T_s + \sqrt{e} \times (1 - r_{\text{cor}} \times T_i > x_1) + \sim B \times (r_{\text{cor}} \times T_s + (1 - \sqrt{e} > x_2)) \), where \( x_1 \) and \( x_2 \) are uniformly distributed random numbers, \( e \) is the correlation coefficient.
of value 0.75, and denotes the negation operation. The uncorrelated processes are generated as \( x_3 > 1 - r_{\text{uncor}} \cdot T_s \), where \( x_3 \) is a uniformly distributed random variable. These presynaptic spikes generate a current of the size of the synaptic weights. At every time step, the currents are summed and accumulated at the neuronal membrane variable \( X \). The neuron has a leak constant of \( T_s \) equal to the time step of the experiment. This leads to a complete leak of the \( X \) at every time step, and only the currents of the current time step contribute to the neuronal firing events. If \( X \) exceeds a threshold of 52, the output neuron fires. The weight update calculation follows an exponential STDP rule where the amount of potentiation is calculated as \( A_+ \cdot e^{-|\Delta t| / \tau_+} \) and depression is calculated as \(-A_- \cdot e^{-|\Delta t| / \tau_-} \). \( A_+ \), \( A_- \) are the learning rates, \( \tau_+ \), \( \tau_- \) are time constants, and \( \Delta t \) is the time difference between the presynaptic and postsynaptic spikes. We set \( 2 \cdot A_+ = 2 \cdot A_- = 0.004 \) and \( \tau_+ = \tau_- = 3 \cdot T_s \).

The weight storage and weight update operations are done on PCM devices. We access each PCM device sequentially for reading and programming. For initialization, a depression pulse is sent to all devices, followed by one potentiation pulse of amplitude \( I_{\text{prog}} = 120 \, \mu A \) and width 50 ns. Although the weight update is calculated using an exponential STDP rule, it is applied following a rectangular STDP rule. For potentiation, a single potentiation pulse of amplitude \( I_{\text{prog}} = 100 \, \mu A \) and width 100 ns is applied when \( \Delta w_+ \geq 0.001 \). For depression, a single depression pulse of amplitude \( I_{\text{prog}} = 440 \, \mu A \) is applied when \( \Delta w_- \leq -0.001 \). The potentiation and depression pulses are sent to one device from the multi-memristive synapse the selection clock points to. When applying depression pulses, a depression frequency clock of length 2 is used for \( N > 1 \). After each programming operation, the device conductances are read. The conductance value \( G \) of a device is converted to its synaptic weight as \( w_n = G \cdot N \cdot 9.5 \, \mu S \). The weights of the devices in a multi-memristive synapse are summed to calculate the total synaptic weight \( \sum_{n=1}^{N} w_n \).

For the large-scale experiment, 144,000 synapses are trained, of which 14,400 receive correlated inputs. Each multi-memristive synapse comprises \( N = 7 \) devices, and a total of 1,008,000 PCM devices are used for this experiment. The same network parameters as in the small-scale experiment are used, except for the neuron threshold. The neuron threshold is scaled with the number of synapses and is set to 7488. The learning algorithm and conductance-to-weight conversion is identical to the small-scale experiment.
REFERENCES

1 LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
2 Schrempp, J. et al. A wafer-scale neuromorphic hardware system for large-scale neural modeling. In *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 1947–1950 (2010).
3 Pinkas, E. et al. SpiNNaker: A 1-W 18-core system-on-chip for massively-parallel neural network simulation. *IEEE Journal of Solid-State Circuits* **48**, 1943–1953 (2013).
4 Merolla, P. A. et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345**, 668–673 (2014).
5 Qiao, N. et al. A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128K synapses. *Frontiers in Neuroscience* **9**, 141 (2015).
6 Beck, A., Bednorz, J., Gerber, C., Rosell, C. & Widmer, D. Reproducible switching effect in thin oxide films for memory applications. *Applied Physics Letters* **77**, 139–141 (2000).
7 Strukov, D. B., Snider, G. S., Stewart, D. R. & Williams, R. S. The missing memristor found. *Nature* **453**, 80–83 (2008).
8 Chua, L. Resistance switching memories are memristors. *Applied Physics A* **102**, 765–783 (2011).
9 Wong, H. S. P. & Salauhadin, S. Memory leads the way to better computing. *Nature Nanotechnology* **10**, 191–194 (2015).
10 Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. In Rumelhart, D. E. & McClelland, J. L. (eds.) *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, vol. 1: Foundations, 318–362 (MIT Press, Cambridge, MA, 1986).
11 Markram, H., Lübke, J., Frotscher, M. & Sakmann, B. Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science* **275**, 213–215 (1997).
12 Anwani, N. & Rajendran, B. NormAD - normalized approximate descent based supervised learning rule for spiking neurons. In *Proc. International Joint Conference on Neural Networks (IJCNN)*, 1–8 (2015).
13 Saighi, S. et al. Plasticity in memristive devices for spiking neural networks. *Frontiers in Neuroscience* **9**, 51 (2015).
14 Burr, G. W. et al. Integration of nanoscale memristor synapses in neuromorphic computing architectures. *Advances in Physics: X* **2**, 89–124 (2016).
15 Rajendran, B. & Alibart, F. Neuromorphic computing based on emerging memory technologies. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* **6**, 198–211 (2016).
16 Qin, T. et al. Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials* **10**, 591–595 (2011).
17 Yen, C. Y., Jeayasingh, R., Kuzum, D. & Wong, H. S. P. An electronic synapse device based on metal oxide resistive switching memory for neuromorphic computation. *IEEE Transactions on Electron Devices* **58**, 2729–2737 (2011).
18 Ambrogio, S. et al. Neuromorphic learning and recognition with one-transistor-one-resistor synapses and bistable metal oxide RRAM. *IEEE Transactions on Electron Devices* **63**, 1508–1515 (2016).
19 Covi, E. et al. Analog memristive synapse in spiking networks implementing unsupervised learning. *Frontiers in Neuroscience* **10**, 482 (2016).
20 van de Burgt, Y. et al. A non-volatile organic electrochemical device as a low-voltage artificial synapse for neuromorphic computing. *Nature Materials* **16**, 414–418 (2017).
21 Wang, Z. et al. Memristors with diffusive dynamics as synaptic elements for neuromorphic computing. *Nature Materials* **16**, 101–108 (2017).
22 Boynt, S. et al. Learning through ferroelectric domain dynamics in solid-state synapses. *Nature Communications* **8**, 14736 (2017).
23 Vincent, A. F. et al. Spin-transfer torque magnetic memory as a stochastic memristive synapse for neuromorphic systems. *IEEE Transactions on Biomedical Circuits and Systems* **9**, 166–174 (2015).
24 Kuzum, D., Jeayasingh, R. G. D., Lee, B. & Wong, H. S. P. Nanoelectronic programmable synapses based on phase change materials for brain-inspired computing. *Nano Letters* **12**, 2179–2186 (2012).
25 Ambrogio, S. et al. Unsupervised learning by spike timing dependent plasticity in phase change memory (PCM) synapses. *Frontiers in Neuroscience* **10**, 56 (2016).
26 Alibart, F., Zamanidoost, E. & Strukov, D. B. Pattern classification by memristive crossbar circuits using ex situ and in situ training. *Nature Communications* **4**, 2072 (2013).
27 Indiveri, G., Linares-Barranco, B., Legenstein, R., Deligeorgis, G. & Prodromakis, T. Integration of nanoscale memristor synapses in neuromorphic computing architectures. *NanoTechnology* **24**, 384010 (2013).
28 Prezioso, M. et al. Training and operation of an integrated neuromorphic network based on metal-oxide memristors. *Nature* **521**, 61–64 (2015).
29 Kim, S. et al. NVM neuromorphic core with 64K-cell (256-by-256) phase memory synaptic array with on-chip neuron circuits for continuous in-situ learning. In *IEEE International Electron Devices Meeting (IEDM)*, 17–1 (IEEE, 2015).
30 Mostafa, H. et al. Implementation of a spike-based perceptron learning rule using TiO$_2$-a memristors. *Frontiers in Neuroscience* **9**, 357 (2015).
31 Tuma, T., Le Gallo, M., Sebastian, A. & Eleftheriou, E. Detecting correlations using phase-change neurons and synapses. *IEEE Electron Device Letters* **37**, 1238–1241 (2016).
32 Wozniak, S., Tuma, T., Pantazi, A. & Eleftheriou, E. Learning spatio-temporal patterns in the presence of input noise using phase-change memristors. In *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, 365–368 (2016).
33 Burr, G. W. et al. Experimental demonstration and tolerancing of a large-scale neural network (165 000 synapses) using phase-change memory as the synaptic weight element. *IEEE Transactions on Electron Devices* **62**, 3498–3507 (2015).
34 Koelmans, W. W. et al. Projected phase-change memory devices. *Nature Communications* **6**, 8181 (2015).
35 Fuller, E. J. et al. Li-ion synaptic transistor for low power analog computing. *Advanced Materials* **29** (2017).
36 Suri, M. et al. Phase change memory as synapse for ultra-dense neuromorphic systems: Application to complex visual pattern extraction. In *Proc. IEEE International Electron Devices Meeting (IEDM)*, 4.4.1–4.4.4 (2011).
37 Bill, J. & Legenstein, R. A compound memristive synapse model for statistical learning through STDP in spiking neural networks. *Frontiers in Neuroscience* **8**, 412 (2014).
38 Garbin, D. et al. HfO$_2$-based OxRAM devices as synapses for convolutional neural networks. In *Proc. IEEE International Electron Devices Meeting (IEDM)*, 2494–2501 (2015).
39 Boybat, I. et al. Recent progress in phase-change memory technology. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* **6**, 146–162 (2016).
40 Sebastian, A., Le Gallo, M. & Krebs, D. Crystal growth within a phase change memory cell. *Nature Communications* **5**, 4314 (2014).
41 Close, G. F. et al. Device, circuit and system-level analysis of noise in multi-bit phase-change memory. In *Proc. IEEE International Electron Devices Meeting (IEDM)*, 29.5.1–29.5.4 (2010).
42 Boybat, I. et al. Stochastic weight updates in phase-change memory-based synapses and their influence on artificial neural networks. In *Proc. 13th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, 13–16 (2017).
ACKNOWLEDGMENTS

We would like to thank N. Papandreou, U. Egger, S. Wozniak, S. Sidler, A. Pantazi and G. Burr for technical input. I. B. and T. M. acknowledge financial support from the Swiss National Science Foundation (SNSF). A.S. would like to acknowledge partial financial support from the European Research Council (ERC) grant no. 682675.

AUTHOR CONTRIBUTIONS

I.B., M. L. G., T. T. and A. S. designed the concept. I. B., N. S. R. and T. M. performed the simulations. I. B., N. S. R., M. L. G. and A. S. performed the experiments. T. M and T. P. provided critical insights. I. B., M. L. G., and A. S. co-wrote the manuscript with input from the other authors. A. S., Y. L., B. R. and E. E. supervised the work.
FIG. 1. The multi-memristive synapse concept. (a) The net synaptic weight of a multi-memristive synapse is represented by the combined conductance \( \sum G_n \) of multiple memristive devices. To realize synaptic efficacy, a voltage signal, \( V \), is applied to all devices. The resulting current flowing through each device is summed up to generate the synaptic output. (b) To capture synaptic plasticity, only one of the devices is selected at any instance of synaptic update. The synaptic update is induced by altering the conductance of the selected device as dictated by a learning algorithm. This is achieved by applying a suitable programming pulse to the selected device. (c) A clock-based arbitration scheme is used to select the devices that get programmed to achieve synaptic plasticity. A global selection clock whose length is equal to the number of devices representing a synapse is used. At any instance of synaptic update, the device pointed to by the selection clock is programmed. Subsequently, the selection clock is incremented by a fixed amount. In addition to the selection clock, independent potentiation and depression frequency clocks can serve to control the frequency of the potentiation or depression events.
FIG. 2. Synapses based on phase change memory. (a) A phase change memory device consists of a top electrode, phase change material and a bottom electrode. The phase change material in an as-fabricated device is found in the crystalline phase. A depression pulse creates an amorphous region that results in an abrupt drop in conductance, irrespective of the original state of the device. The crystalline region can gradually be increased by the application of potentiation pulses. (b) Evolution of mean conductance as a function of the number of pulses for different programming current amplitudes \(I_{\text{prog}}\). Each curve is obtained by averaging the conductance measurements from over 9,700 devices. The maximum conductance value that can be achieved is approximately 10 \(\mu\text{S}\) using \(I_{\text{prog}} = 100 \mu\text{A}\). Shown in inset is a transmission electron microscopy (TEM) image of a phase change memory device used in this study. (c) Mean cumulative conductance change observed upon the application of repeated potentiation and depression pulses. The initial conductance of the devices is approximately 5 \(\mu\text{S}\). The potentiation pulses result in gradual conductance changes; whereas a depression pulse causes an abrupt conductance decrease. (d) The top panel shows a representative distribution of 1,000 conductance changes induced by a single pulse on a single PCM device. The same measurement was repeated on 1,000 different PCM devices and the mean \((\mu)\) and standard deviation \((\sigma)\) averaged over the 1,000 devices are shown in the inset. The bottom panel shows a representative distribution of one conductance change induced by a single pulse on 1,000 devices. The same measurement was repeated for 1,000 conductance changes and the mean and standard deviation averaged over the 1,000 conductance changes are shown in the inset. It can be seen that inter- and intra-device variability is comparable. The negative conductance changes are attributed to drift variability.
FIG. 3. Multi-memristive synapses based on phase change memory. (a) The mean cumulative conductance change is experimentally obtained for synapses comprising 1, 3 and 7 PCM devices. The measurements are based on 1,000 synapses, whereby each individual device is initialized to a conductance of approximately 5 µS. For potentiation, a programming pulse of $I_{\text{prog}} = 100 \mu A$ was used, whereas for depression, a programming pulse of $I_{\text{prog}} = 450 \mu A$ was used. For depression, the conductance response can be made more symmetric by adjusting the length of the depression frequency clock. (b) Distribution of the cumulative conductance change after the application of the 10th, 30th and 70th potentiation pulse for synapses with 1, 3, and 7 PCM devices per synapse. The mean ($\mu$) and the variance ($\sigma^2$) scale almost linearly with number of devices per synapse, leading to a higher weight update resolution.
FIG. 4. Applications of multi-memristive synapses in neural networks. (a) An artificial neural network is trained using backpropagation to perform handwritten digit classification. Bias neurons are used for the input and hidden neuron layers (white). A multi-memristive synapse model is used to represent the synaptic weights in simulations. Increasing the number of devices in multi-memristive synapses (both in the differential and the non-differential architecture) improves the test accuracy. The dotted line shows the test accuracy obtained from a double-precision floating point software implementation. (b) A spiking neural network is trained using an STDP-based learning rule for handwritten digit classification. Here again, a multi-memristive synapse model is used to represent the synaptic weights in simulations where the devices are arranged in the differential or the non-differential architecture. The network can classify digits with higher accuracies with increasing number of devices per synapse. The dotted line shows the test accuracy obtained from a double-precision floating point implementation.
FIG. 5. **Experimental demonstration of multi-memristive synapses used in a spiking neural network.** (a) A spiking neural network is trained to perform the task of temporal correlation detection through unsupervised learning. Our network consists of 1,000 synapses connected to one integrate-and-fire (I&F) neuron. The synapses receive event-based data streams generated with Poisson distributions as presynaptic spikes. 100 of the synapses receive correlated data streams with a correlation coefficient of 0.75, whereas the rest of the synapses receive uncorrelated data streams. The correlated and the uncorrelated data streams both have the same rate. The resulting postsynaptic outputs are accumulated at the neuronal membrane. The neuron fires, i.e., sends an output spike, if the membrane potential exceeds a threshold. The weight update amount is calculated using an exponential STDP rule based on the timing of the presynaptic and postsynaptic spikes. A potentiation (depression) pulse with fixed amplitude is applied if the desired weight change is higher (lower) than threshold. (b) The synaptic weights are shown for synapses comprising $N=1$, 3 and 7 PCM devices at the end of the experiment (5,000 time steps). It can be seen that the weights of the synapses receiving correlated inputs tend to be larger compared to the weights of those receiving uncorrelated inputs. The weight distribution shows a clearer separation with increasing $N$. (c) The weight evolution of synapses in the first 300 time steps of the experiment is shown. The weight evolves more gradually with increasing number of devices per synapse. (d) The synaptic weight distribution of an SNN comprising 144,000 multi-memristive synapses with 7 PCM devices is shown at the end of an experiment (3,000 time steps). 14,400 synapses receive correlated input data streams with a correlation coefficient of 0.75. This large-scale experiment employed a total of 1,008,000 PCM devices.