Experimental Demonstration and Tolerancing of a Large-Scale Neural Network (165 000 Synapses) Using Phase-Change Memory as the Synaptic Weight Element

Geoffrey W. Burr, *Senior Member, IEEE*, Robert M. Shelby, Severin Sidler, Carmelo di Nolfo, Junwoo Jang, Irem Boybat, *Student Member, IEEE*, Rohit S. Shenoy, *Member, IEEE*, Pritish Narayanan, *Member, IEEE*, Kumar Virwani, *Member, IEEE*, Emanuele U. Giacometti, Bülent N. Kurdi, and Hyunsang Hwang, *Member, IEEE*

***Abstract*— Using two phase-change memory devices per synapse, a three-layer perceptron network with 164 885 synapses is trained on a subset (5000 examples) of the MNIST database of handwritten digits using a backpropagation variant suitable for nonvolatile memory (NVM) + selector crossbar arrays, obtaining a training (generalization) accuracy of 82.2% (82.9%). Using a neural network simulator matched to the experimental demonstrator, extensive tolerancing is performed with respect to NVM variability, yield, and the stochasticity, linearity, and asymmetry of the NVM-conductance response. We show that a bidirectional NVM with a symmetric, linear conductance response of high dynamic range is capable of delivering the same high classification accuracies on this problem as a conventional, software-based implementation of this same network.**

***Index Terms*— Artificial neural networks, Machine learning, Multilayer perceptrons, Nonvolatile memory, Phase change memory.**

I. Introduction

DENSE arrays of nonvolatile memory (NVM) and selector device pairs (Fig. 1) can implement neuro-inspired non-Von Neumann computing [1], [2], using pairs [2] of NVM devices as programmable (plastic) bipolar synapses.

Manuscript received May 4, 2015; revised May 17, 2015; accepted May 28, 2015. Date of publication July 7, 2015; date of current version October 20, 2015. The review of this paper was arranged by Editor J. S. Suehle.

G. W. Burr, R. M. Shelby, S. Sidler, C. di Nolfo, P. Narayanan, K. Virwani, and B. N. Kurdi are with IBM Research–Almaden, San Jose, CA 95120 USA (e-mail: gwburr@us.ibm.com; rshelby@us.ibm.com; severin.sidler@epfl.ch; carmelodinolfo@gmail.com; pnaraya@us.ibm.com; kvirwan@us.ibm.com; bulent@us.ibm.com).

J. Jang is with the Department of Creative IT Engineering, Pohang University of Science and Technology, Pohang 790-784, Korea (e-mail: junwoo410@gmail.com).

I. Boybat is with the École Polytechnique Fédérale de Lausanne, Lausanne 1015, Switzerland (e-mail: irem.boybat@epfl.ch).

R. S. Shenoy is with Intel, Santa Clara, CA 95054 USA (e-mail: rohits@gmail.com).

E. U. Giacometti was with IBM Research–Almaden, San Jose, CA 95120 USA (e-mail: giacometti.emanuele@gmail.com).

H. Hwang is with the Department of Material Science and Engineering, Pohang University of Science and Technology, Pohang 790-784, Korea (e-mail: hwanghs@postech.ac.kr).

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

Digital Object Identifier 10.1109/TED.2015.2439635

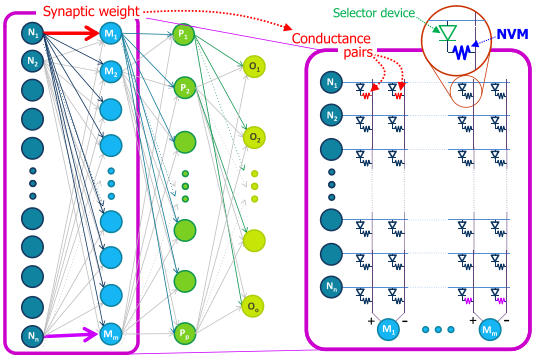


Fig. 1. Neuroinspired non-Von Neumann computing [1], [2], in which neurons activate each other through dense networks of programmable synaptic weights, can be implemented using dense crossbar arrays of NVM and selector device pairs.

Work to date has emphasized the spike-timing-dependent-plasticity (STDP) algorithm [1], [2], motivated by synaptic measurements in real brains. However, experimental NVM demonstrations have been limited in size ($\leq$100 synapses), and few results have reported quantitative performance metrics such as classification accuracy. Worse yet, it has been difficult to be sure whether the relatively poor metrics reported to date might be due to immaturities or inefficiencies in the STDP learning algorithm (as it is currently implemented), or if these results are truly reflective of problems introduced by imperfections in the NVM devices.

Unlike STDP, backpropagation is a widely used, well-studied method in training artificial neural networks (NNs), offering benchmarkable performance on datasets such as handwritten digits (MNIST) [3]. Although proposed earlier, it gained great popularity in the 1980s [3], [4], and with the advent of graphics processor units (GPUs), backpropagation now dominates the NN field. In this paper, we use backpropagation to train a relatively simple multilayer perceptron network (Fig. 2). During forward evaluation of this network, each layer's inputs (x_i) drive the next layer's neurons through a weight w_{ij} and a nonlinearity $f()$ (Fig. 2). Supervised learning occurs (Fig. 3) by then backpropagating the error term δ_j to adjust each weight w_{ij}. A three-layer network is capable of accuracies, on

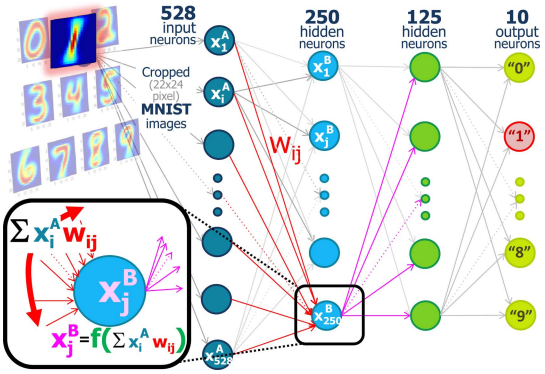


Fig. 2. In forward evaluation of a multilayer perceptron, each layer's neurons drive the next layer through a weight w_{ij} and a nonlinearity $f()$. Input neurons are driven by pixels from successive MNIST images (cropped to 22 × 24); the ten output neurons identify which digit was presented.

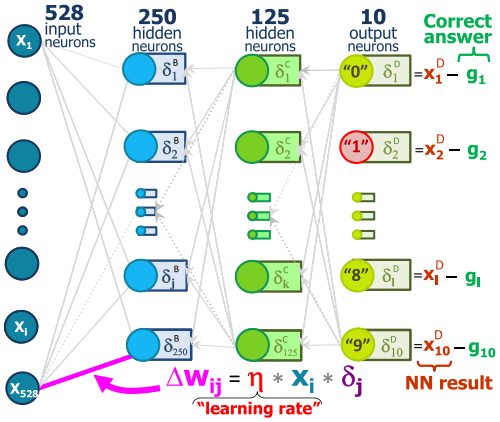


Fig. 3. In supervised learning, error term δ_j is backpropagated, adjusting each weight w_{ij} to minimize an energy function by gradient descent and reducing the classification error between the computed (x_l^D) and the desired output vectors (g_l).

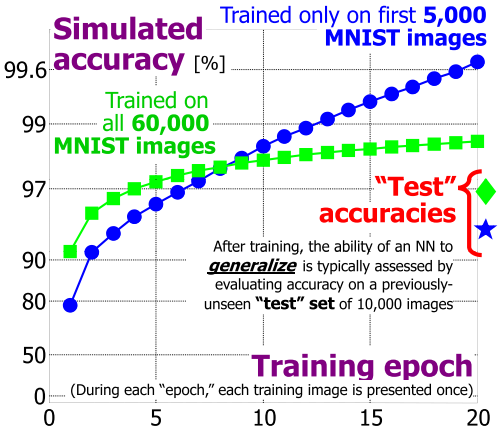


Fig. 4. A three-layer perceptron network can classify previously unseen (test) MNIST handwritten digits with up to ∼97% accuracy [3]. Training on a subset of the images sacrifices some generalization accuracy, but speeds up training.

previously unseen test images (generalization), of ∼97% [3] (Fig. 4); even higher accuracy is possible by first pretraining the weights in each layer [5]. Here, we use tanh() as the nonlinear function $f()$. In addition to those neurons shown in Fig. 2, one bias (always-ON) neuron is added to each layer other than the output layer. Like with STDP, low-power neurons should be achievable by emphasizing brief spikes [6] and local-only clocking. However, note that no CMOS neuron circuitry is built or even specified in this paper—our focus will be solely on the effects of the imperfections in the NVM elements.

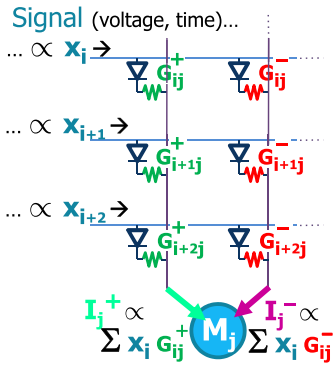


Fig. 5. By comparing total read signal between pairs of bitlines, the summation of synaptic weights (encoded as conductance differences, $w_{ij} = G^+ - G^-$) is highly parallel.

We choose to work with phase-change memory (PCM), since we have access to large PCM arrays in hardware. We discuss the consequences of the fundamental asymmetry in the PCM conductance response: the fact that small conductance increases can be implemented through partial-SET pulses, but the RESET (conductance decrease) operation tends to be quite abrupt. However, we also discuss the use of bidirectional NVM devices (such as nonfilamentary RRAM [7]). We show that such a bidirectional NVM with a symmetric, linear conductance response is fully capable of delivering the same high classification accuracies (on the problem we study, handwritten digit recognition) as a conventional, software-based implementation of the same NN.

II. Considerations for a Crossbar Implementation

By encoding synaptic weight in the conductance difference between a pair of NVM devices, $w_{ij} = G_{ij}^+ - G_{ij}^-$ [2], forward propagation simply compares total read signal on bitlines (Fig. 5). This can be performed by encoding x using some combination of voltage-domain or time-domain encoding (either the number of read pulses or the pulse duration). These CMOS circuitry choices are interesting and important topics, but are beyond the scope of this paper.

Any NVM device that can offer a nondestructive parallel read, as shown in Fig. 5, of memory states that can be smoothly adjusted up or down through a wide range of analog values could potentially be used in this application. Here, we focus on NVM devices that offer a range of analog conductance states.

This paper is concerned with how real NVM devices will respond to programming instructions during *in situ* training of their artificial NN. Unfortunately, the conventional

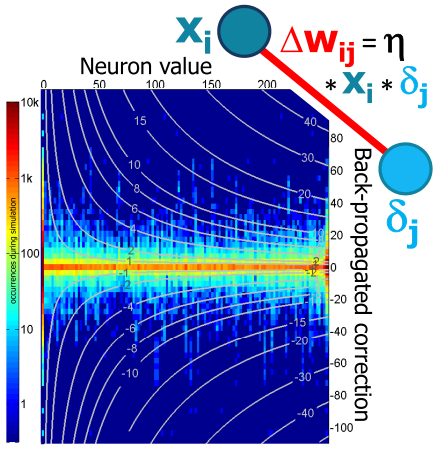


Fig. 6. Backpropagation calls for each weight to be updated by $\Delta w_{ij} = \eta x_i \delta_j$, where η is the learning rate. Colormap: log(occurrences) in the first layer during NN training [Fig. 4 (blue curve)]; white contours identify the quantized increase in the integer weight.

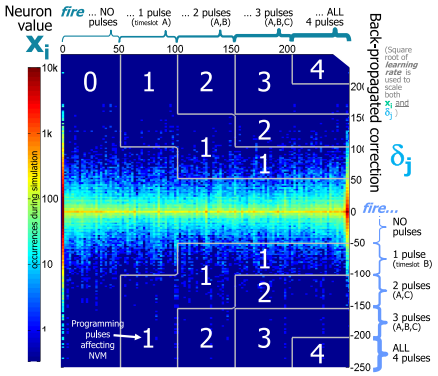


Fig. 7. In a crossbar array, efficient learning requires neurons to update weights in parallel, firing pulses whose overlap at the various NVM devices implements training. Colormap: log(occurrences) in the first layer during NN training [Fig. 8 (red curve)]; white contours identify the quantized number of overlapping programming pulses.

backpropagation algorithm [4] calls for weight updates $\Delta w_{ij} \propto x_i \delta_j$ (Fig. 6), which forces upstream i and downstream j neurons to exchange information uniquely for each and every synapse. This serial, element-by-element information exchange between neurons is highly undesirable in a crossbar array implementation. One alternative is to have each neuron, downstream and upstream, fire pulses based on their local knowledge of δ_j and x_i, respectively. The presence of a nonlinear selector is critical to ensure that NVM programming occurs only when the pulses from both the upstream and the downstream neurons overlap. This allows the neurons to modify weights in parallel, making learning much more efficient [1] (Fig. 7) (note that to reduce peak power, one might choose to stagger these write pulses across the array, one sub-block at a time). Fig. 8 shows, using a simulation of the NN in Figs. 2 and 3, that this adaptation for the NVM implementation has no adverse effect on accuracy.

However, while modifying the update rule is clearly not a problem, the conductance response of any real NVM device exhibits imperfections that can decidedly affect

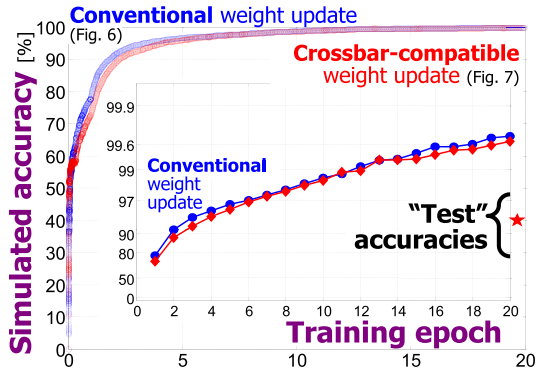


Fig. 8. Computer NN simulations show that a crossbar-compatible weight-update rule (Fig. 7) is just as effective as the conventional update rule (Fig. 6).

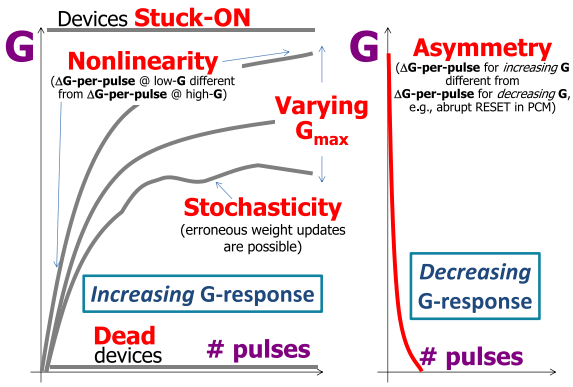


Fig. 9. Conductance response of an NVM device exhibits imperfections, including nonlinearity, stochasticity, varying maxima, asymmetry between increasing/decreasing responses, and nonresponsive devices (at low or high G).

the NN performance. These imperfections include nonlinearity, stochasticity, varying maxima, asymmetry between increasing/decreasing responses, and nonresponsive devices at low or high conductance (Fig. 9). The initial version of this work [8] was the first paper to study the relative importance of each of these factors. This expanded version adds significantly more explanatory details, as well as several new plots detailing paths for future improvement.

Bounding G values would appear to reduce NN training accuracy slightly, as shown by the difference between the blue and the red (top two) curves in Fig. 10. However, unidirectionality and nonlinearity in the G-response strongly degrade accuracy (bottom two curves: green and magenta). Figure insets (Fig. 10) map NVM-pair synapse states on a diamond-shaped plot of G^+ versus G^- (weight is in vertical position). In this context (Fig. 11), a PCM-based synapse with a highly asymmetric G-response (only partial-SET can be done gradually) moves only unidirectionally, from left to right (bipolar filamentary RRAM or Conductive-Bridging RAM (CBRAM) would have an identical problem, except that SET is the abrupt step and it is the RESET step which can be performed gradually).

Once one G value is saturated, subsequent training can only increase the other G value, reducing weight magnitude. Nonlinearity in the G-response further encourages weights of

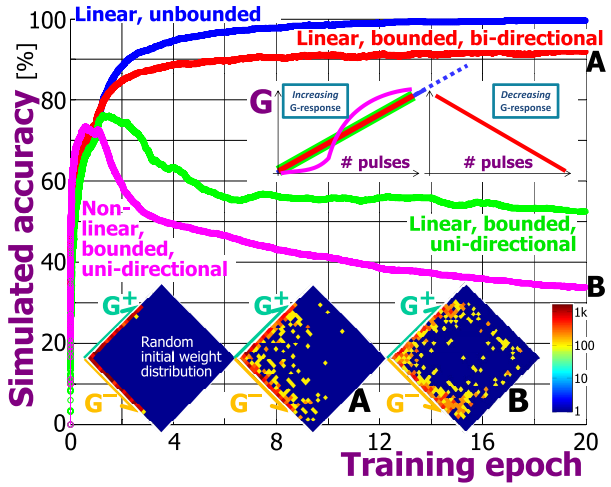


Fig. 10. Bounding G values reduces NN training accuracy slightly, but unidirectionality and nonlinearity in G-response can strongly degrade accuracy. Figure insets map NVM-pair synapse states on a diamond-shaped plot of G^+ versus G^- (weight is vertical position) for a sampled subset of the weights.

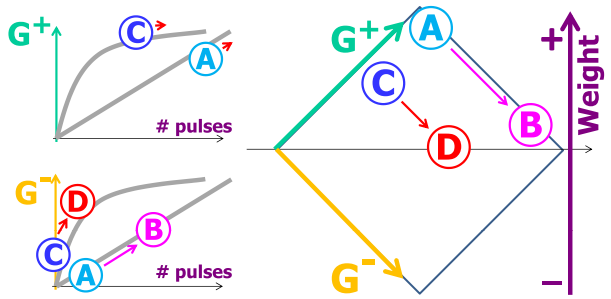


Fig. 11. If G values can only be increased (asymmetric G-response), a synapse at point **A** (G^+ saturated) can only increase G^-, leading to a low weight value (**B**). If the response at small G values differs from that at large G values (nonlinear G-response), alternating weight updates can no longer cancel. As synapses tend to get herded into the same portion of the G-diamond (**C** → **D**), the decrease in average weight can lead to network freeze-out.

low value. If the response at small G values differs from that at large G values, alternating weight updates no longer cancel. As synapses are herded into the same portion of the G-diamond (Fig. 11), the decrease in average weight can lead to network freeze-out [Fig. 10 (inset)]. In such a condition, the network chooses to update very few, if any, weights, meaning that the network stops evolving toward higher accuracy. Worse yet, since the few weight updates that do occur are quite likely to lead to weight magnitude decay, previously trained information is steadily erased and accuracy can actually decrease [Fig. 10 (bottom two curves)].

One solution to the highly asymmetric response of PCM devices is occasional RESET [2], moving synapses back to the left edge of the G-diamond while preserving weight value [using an iterative SET procedure, Fig. 12 (inset)]. However, if this is not done frequently enough, weight stagnation will degrade NN accuracy (Fig. 12) (an analogous approach for bipolar filamentary RRAM or CBRAM would be occasional SET).

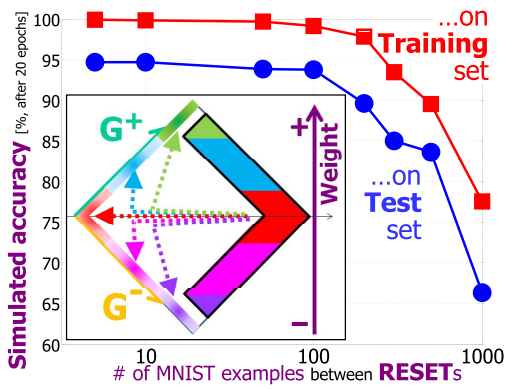


Fig. 12. Synapses with large conductance values (inset, right edge of G-diamond) can be refreshed (moved left) while preserving the weight (to some accuracy), by RESETs to both the G values followed by a partial-SET of one. If such RESETs are too infrequent, weight evolution stagnates and NN accuracy degrades.

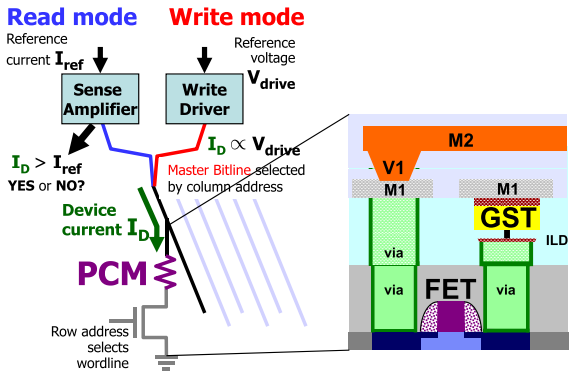


Fig. 13. Mushroom-cell [9], 1T1R PCM devices (180-nm node) with two metal interconnect layers enable 512 × 1024 arrays. A 1-bit sense amplifier measures G values, passing the data to software-based neurons. Conductances are increased by identical 25 ns partial-SET pulses to increase G^+ (G^-) (Fig. 7), or by RESETs to both the G values followed by an iterative SET procedure (Fig. 12).

III. Experimental Results

We implemented a three-layer perceptron of 164 885 synapses (Figs. 2 and 3) on a 500 × 661 array of mushroom-cell [9], 1T1R PCM devices (180-nm node, Fig. 13). While the weight update algorithm (Fig. 7) is fully compatible with a crossbar implementation, our hardware allows only sequential access to each PCM device (Fig. 13). For read, a sense amplifier measures G values, passing the data to software-based neurons. Although this measurement is performed sequentially, weight summation and weight update procedures in the software-based neurons closely mimic the column- and row-based integrations (again, since no particular CMOS circuitry has been specified, we assume that the 8-bit value of x_i is implemented completely accurately; any problems introduced by inaccurate encoding of x_i values by real CMOS hardware could be easily assessed using our tolerancing simulator).

Weights are increased (decreased) by the identical partial-SET pulses (Fig. 7) to increase G^+ (increase G^-) (Fig. 14). The deviation from true crossbar implementation

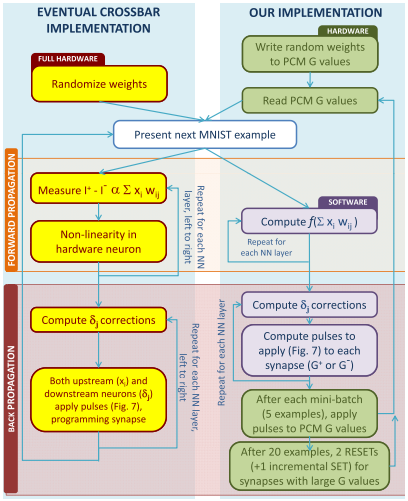


Fig. 14. Although G values are measured sequentially, weight summation and weight update procedures in our software-based neurons closely mimic the column (and row) integrations and pulse-overlap programming needed for parallel operations across a crossbar array. However, since occasional RESET is triggered when both the G^+ and G^- values are large, serial device access is required to obtain and then reprogram individual conductances.

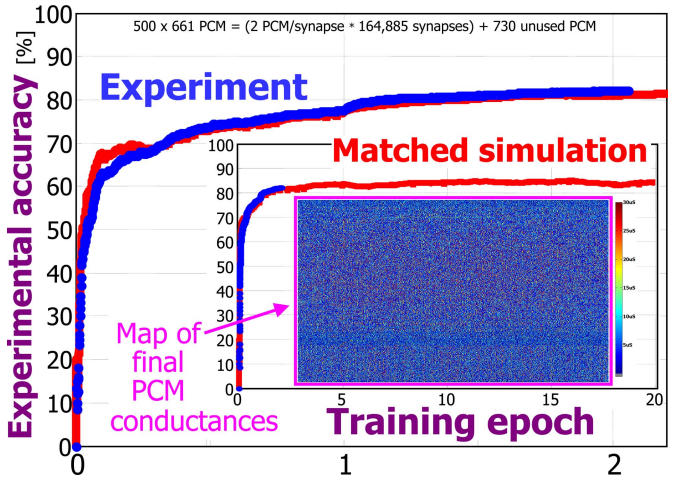


Fig. 15. Training and test accuracy for a three-layer perceptron of 164 885 hardware-synapses, with all weight operations taking place on a 500×661 array of mushroom-cell [9] PCM devices (Fig. 13). Also shown is a matched computer simulation of this NN using parameters extracted from the experiment.

occurs upon occasional RESET (Fig. 12), triggered when either G^+ or G^- are large, thus requiring both knowledge of and control over individual G values. Serial device access is required, both to measure the G values (to determine which are in the L-shaped region at the right side of the G-diamond) and then to fire two RESET pulses (at both G^+ and G^-) followed by an iterative SET procedure to increase one of those two conductances until the correct synaptic weight is restored. Since the time and energy associated with this process are large, it is highly desirable to perform occasional RESET as infrequently and as inaccurately as possible.

Fig. 15 shows measured accuracies for a hardware-synapse NN, with all weight operations taking place on PCM devices. To reduce test time, weight updates for each mini-batch of

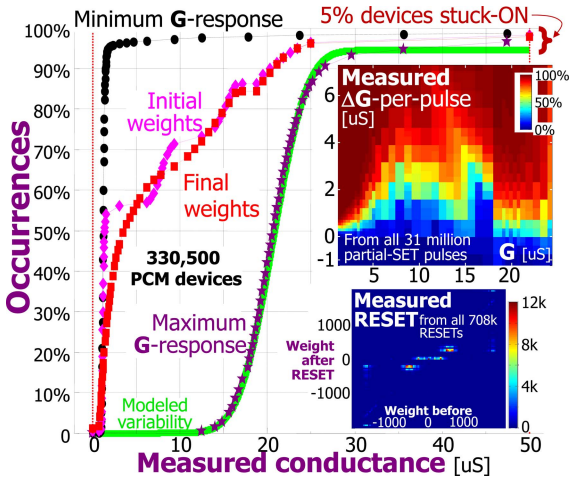


Fig. 16. 50-point cumulative distributions of experimentally measured conductances for the 500×661 PCM array, showing variability and stuck-ON pixel rate. Insets: measured RESET accuracy, and the rate and stochasticity of the G-response, plotted as a colormap of ΔG-per-pulse versus G.

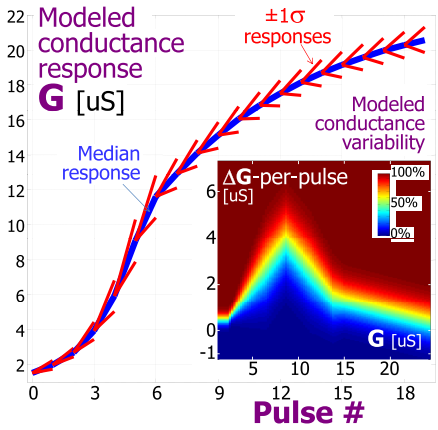


Fig. 17. Fitted G-response versus the number of pulses (blue average, red $\pm 1\sigma$ responses), obtained from our computer model (inset) for the rate and stochasticity of the G-response (ΔG-per-pulse versus G) matched to the experiment (Fig. 16).

five MNIST examples were applied together. Fig. 16 plots the measured G-response, stochasticity, variability, stuck-ON pixel rate, and RESET accuracy. By matching all parameters including stochasticity (Fig. 17) to those measured during the experiment, our NN computer simulation was able to precisely reproduce the measured accuracy trends (Fig. 15).

IV. Tolerancing and Power Considerations

We can now use this matched NN simulation to explore the importance of NVM imperfections. Fig. 18 shows final training (test) accuracy as a function of variations in NVM and NN parameters away from the conditions used in our hardware demo (green dotted line). NN performance is highly robust to stochasticity [Fig. 18(a)], variable maxima [Fig. 18(c)], the presence of nonresponsive devices [Fig. 18(d) and (e)], and infrequent and inaccurate RESETs [Fig. 18(f) and (g)]. A mini-batch of size 1 allows weight updates to be applied immediately [Fig. 18(h)]. However, as mentioned earlier,

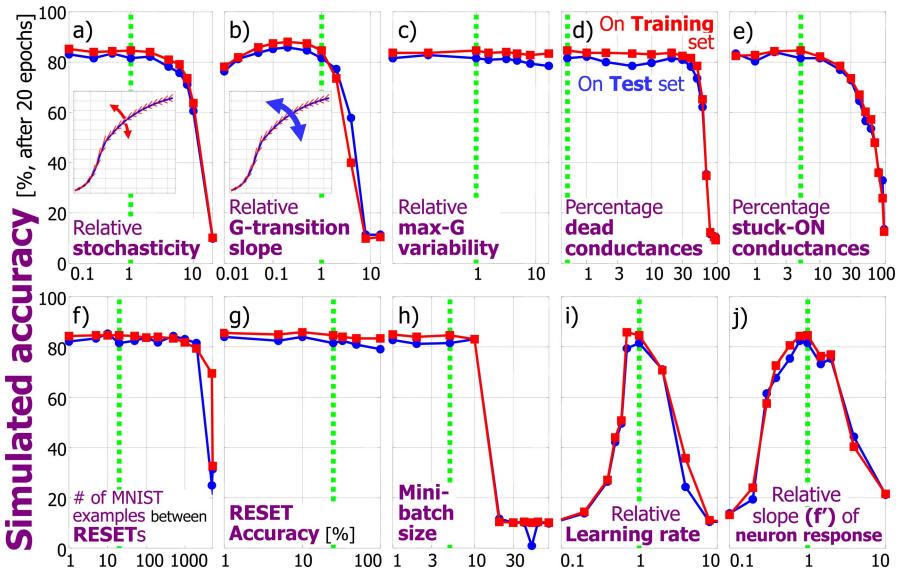


Fig. 18. Matched simulations show that an NVM-based NN is highly robust to (a) stochasticity, (c) variable maxima, (d,e) the presence of nonresponsive devices, and (f) infrequent or (g) inaccurate RESETs. A (h) mini-batch size of 1 avoids the need to accumulate weight updates before applying them. However, the (b) nonlinear and asymmetric G-response limits accuracy to ~85%, and requires (i) learning rate and (j) neuron-response (f') to be precisely tuned.

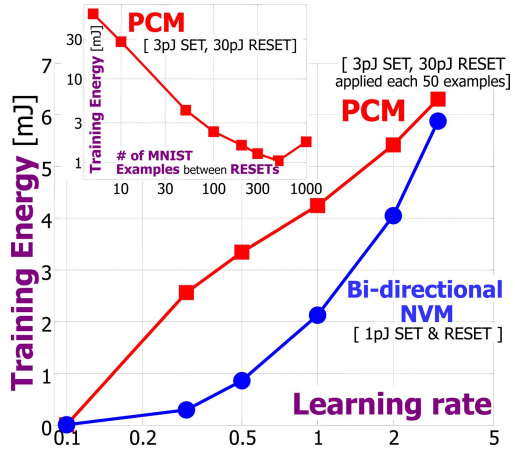


Fig. 19. Despite the higher power involved in RESET rather than partial-SET (30 and 3 pJ for highly scaled PCM [1]), the total energy cost of training can be minimized if the RESETs are sufficiently infrequent (inset). Low-energy training requires low learning rates, which minimize the number of synaptic programming pulses. At higher learning rates, even a bidirectional, linear NVM requiring no RESET and offering low power (1 pJ per pulse) can lead to large training energy.

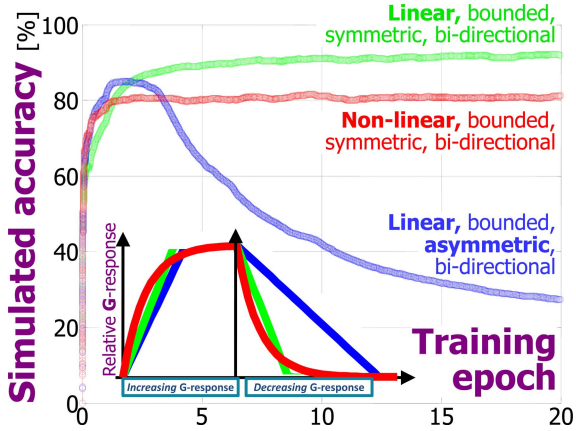


Fig. 20. NN performance is improved if G-response is linear and symmetric (green curve) rather than nonlinear (red). However, asymmetry between the up-going and the down-going G-responses (blue), if not corrected in the weight-update rule (Fig. 7), can strongly degrade performance by favoring particular regions of the G-diamond (Figs. 10 and 11).

nonlinearity and asymmetry in G-response [Fig. 18(b)] limit the maximum possible accuracy (here, to ~85%), and require precise tuning of the learning rate and neuron-response (f') [Fig. 18(i) and (j)]. Too low a learning rate and no weight receives any update; too high, and the imperfections in the NVM response generate chaos. The narrow distribution of these parameters means that the experiment must be tuned very carefully. An extension of an existing NN technique to a crossbar-based NN has been found to provide a much broader distribution of the learning rate. This technique is currently under investigation, and will be the subject of a future publication.

V. Discussion

While the asymmetric G-response of PCM makes it necessary to occasionally stop training, measure all conductances, and apply RESETs and iterative SETs, energy usage can be reasonable if the RESETs are infrequent [Fig. 19 (inset)] and if the learning rate is low (Fig. 19).

An NN with bidirectional NVM-based synapses can deliver high classification accuracy if the G-response is linear and symmetric [Fig. 20 (green curve)] rather than nonlinear (red curve). Asymmetry in the G-response (blue curve) strongly degrades performance. In Fig. 21, we further explore the trends with an ideal but nonlinear NVM, varying both the initial steepness of the G-response and the choice of fully bidirectional weight updates (when increasing weight, for instance, we both increase G^+ and decrease G^- together) or alternating bidirectional (we choose one, but not both, of these two steps). Clearly, a less-steep response is favorable, and the distinction between fully and alternating bidirectional has the most impact for steeply nonlinear G-responses.

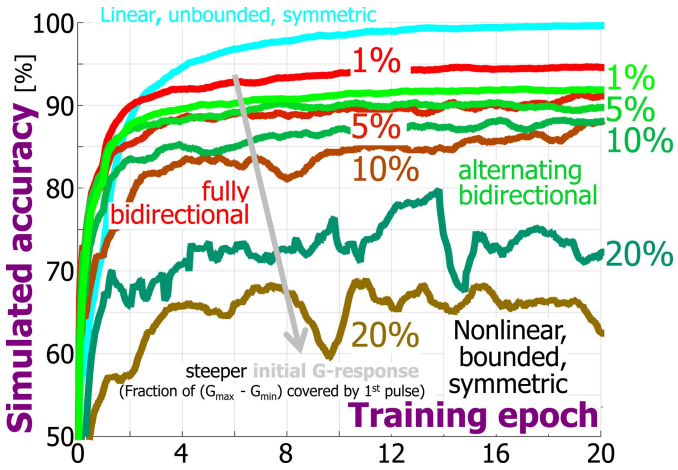


Fig. 21. NN performance with a bidirectional but nonlinear G-response [same basic shape as Fig. 20 (red inset curve)] improves as the response becomes more gentle and the initial slope is less steep. The choice between always updating both conductances when updating the weight (fully bidirectional) and alternating between updating G^+ and G^- but not both (alternating bidirectional), has the most impact when the G-response is steeply nonlinear [note that due to changes in the learning rate and the slope of the nonlinear function $f()$, the red curve in Fig. 20 is not duplicated here].

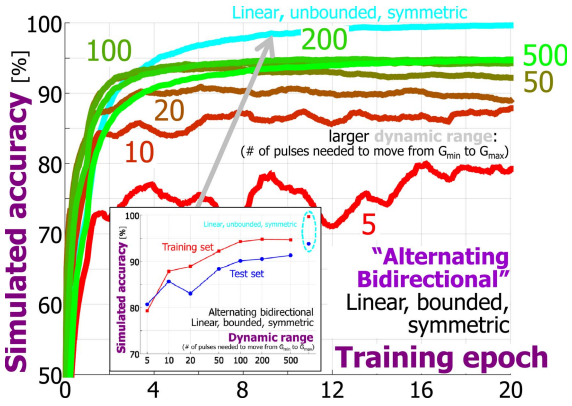


Fig. 22. NN performance when alternating between updating G^+ and G^- but not both (alternating bidirectional), with a linear G-response. This update method cannot reach the performance of a network with unbounded synaptic weights, even when the dynamic range of the linear response is large (e.g., when the change due to any one pulse is only a small fraction of the range between the minimum and the maximum conductances).

The most ideal NVM, with a linear and symmetric conductance response in both directions, would result in more regularly distributed weight values and less freeze-outs, leading to higher accuracies. In Figs. 22 and 23, we show that a gentle linear response (e.g., a large number of identical pulses are needed to change the conductance from minimum to maximum conductance and vice versa), is advantageous compared with a steep response. While both the alternating bidirectional and the fully bidirectional update schemes deliver higher accuracies than an NVM with a nonlinear conductance response, only the fully bidirectional update scheme reaches the same high test accuracies exhibited by networks in which the NVM conductances are unbounded [Fig. 23 (inset)]. Fig. 24 replots the same data from Fig. 23 on a logarithmic vertical scale, to accentuate the high accuracy region.

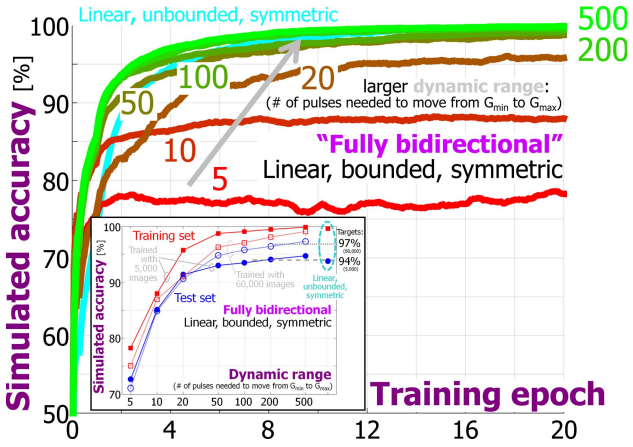


Fig. 23. NN performance (classification accuracy during training) when updating both G^+ and G^- (fully bidirectional scheme), with a linear G-response. The inset shows that, when the dynamic range of the linear response is large, the classification accuracy can now reach that of the original network (a test accuracy of 94% when trained with 5000 images; 97% when trained with all 60 000 images).

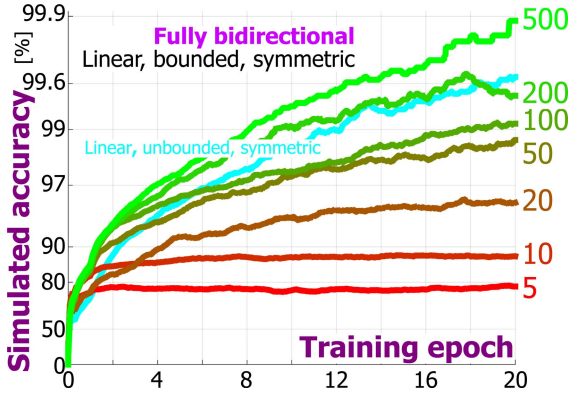


Fig. 24. Same NN performance data for a linear, symmetric NVM under the fully bidirectional scheme (Fig. 23), here replotted on a logarithmic scale that accentuates the high accuracy region. Here, only the classification accuracy on the training set is shown, which can reach close to 100%, at which point the accuracy on the test set of 10 000 images that the network has never seen before (Fig. 23) becomes the only relevant way to gauge the network performance.

The reason for this difference is shown in Fig. 25, where one example of a desired sequence of weight updates is contrasted to the actual weight updates that get implemented in these two update schemes. In Fig. 25(b), we show that when the state of the synapse is at the boundaries of the G-diamond, there is a significant chance that the next weight update using the alternating bidirectional scheme will have little or no impact, simply because a conductance that is already saturated cannot be increased (decreased) any further. In the fully bidirectional update scheme, some amount of weight updates will still occur at the edges of the G-diamond, leading to smaller discrepancies between the desired and the actual weight changes, and thus higher performance. In addition, because the weights only move up and down the G-diamond in the fully bidirectional scheme, the synapses stay in the center stripe of the G-diamond [Fig. 26(b)], where they have access to the full dynamic range available. In contrast, because each

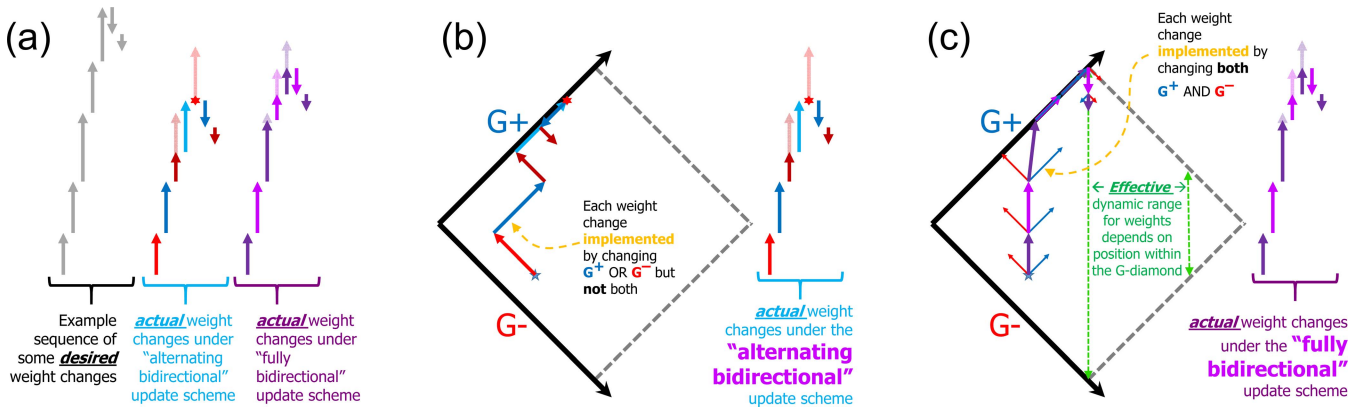


Fig. 25. Because of the finite bounds on conductance values, any desired sequence of weight changes [left-hand portion of (a)] will not be fully implemented in an NVM-based neuromorphic system. Actual weight updates that occur in (b) an alternating bidirectional update scheme, in which we alternate between updating G^+ and G^- but not both, and (c) a fully bidirectional update scheme, in which we always update both G^+ and G^-. With the alternating bidirectional scheme, synapses whose conductance values are located at/near the boundaries of the G-diamond can potentially lead to a situation where a large weight update is completely ignored. In contrast, in the bidirectional scheme, such large weight updates are simply reduced in the magnitude of their effect, and synapses tend to remain in the center of the G-diamond.

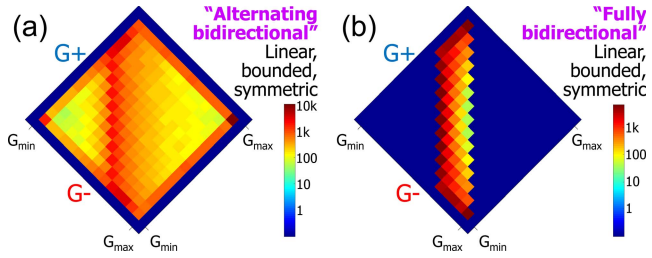


Fig. 26. When the G-response is steeply nonlinear, a fully bidirectional scheme exhibits lower accuracy (Fig. 21), because any single weight update could potentially make two overly large conductance changes instead of just one. However, the fully bidirectional scheme provides better performance for a linear response with high dynamic range (compare Figs. 22 and 23), because (b) the small symmetric changes of each conductance move the synaptic weight up and down along the central vertical axis of the G-diamond. In contrast, (a) the alternating bidirectional scheme can move some synapses to the left or right edges of the G-diamond, where the effective dynamic range (maximum weight magnitude) is significantly reduced.

weight update in the alternating bidirectional scheme moves along a diagonal line, some number of synapses end up at the edges of the G diamond, where the effective dynamic range that they can access is significantly reduced [Fig. 26(a)].

These results demonstrate conclusively that NVM devices should be fully capable of delivering the same classification accuracy on the MNIST handwritten digits as a conventional implementation of this artificial NN. All that is required of the NVM device is that it offers a bidirectional, linear, and symmetric response in conductance with large dynamic range (e.g., the change due to any one pulse represents only a small fraction of the entire conductance range available).

Other future work will be needed to demonstrate a full crossbar-array implementation, including dedicated CMOS circuitry for the summation of synaptic weights during both forward propagation and backpropagation through nearly identical high-performance nonlinear selector devices. The values of neurons (x) and backpropagated errors (δ) will need to be stored in CMOS circuitry and presented to the crossbar, through some combination of analog voltage levels, the number of read pulses, and/or the duration of read pulses. The need to synchronize write pulse timing between upstream and downstream neurons, and techniques to disperse the high-energy writes in time (to reduce the load on write drivers and voltage supplies) must also be addressed in future work.

VI. Conclusion

Using two PCM devices per synapse, a three-layer perceptron with 164 885 synapses was trained with backpropagation on a subset (5000 examples) of the MNIST database of handwritten digits to the high accuracy of (82.2%, 82.9% on test set). A weight-update rule compatible with NVM + selector crossbar arrays was developed, and was shown to have no adverse effect on accuracy. A novel G-diamond concept (Fig. 11) was introduced to illustrate issues created by nonlinearity and asymmetry in NVM conductance response. Asymmetry can be mitigated by an occasional RESET strategy (Fig. 12), which can be both infrequent and inaccurate [Figs. 12 and 18(f) and (g)].

Using an NN simulator matched to the experimental demonstrator, extensive tolerancing was performed (Fig. 18). Results show that network parameters, such as the learning rate and the slope of the nonlinear neuron-response function [Fig. 18(i) and (j)], and the nonlinearity, symmetry, and bounded nature of the conductance response (Figs. 9–11 and 20–26), are critical to achieving high performance in an NVM-based NN.

Our results show that all NVM-based NNs (not just those based on PCM) can be expected to be highly resilient to random effects (NVM variability, yield, and stochasticity), but will be highly sensitive to gradient effects that act to steer all synaptic weights. A learning rate just high enough to avoid network freeze-out is shown to be advantageous for both high accuracy and low training energy. We also prove that a bidirectional NVM with a symmetric, linear conductance response of high dynamic range (each conductance step is relatively small) would be fully capable of delivering the same high classification accuracies on the MNIST handwritten digit

database as a conventional, software-based implementation, ranging from >94% when trained on 5000 examples to >97% when trained on the full set of 60 000 training examples.

REFERENCES

[1] B. L. Jackson *et al.*, "Nanoscale electronic synapses using phase change devices," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 9, no. 2, 2013, Art. ID 12.

[2] M. Suri *et al.*, "Phase change memory as synapse for ultra-dense neuromorphic systems: Application to complex visual pattern extraction," in *IEDM Tech. Dig.*, Dec. 2011, pp. 4.4.1–4.4.4.

[3] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[4] D. E. Rumelhart, G. E. Hinton, and J. L. McClelland, "A general framework for parallel distributed processing," in *Parallel Distributed Processing*. Cambridge, MA, USA: MIT Press, 1986.

[5] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[6] B. Rajendran *et al.*, "Specifications of nanoscale devices and circuits for neuromorphic computational systems," *IEEE Trans. Electron Devices*, vol. 60, no. 1, pp. 246–253, Jan. 2013.

[7] J.-W. Jang, S. Park, G. W. Burr, H. Hwang, and Y.-H. Jeong, "Optimization of conductance change in $Pr_{1-x}Ca_xMnO_3$-based synaptic devices for neuromorphic systems," *IEEE Electron Device Lett.*, vol. 36, no. 5, pp. 457–459, May 2015.

[8] G. W. Burr *et al.*, "Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element," in *Proc. IEEE IEDM*, Dec. 2014, pp. 29.5.1–29.5.4.

[9] M. Breitwisch *et al.*, "Novel lithography-independent pore phase change memory," in *Proc. Symp. VLSI Technol.*, Jun. 2007, pp. 100–101.

Geoffrey W. Burr (S'87–M'96–SM'13) received the Ph.D. degree in electrical engineering from the California Institute of Technology, Pasadena, CA, USA, in 1996.

He joined IBM Research–Almaden, San Jose, CA, USA, in 1996, where he is currently a Principal Research Staff Member. His current research interests include nonvolatile memory and cognitive computing.

Robert M. Shelby received the Ph.D. degree in chemistry from the University of California at Berkeley, Berkeley, CA, USA.

He joined IBM, Armonk, NY, USA, in 1978. He is currently a Research Staff Member with IBM Research–Almaden, San Jose, CA, USA.

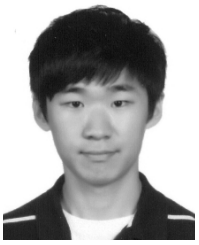
Dr. Shelby is a Fellow of the Optical Society of America.

Severin Sidler received the B.S. degree from the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, where he is currently pursuing the M.S. degree in electrical engineering.

He is currently an Intern with IBM Research–Almaden, San Jose, CA, USA. His current research interests include cognitive computing and systems engineering.

Carmelo di Nolfo received the B.S. and M.S. degrees in electrical and computer engineering from the University of Liège, Liège, Belgium, in 2012 and 2014, respectively.

He studied at the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, during the M.S. degree work, and spent six months as an Intern at IBM Research–Almaden, San Jose, CA, USA. He joined the SyNAPSE team, IBM Research–Almaden, in 2015.



Junwoo Jang received the B.S. degree in electrical engineering from the Pohang University of Science and Technology, Pohang, Korea, in 2012, where he is currently pursuing the Ph.D. degree with the Department of Creative IT Engineering.

He spent four months visiting IBM Research–Almaden, San Jose, CA, USA, in 2013. His current research interests include circuit design for cognitive computing systems.

Irem Boybat (S'15) received the B.S. degree in electronics engineering from Sabanci University, Istanbul, Turkey. She is currently pursuing the M.S. degree in electrical engineering with the École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland.

She was an Intern with IBM Research–Almaden, San Jose, CA, USA, from 2014 to 2015. Her current research interests include cognitive computing, semicustom design, and embedded systems.

Rohit S. Shenoy (M'04) received the B.Tech. degree in engineering physics from IIT Bombay, Mumbai, India, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA.

He was a Research Staff Member with IBM Research–Almaden, San Jose, CA, USA, from 2005 to 2014. He joined Intel, Santa Clara, CA, USA, in 2014, as a Device Engineer, where he is currently involved in NAND flash development.

Pritish Narayanan (M'14) received the Ph.D. degree in electrical and computer engineering from the University of Massachusetts–Amherst, Amherst, MA, USA, in 2013.

He joined IBM Research–Almaden, San Jose, CA, USA, as a Research Staff Member. His current research interests include emerging technologies for logic, nonvolatile memory, and cognitive computing.

Kumar Virwani (S'05–M'10) received the Ph.D. degree from the University of Arkansas at Fayetteville, Fayetteville, AR, USA, in 2007.

He has been with IBM Research–Almaden, San Jose, CA, USA, since 2008. He is currently involved in projects on storage class memory, lithium-air batteries, low-k dielectrics, and photovoltaics.

Emanuele U. Giacometti received the B.S. degree in electronics engineering and the master's degree in nanotechnologies from the Politecnico di Torino, Turin, Italy, in 2012 and 2014, respectively. He will begin pursuing the Ph.D. degree in 2015.

He spent six months with IBM Research–Almaden, San Jose, CA, USA, as an Intern.

Bülent N. Kurdi received the Ph.D. degree from the Institute of Optics, University of Rochester, Rochester, NY, USA, in 1989.

He spent eleven years with IBM Research–Almaden, San Jose, CA, USA. He joined Wavesplitter Technologies, Inc., Fremont, CA, USA, in 2000. He joined IBM Research–Almaden, in 2003, where he is currently Senior Manager of the Novel Device Prototyping and Characterization Department.

Hyunsang Hwang (M'88)received the Ph.D. degree in materials science from The University of Texas at Austin, Austin, TX, USA in 1992.

He was with LG Semiconductor Corporation, Morristown, NJ, USA. He became a Professor of Materials Science and Engineering with the Gwangju Institute of Science and Technology, Gwangju, Korea, in 1997. He joined the Department of Materials Science and Engineering, Pohang University of Science and Technology, Pohang, Korea, in 2012.