

Modeling-based design of brain-inspired spiking neural networks with RRAM learning synapses

G. Pedretti^{1*}, S. Bianchi¹⁺, V. Milo¹, A. Calderoni², N. Ramaswamy², and D. Ielmini¹

¹Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano and IU.NET, piazza L. da Vinci 32, 20133, Milano, Italy, ⁺Authors contributed equally *email: giacomo.pedretti@polimi.it ²Micron Technology, Boise, ID, USA

Abstract—Brain-inspired computing is currently gaining momentum as a viable technology for artificial intelligence enabling recognition, language processing and online unsupervised learning. Brain-inspired circuit design is currently hindered by 2 fundamental limits: (i) understanding the event-driven spike processing in the human brain, and (ii) developing predictive models to design and optimize cognitive circuits. Here we present a comprehensive model for spiking neural networks based on spike-timing dependent plasticity (STDP) in resistive switching memory (RRAM) synapses. Both a Monte Carlo (MC) model and an analytical model are presented to describe experimental data from a state-of-the-art neuromorphic hardware. The model can predict the learning efficiency and time as a function of the input noise and pattern size, thus paving the way for model-based design of cognitive brain-like circuits.

I. INTRODUCTION

Neuromorphic computing with RRAM or phase change memory (PCM) has been recently shown to enable recognition of handwritten characters [1] and faces [2], thus providing the basis for artificial vision and drone/robot/car navigation. All these achievements generally rely on deep learning architectures with supervised backpropagation, which, however, lack analogy with the human brain and is not suitable for online unsupervised learning. On the other hand, spiking neural networks with spike-timing dependent plasticity (STDP) are capable of replicating bio-realistic online/unsupervised learning [3,4]. A RRAM-based neuromorphic circuit was shown to perform learning and recognition of static and dynamic patterns by implementation of a STDP learning rule in hardware [5]. Learning of multiple patterns was demonstrated in experiments [5] and simulations [6] by STDP weight update. However, the capability to design and optimize brain-inspired circuits has been hindered by the lack of accurate, predictive models.

We present new models for STDP-based brain-inspired learning circuits. After describing the reference hardware for STDP learning with RRAM synapses, we present both Monte Carlo (MC) and analytical models that can predictively describe the time evolution of weights and the impact of circuit parameters (integration threshold) and input parameters (noise) on pattern learning time and efficiency. Finally, we present the full capability of the models to optimize the problem of MNIST dataset learning and recognition.

II. NEUROMORPHIC HARDWARE

We studied feedforward neural networks with unsupervised learning via STDP in the one-transistor/one-resistor (1T1R) synapse in Fig. 1a, consisting of a transistor-selected RRAM device [7]. The synapse gate is connected to the pre-synaptic neuron (PRE), while the top electrode (TE) and bottom electrode (BE) are connected to the post-synaptic neuron (POST). The PRE spike induces a current spike across the 1T1R synapse reaching the POST through the BE. Integration of current spikes in the POST raises the internal potential, eventually causing fire, *i.e.*, a POST spike. At fire, a feedback spike is applied to the TE, where it can lead to RRAM set/reset if

overlapped with the PRE spike. Positive spike delay, where the POST spike occurs in response to the PRE spike, causes set process, or potentiation. Negative spike delay, where the POST spike occurs in response to a previous PRE spike, causes reset process, or depression. Adopting a feedforward fully-connected perceptron structure (Fig. 1b) allows visual pattern recognition by stochastic learning, where pattern and noise are alternatively submitted by the 1st layer (PRE) to the 2nd layer (POST) via STDP synapses.

Fig. 1c shows a neuromorphic hardware implemented with 1T1R synapses and an Arduino microcontroller (μ C). This neural network was adopted to demonstrate, for the first time, full-hardware unsupervised learning with RRAM synapses [5]. HfO_x-based RRAM devices with set/reset current $<50 \mu$ A (Fig. 2a) and average resistance window around 10x (Fig. 2b) were used as synaptic element. In this neuromorphic circuit, the synaptic weights can be monitored online as shown in Fig. 3: as pattern and noise input spikes are submitted to the neural network, synapses in the pattern (a diagonal ‘/’ in the figure) are potentiated while synapses in the background (out of the pattern) are depressed. Monitoring the synaptic weights allows to directly assess the learning efficiency, energy and time. This state-of-the-art brain-inspired hardware with 4x4 synaptic network was used to collect experimental data of unsupervised learning by STDP.

III. CIRCUIT MODELS FOR STDP NETWORKS

To design, simulate and optimize RRAM networks for unsupervised learning, circuit models at various levels of abstraction are needed, spanning from detailed spice-like models to high-level analytical compact models for the average weight in the synaptic network.

A. MC model

A MC model was developed to describe set/reset processes occurring at each epoch (*i.e.*, event of pattern/noise presentation) in the training process. The RRAM response to overlapping PRE/POST spikes was simulated by an analytical model for filamentary switching (see simulations in Fig. 2a) [8]. Set/reset stochastic variability was simulated by assuming log-normal distributions of HRS and LRS resistance R at increasing voltage, as shown in Fig. 2b. Fig. 3b shows the calculated weights during the stochastic submission of pattern input spikes, highlighting accurate prediction of potentiation and depression with realistic variations of synaptic weights $1/R$.

B. Analytical Compact model

Predicting the average learning behavior of a network with the MC model requires time-consuming simulations to average variability effects. To speed up the evaluation of STDP learning, we have developed a high-level analytical compact model adopting rate equations for synaptic weight in the pattern [G_P in Eq. (1), Fig. 4a] and in the background [G_B in Eq. (2), Fig. 4a]. The rate equations include all possible sequences of input spikes (*e.g.*, pattern/noise, noise/pattern, etc.) as driving forces for potentiation/depression

(Tab. I). For instance, pattern/noise sequence causes pattern potentiation in Eq. (1) and background depression in Eq. (2). Conversely, noise/pattern sequence causes pattern depression and background potentiation, while noise/noise induces weight randomization and pattern/pattern is prevented by the refractory time, where any PRE remains inactive for one epoch after spiking. The analytical compact model can predict the time evolution of G_P and G_B for any set of input variables in Fig. 4b, according to the equation parameters in Fig. 4c.

IV. SIMULATION RESULTS

We validated the predictive models by extensive variation of the parameters of stochastic learning. Fig. 5 illustrates the stochastic learning approach in STDP networks, where pattern and noise epochs are randomly alternated (Fig. 5a). In the pattern epoch, the pattern is submitted with density P given by the number of pattern pixels divided by the overall image size. In the noise epoch, noise spikes are randomly activated with a probability N , e.g., 1/10 of the PREs are activated at each noise epoch for $N = 10\%$ (Fig. 5b). The probability of pattern presentation R_P and noise presentation R_N were both 50% except where noted. A threshold of integrated current $Q = 0.5 \mu C$ per pattern synapse was assumed. During the learning process, typically lasting 1000 epochs (10 s), synaptic weights were monitored and learning efficiency was studied for various values of P , N , R_P and R_N .

C. Impact of noise density N

As noise accelerates background depression which is the slowest process in the learning dynamics (e.g., see Fig. 3), we studied the impact of N on learning efficiency and speed. Fig. 6 shows the submitted input (top), the measured synaptic weights (center) and the MC calculations (bottom), for increasing noise density $N = 5\%$ (a), 10% (b) and 15% (c). As N increases, the depression becomes faster as expected, however learning also becomes less stable, as shown by the weight fluctuations with time. Unstable learning at large N is due to noise spikes inducing fire: if fire is then followed by pattern presentation, all pattern synapses undergo depression, and background potentiation takes place, as seen in Fig. 6 for large N . MC simulations account for the improved learning speed and the instability at increasing N . Fig. 7a summarizes the learning efficiency by showing the probability of true fire P_{learn} (fire in response to the presentation of the pattern), and the probability of false fire P_{err} (fire in response to a wrong pattern), showing that P_{learn} decreases and P_{err} increases at increasing N , as a result of the unstable learning. MC calculations agree well with data, thus demonstrating the predictability of STDP learning. Fig. 7b shows the measured and calculated weights $1/R$ after 1000 epochs as a function of N , again showing window closure due to unstable learning at large N . Fig. 7c shows the learning time t_{learn} , namely the time for the average weight of background synapses dropping below $1.5 \times 10^{-5} \Omega^{-1}$ (see inset). The best trade-off between learning speed (Fig. 7c) and stability (Fig. 7a and b) is for $N = 3\%$.

D. Impact of pattern density P

Pattern presentation leads to fire, thus potentiation of pattern synapses is most efficient if the pattern density P is much larger than N . To study the impact of pattern density P on learning, Fig. 8a shows the measured and calculated P_{learn} and P_{err} as a function of P , for $N = 3\%$. P_{learn} increases and P_{err} decreases for $P > 20\%$, thus much larger than N . For P comparable to N , noise and pattern compete in potentiation, thus inhibiting learning. Fig. 8b shows the measured and calculated weights as a function of P , indicating window closure for decreasing P . The learning time in Fig. 8c shows similar improvement of speed for increasing P , thanks to accelerated pattern potentiation.

E. Impact of pattern/noise probabilities

Varying pattern probability R_P and noise probability R_N , with $R_N + R_P = 100\%$, impacts learning efficiency and speed as shown in Fig. 9. Increasing R_P leads to pattern potentiation and consequently to an increase of P_{learn} and a decrease of P_{err} . For the same reason, the window between pattern and background weights increases in Fig. 9b. These results suggest that an optimum probability is $R_P = R_N = 50\%$, which can be achieved by systematic (rather than random as in Fig. 5) alternation between the pattern and noise input channels.

V. OPTIMIZED DIGIT RECOGNITION

Fig. 10 shows a contour plot of t_{learn} as a function of P and N , according to the analytical model. In general, the learning time is minimized for $N \ll P$, where the pattern is potentiated and the background is depressed efficiently. In the figure, the points of maximum window are highlighted, corresponding to the optimized value of N for any given value of P . This allows to optimize the stochastic learning process concerning learning time and efficiency.

To test the impact of optimized parameters on learning, we simulated the unsupervised training and classification of handwritten digits from the MNIST database. Fig. 11 shows the adopted circuit, consisting of a 3-layer perceptron with 28×28 PREs in the 1st layer, 50,000 POSTs in the 2nd layer, and 10 classification neurons in the 3rd layer. [9]. Synapses between the 1st and 2nd layers are trained by unsupervised (STDP) process and synapses between the 2nd and 3rd layers are trained by supervised process for classification. Unsupervised learning was carried out with either fixed N ($N = 5\%$ and $N = 10\%$) or optimized N (Fig. 10). Fig. 12 shows the learning efficiency, namely the matching of synaptic weights with the pattern after training, and the classification efficiency, namely the probability of true fire and false fire in the classification layer. Overall, the optimized N results in the best efficiency of about 92% (learning) and 85% (classification). Fig. 13 summarizes the true fire/false fire probabilities for optimized N , supporting the robust pattern recognition by modeling-based design.

VI. CONCLUSIONS

We present a new methodology for the design and optimization of a spiking neural network for brain-inspired unsupervised learning with STDP in RRAM synapses. We show that MC models of RRAM circuits and analytical compact models of the STDP dynamics accurately predict the learning behavior in a state-of-the-art spiking network with RRAM synapses. We finally show an improvement up to 92% of learning efficiency by using optimized noise during unsupervised learning of handwritten digits from the MNIST database, thus paving the way for the predictive design and control of STDP learning circuits.

VII. ACKNOWLEDGMENTS

This work was supported in part by the European Research Council (grant ERC-2014-CoG-648635-RESCUE).

REFERENCES

- [1] G. W. Burr, *et al.*, IEEE Trans. Electron Devices 62, 3498 (2015).
- [2] P. Yao, *et al.*, Nat. Commun. 8, 15199 (2017).
- [3] P. A. Merolla, *et al.*, Science 345, 668–673 (2014).
- [4] N. Qiao, *et al.*, Front. Neurosci. 9:141. (2015).
- [5] G. Pedretti, *et al.*, Sci. Rep. 7:5288 (2017).
- [6] P. U. Diehl and M. Cook, Front. Comput. Neurosci. 9, 99 (2015).
- [7] S. Ambrogio, *et al.*, IEEE Trans. Electron Devices 63, 1508 (2016).
- [8] S. Ambrogio, *et al.*, IEEE Trans. Electron Devices 61, 2378 (2014).
- [9] S. Ambrogio, *et al.*, Symp. VLSI Tech. Dig., 196 (2016).

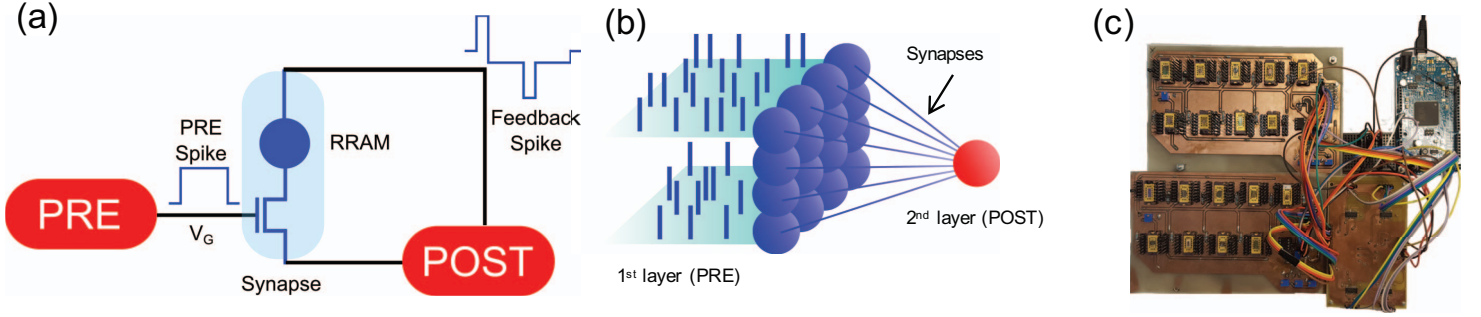


Fig. 1 Schematic illustration of the 1T1R synapse with RRAM device for STDP (a), 2-layer perceptron structure for pattern learning and recognition (b), and picture of the neuromorphic hardware implemented on a PCB (c).

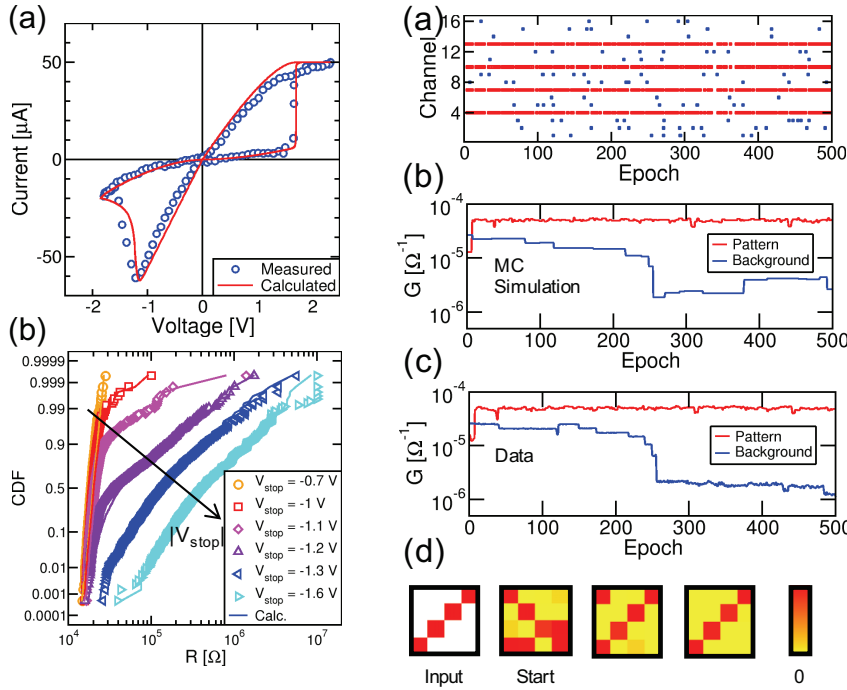


Fig. 2 Measured and calculated I-V curves of the RRAM device used in this work (a) and distributions of measured and calculated R at increasing stop voltage (b).

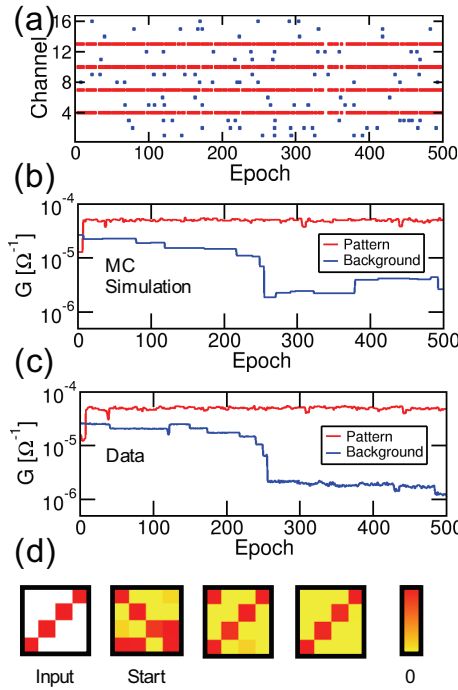


Fig. 3 Input data for stochastic on-line learning (a), evolution of average pattern/background weights from MC calculations (b) and experiments (c), and sketch of the pattern and of synaptic weights at increasing number of epochs (d).

SEQUENCE	POTENTIATION	DEPRESSION
Pattern / Pattern	Not possible	
Pattern / Noise	Pattern	Background
Noise / Pattern	Background	Pattern
Noise / Noise	Random	Random

Tab. 1 Summary of possible input sequences (pattern/pattern, pattern/noise, etc.) and their corresponding weight update by STDP, i.e., potentiation/depression of the pattern or background synapses.

$$(a) \frac{dG_p}{dt} = ANR_N(G_{max} + G_{min} - 2G_p) + C(G_{max} - G_p)(G_p - \alpha NG_{min})(P - N) * R_p \quad (1)$$

$$\frac{dG_B}{dt} = A'NR_N(G_{max} + G_{min} - 2G_B) + D(\beta G_{max} - G_B)(G_B - G_{min})(N - P) * R_N R_p * N(1 - P) \quad (2)$$

Variable	Definition	Parameter	Value
P	Pattern Density	A	10 s^{-1}
N	Noise Density	A'	0.5 s^{-1}
R_N	Noise Probability	C	$R_p * 30 * 10^5 \text{ } \Omega \text{ s}^{-1}$
R_p	Pattern Probability	D	$R_p * 35 * 10^6 \text{ } \Omega \text{ s}^{-1}$
		α	60
		β	0.69

Fig. 4 Analytical rate equations Eq. (1) and (2) for the pattern weight G_p and the background weight G_B respectively (a), definition of the stochastic training variables (b) and definition of the parameters entering the rate equations (c).

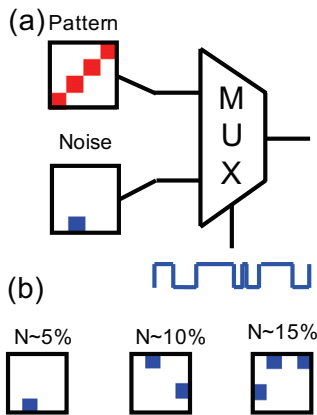


Fig. 5 Illustration of the random alternation between pattern and noise in stochastic training (a) and typical noise input at increasing N (b).

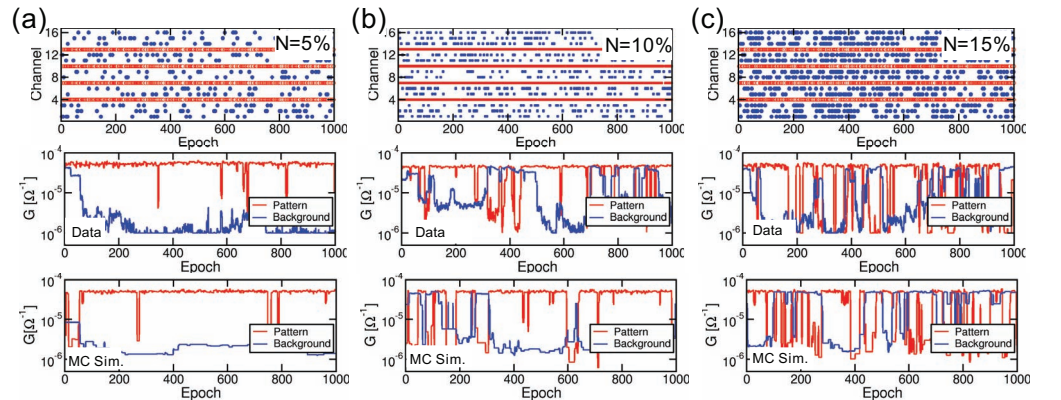


Fig. 6 Training input data (top), measured average weights (center) and calculated average weights (bottom) at increasing noise density, namely N = 5% (a), 10% (b) and 15% (c). As N increases, the learning time decreases, while the pattern and background weights show increasing fluctuations due to input noise inducing fire in Tab. I.

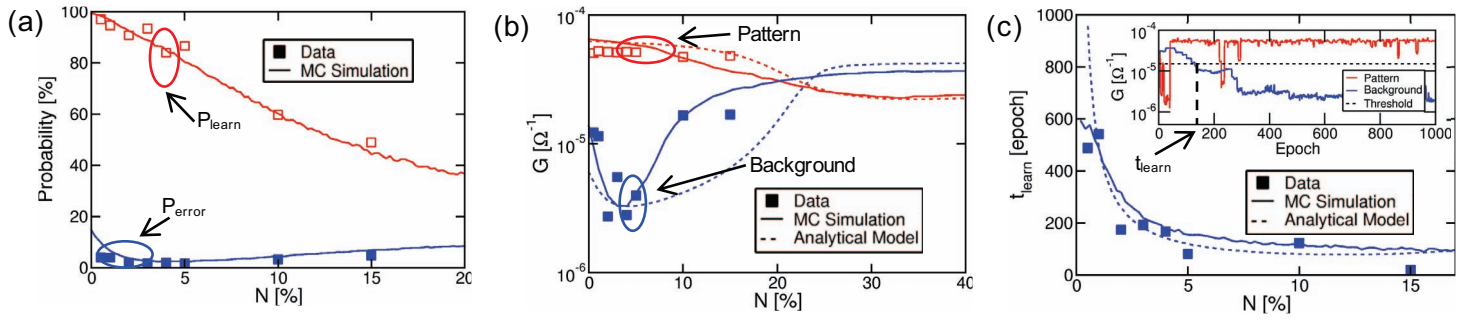


Fig. 7 Measured and calculated learning probability P_{learn} and error probability P_{err} (a), average weight of the pattern and background (b), and learning time (c) as a function of N . Experimental data from the neuromorphic hardware, MC simulations, and simulations from the analytical model are compared. The best trade-off between learning speed and accuracy is around $N = 3\%$.

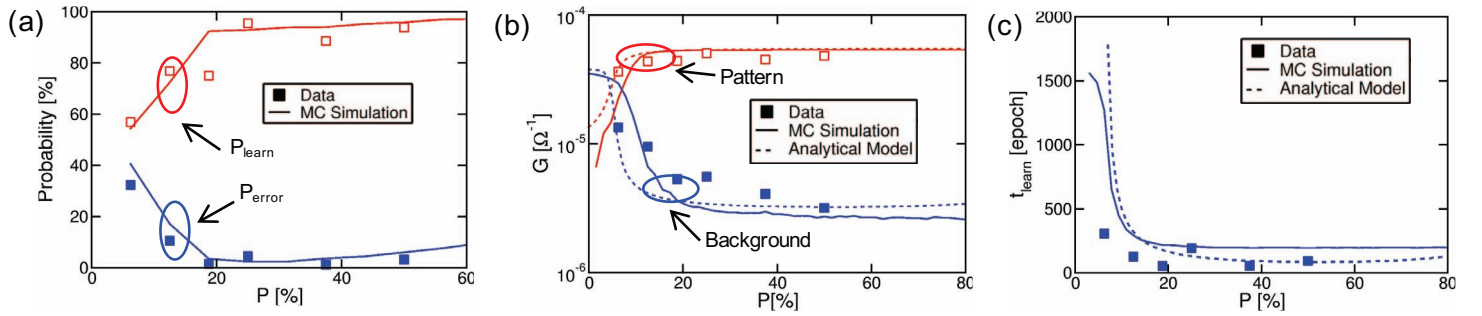


Fig. 8 Measured and calculated learning probability P_{learn} and error probability P_{err} (a), average weight of the pattern and background (b), and learning time (c) as a function of P , with $N = 3\%$. Stable and fast learning takes place for $P \gg N$.

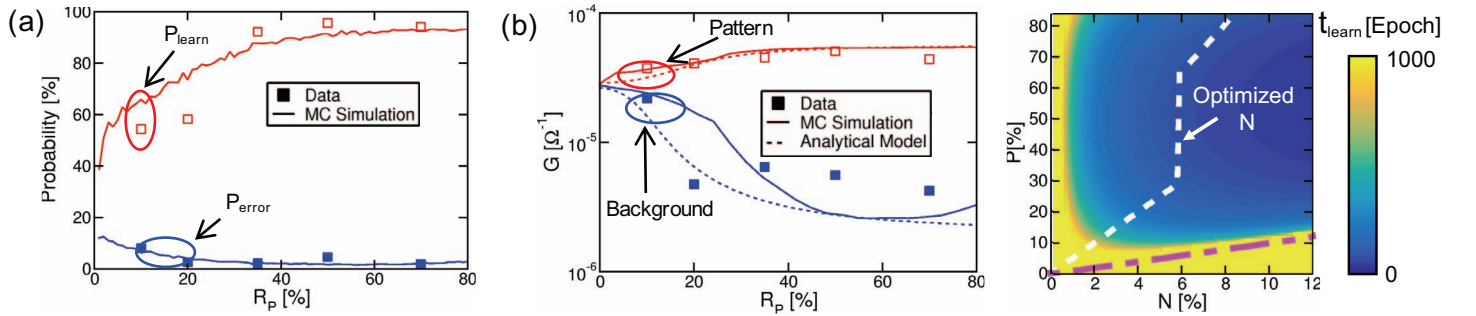


Fig. 9 Measured and calculated learning probability P_{learn} and error probability P_{err} (a) and average weight of the pattern and background (b) as a function of R_P , with $R_N = 1 - R_P$. Stable learning takes place for $R_P > 50\%$.

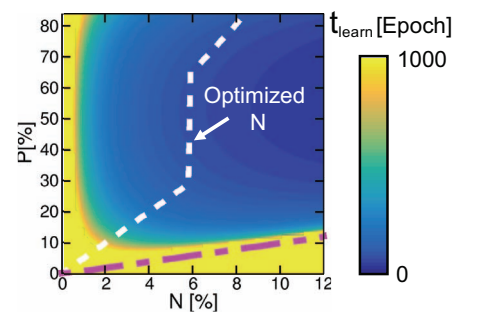


Fig. 10 Calculated t_{learn} from the analytical model as a function of P and N in a color map. The white line indicates optimized N at variable P to maximize the learning speed.

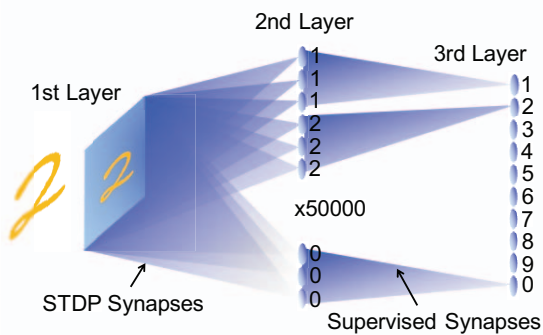


Fig. 11 Schematic illustration of the 3-layer perceptron for MNIST digit learning and recognition.

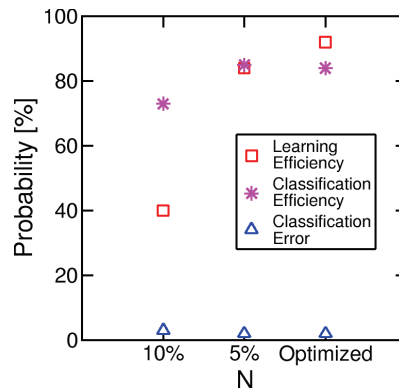


Fig. 12 Calculated learning efficiency and classification efficiency for $N = 5\%$, 10% and for optimized N according to Fig. 10.

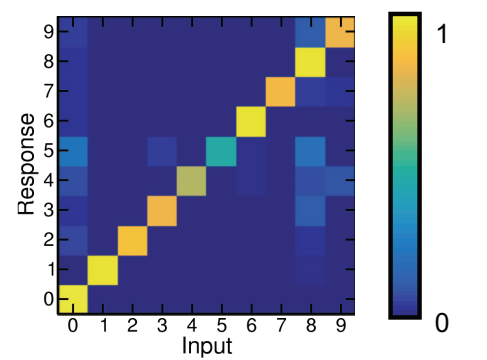


Fig. 13 Classification efficiency of MNIST digits indicating high true positive (diagonal) and low false positive (out of diagonal) classification efficiency.