Beyond-CMOS Non-Boolean Logic Benchmarking: Insights and Future Directions

Chenyun Pan and Azad Naeemi
Electrical and Computer Engineering, Georgia Institute of Technology,
Atlanta, Georgia 30332, USA

***Abstract*—Emerging technologies are facing significant challenges to compete with CMOS with respect to Boolean logic. There is an increasing need for using non-traditional circuits to realize the full potential of beyond-CMOS devices. This paper presents a uniform benchmarking methodology for non-Boolean computation based on the cellular neural network (CNN) for a variety of beyond-CMOS device technologies, including charge-based and spintronic devices. Three types of CNN implementations are investigated benchmarked for a given input noise and recall accuracy target using analog, digital, and spintronic circuits. Results demonstrate that spintronic devices are promising candidates to implement CNNs, where up to 3× EDP improvement is predicted in domain wall devices compared to its conventional CMOS counterpart. This shows that alternative non-Boolean computing platforms are crucial for developing future emerging technologies.**

***Keywords*—cellular neural network; beyond-CMOS technology; performance benchmarking**

I. INTRODUCTION

MANY beyond-CMOS device technologies are being proposed to augment or even replace the conventional Si CMOS technology because of the scaling limit of the CMOS [1, 2]. Some of these devices, such as tunneling and negative-capacitance FETs, can work with lower supply voltages as they may offer steep threshold swings. This can potentially improve the computing energy efficiency because a decrease in the power supply voltage reduces the dynamic energy dissipation in devices and interconnects quadratically. Another major category of emerging devices includes spintronic devices that use magnets and spin transfer torque mechanism to store and process information [3]. These devices can operate with supply voltages of ~100mV or even lower and have an additional feature of non-volatility. Although the intrinsic energy required to switch a very stable magnet at room temperature can be as low as 40kT, these devices are quite energy hungry because of the inefficiency of the spin transfer torque mechanism. Furthermore, the switching delay of the ferromagnet is typically in the nanosecond range as compared to FETs which can switch in less than tens of picoseconds.

Recent benchmarking research based on Boolean circuits, such as 32-bit adders, has projected a limited performance gain for only a few beyond-CMOS device candidates [2], shown in Fig. 1. For spintronic devices, orders of magnitude worse performance in terms of energy-delay product (EDP) has been predicted. Research in the area of beyond-CMOS devices is progressing fast and the proposed devices are being continuously revised and reinvented. Such innovations, which are hard to predict, will with little doubt make emerging devices more competitive. However, one needs to recognize that conventional CMOS devices and their corresponding circuits and architectures have evolved together over many years. Some of the emerging beyond-CMOS devices offer fundamentally different and in some cases unique characteristics because of which novel and nontraditional circuit concepts are needed to realize their full potential.

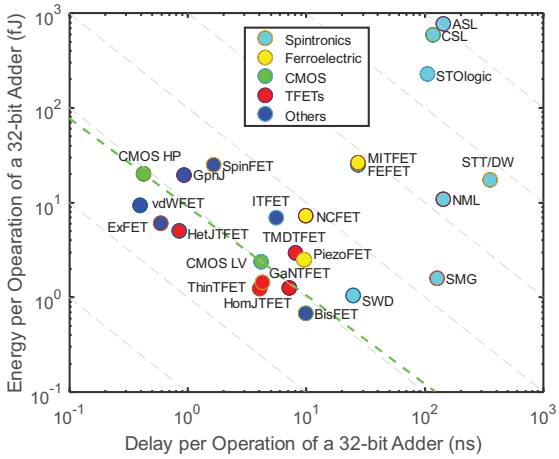

Fig. 1. Energy-delay projections for a 32-bit adder using emerging charge- and spin-based logic devices. Results are adopted from [2].

To better utilize emerging charge- and spin-based technologies, alternative non-Boolean platforms based on neuromorphic circuits are quite attractive [4-6]. Biologically-inspired computing platforms are highly-efficient for solving many problems, particularly in the voice, image, and video processing, by taking advantages of massive parallel low-power computing blocks [7, 8]. Many proposals have studied the neuromorphic systems based on spintronic devices, and they are shown to provide a low energy per operation compared to the conventional CMOS technology [9, 10]. For the charge-based devices, recent studies demonstrated that using TFETs to build a cellular neural network (CNN) can potentially lower the energy per operation thanks to their low supply voltage and steep subthreshold slope [11, 12].

In this paper, a uniform non-Boolean benchmarking is performed for a variety of beyond-CMOS devices based on the CNN architecture. The CNN is a suitable platform for the purpose of benchmarking because a variety of charge- and spin-

978-3-9815370-8-6/17/$31.00 ©2017 IEEE

based devices can be used to implement CNN circuits efficiently [11-13]. Moreover, the mathematical framework for CNN circuits is well defined and understood which facilitates benchmarking various implementations for a given task and desired accuracy. Furthermore, there has been a great deal of research on both digital and analog implementation of CNN circuits with CMOS devices [14, 15]. The benchmarking in this paper covers three CNN implementations based on analog and digital charge-based switches and spintronic devices. The performance is compared in terms of the energy and delay for a given associative memory application with a certain accuracy target and input noise level. It is crucial to understand and identify the advantage and drawback of each device technology by means of a rigorous and fair benchmarking to guide device researchers to develop a device for optimal circuit performance.

II. CNN Cell Implementations

A. Overview

The CNN is a non-Boolean computing architecture that contains an array of computing cells that are connected to nearby cells. Since interconnects are major limitations in modern VLSI systems, CNN systems can take advantage of the local communication and encounter fewer constraints imposed by interconnects. The CNN can be considered as a brain-inspired computing architecture that relies on neurons to integrate the incoming currents. The accumulated and activated output signal drives nearby neurons through weighted synapses. The underlying mathematics of a CNN was proposed by Chua [16] and the dynamic state equation of each CNN cell circuit is written as

$$C_f \frac{dx_{ij}}{dt} = -\frac{1}{R_f} x_{ij} + \sum_{kl \in S_{ij}} A_{ij,kl} y_{kl} + \sum_{kl \in S_{ij}} B_{ij,kl} u_{kl} + I_{ij}, \quad y_{ij} = f(x_{ij}) \tag{1}$$

where x_{ij} is the state voltage of the cell, R_f and C_f are linear resistance and capacitance of each cell, y_{kl} and u_{kl} are the outputs and inputs of neighboring cells, respectively, $f(x)$ is the sigmoid function that describes the characteristic between the output voltage and cell state voltage, A_{kl} and B_{kl} are templates of each cell, whose values represent the weights of synapses connecting two nearby cells, and I_{ij} is the input bias current of each cell.

B. Analog Implementation

Many prior publications have focused on the CNN implemented by analog circuits using CMOS transistors. A widely used implementation is based on operational amplifiers and operational transconductance amplifiers (OTAs) as neurons and synapses, respectively [14, 15]. Some recent work has also investigated CNN using beyond-CMOS charge-based devices, such as TFETs, to potentially improve the energy efficiency [11, 12] thanks to their steep subthreshold slope and low operating voltage. In this work, the delay per operation is numerically solved based on CNN dynamics shown in (1), where the linear feedback capacitance, C_f, is the summation of the input and output capacitances of nearby OTAs to reliably sink current, the feedback resistance, R_f, is set as $2/G_m$ to achieve a stable output, where G_m is the summation of the conductance of input synapses. Since the synapses are realized with OTAs [11], the conductance of the synapse is equal to the transconductance of transistor connecting to the input. In this work, the bias current of the transistor is set as the geometric mean current of ON and OFF currents, which are adopted from the previous benchmarking work [2]. The transistor width is set as 10F, where F is the minimum feature size of 15 nm.

The settling time of the system to reach the equilibrium state is proportional to the RC time constant according to the CNN dynamic equation (1). In general, TFET has a steeper subthreshold slope compared to the conventional CMOS HP and LV devices. In addition, TFET operates at a low supply voltage. However, the low ON current limits the bias current, leading to a slow operation speed. A device with a large bias current and a small subthreshold slope and capacitance can potentially be faster; a device with a small subthreshold slope, supply voltage, and capacitance may potentially consume less energy. These trends can be observed based on more rigorous numerical simulation results of CNN dynamics shown in Section III.

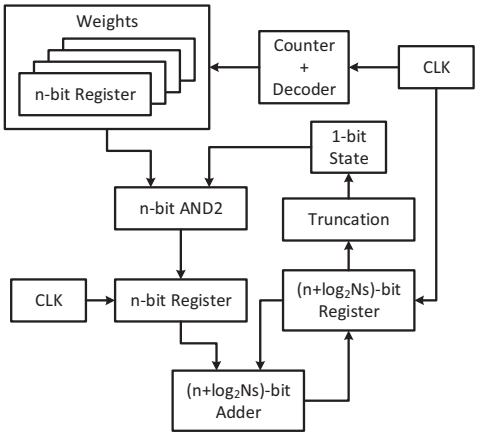

Fig. 2. Diagram of the CNN cell implementation based on the digital circuit.

C. Digital Implementation

The diagram of the digital CNN cell implementation is illustrated in Fig. 2. The n-bit weights of synapses connecting nearby cells are stored in n-bit registers. The following operations are performed in each cell in parallel: 1) The counter is activated by the clock and the output of the decoder will select one weight. 2) The weight gets multiplied by the corresponding one-bit state from either current cell or one of the nearby cells. The multiplication is performed by the n-bit two-input AND gates, whose outputs are stored in the n-bit register. 3) At the next clock cycle, the weighted state gets added by the (n+$\log_2$Ns)-bit adder, and the output is stored in an (n+$\log_2$Ns)-bit register, where Ns is the number of synapses connecting to the cell, namely the number of weights. 4) After all weighted states are summed, the final weighted state is truncated and updates the one-bit state in the current cell. After all cell states in the CNN system are updated, the system goes to the next time step. This iteration continues until the CNN system reaches the

steady state. For the performance modeling, the delay and energy dissipation of the register, counter, decoder, two-input AND gate, and adder follows the previous benchmarking work for the Boolean logic circuits [2].

D. Spintronic Implementation

In this subsection, CNN implementations based on three major types of current-driven spintronic devices are investigated, including the spin diffusion, spin Hall effect (SHE), and domain wall devices. Magnet switching dynamics follow the Landau-Lifshitz-Gilbert (LLG) equation with a spin-transfer-torque term [17, 18]. The magnetization of a magnet, $\vec{m}$, under a perpendicular spin-polarized current, $\vec{I}_{S,\perp}$, is

$$\frac{d\vec{m}}{dt} = -\gamma\mu_0[\vec{m}\times\vec{H}_{eff}] + \alpha\left[\vec{m}\times\frac{d\vec{m}}{dt}\right] + \frac{\vec{I}_{S,\perp}}{qN_s}, \qquad (2)$$

where $\vec{H}_{eff}$ is the effective field, γ is the gyro ratio, μ_0 is the permittivity, α is the damping factor, q is the elementary charge, N_s is the number of magnetons, and $\vec{I}_{S,\perp}$ can be expressed as $i_0 \cdot \left(\sum_{kl\in S_{ij}} A_{ij,kl}y_{kl} + \sum_{kl\in S_{ij}} B_{ij,kl}u_{kl} + I_{ij}\right)$, where i_0 is the unit spin-polarized current when the template value is unity. The amplitude and the direction of the spin-polarized current depend on the output and input voltage polarities of nearby cells and the weights of synapses connecting those cells.

1) Spin Diffusion based Device

The CNN using spin diffusion based devices has been investigated thoroughly in a previous study [13]. It relies on the all-spin logic (ASL) as the basic building block, where PMA magnets are assumed in the simulation. For the CNN benchmarking in this paper, IMA magnets are also included for the spin diffusion based devices.

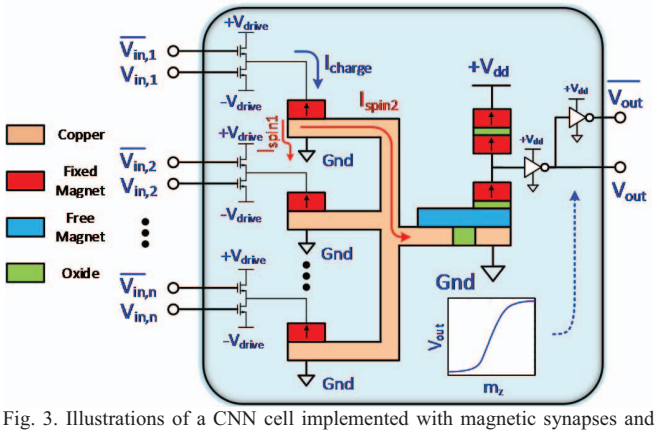

Fig. 3. Illustrations of a CNN cell implemented with magnetic synapses and neurons [13].

Fig. 3 illustrates a CNN implemented with magnetic synapses and neurons [13]. The inputs of the CNN cell, V_{in} and $\overline{V_{in}}$, come from both inputs, u_{kl}, and outputs, y_{kl}, of nearby cells, which are connected to n-type driving MOSFETs with various driving sizes determined by the absolute values of templates A_{ij} and B_{ij}. Depending on the input voltage, one of the driving transistors is turned on, generating a charge current. This charge current gets spin-polarized, and the injected (or extracted) spins diffuse towards the free magnet (the neuron). The polarity of the spin current depends on the magnetization direction of the magnet and the charge current direction. For instance, if the input, V_{in}, is logic zero, the charge current flows from $+V_{drive}$ through the input magnet into the ground connection underneath the magnet, as shown by the blue arrow in Fig. 3. Electrons with spins that are opposite to the magnetization direction of the magnet accumulate underneath the magnet. This spin accumulation creates a spin-polarized current, which separates into two paths, I_{spin1} and I_{spin2}, toward the ground and the output free magnet, respectively, as shown by red arrows in Fig. 3. These two spin-polarized currents satisfy the spin drift-diffusion equation [19], which is the function of the spin accumulation. The boundary condition of the spin accumulations underneath the output magnet and the ground contact are zero, absorbing spins coming from the input magnets. For the simplicity, the ground is drawn to be right underneath the input magnets, but in reality, there are long interconnects in the ground distribution network. This ground interconnect is much longer compared to the distance between the input and output magnets. Therefore, the majority of the spins will diffuse and be absorbed by the output magnet instead of the ground, and this spin current is I_{spin2}.

For CNN cells with multiple inputs, spin-polarized currents from various input magnets superimpose, achieving the integration functionality. The overall net current inserts a spin torque into the output magnet, setting the magnetization direction of the free magnet in the neuron. The magnet magnetizations are preset to implement signs of template values. The amplitude of voltage source, V_{drive}, can be substantially smaller than the V_{dd} to allow low-power operation and improve the energy efficiency.

The magnetization direction of the output magnet is read by building two magnetic tunneling junctions (MTJs). The top MTJ is set as the reference MTJ with a parallel configuration. For the bottom MTJ, if the magnetization of the free magnet is in the same direction as that of the fixed magnet, it has a parallel configuration and a small resistance, lowering the voltage between two MTJs; if the magnetization of the free magnet is in the opposite direction, a high voltage is generated. Since the output voltage of each cell needs to switch driving transistors in multiple synapses in nearby cells, an inverter is added at the output to amplify the voltage between two MTJs and provide a large voltage swing between the ground and V_{dd}. The corresponding MTJ parameters are from the experimental data reported in [20]. Since the sensing current from V_{dd} to ground is much smaller compared to the critical switching current, the output magnet will not be disturbed during the read.

2) Spin Hall Effect based Device

Fig. 4 (a) shows the schematic of a CNN cell implemented by MTJs as synapses and SHE based device as the neuron. In this example, each 3-bit synapse has two fixed resistances and three MTJs that are digitally programmable to achieve eight different combinations of parallel and antiparallel states. Each combination represents one weight, and overall eight quantized

output currents can be generated accordingly, shown in the bar chart in Fig. 4 (b). The resistances are adjusted such that the current linearly increases with the weight. Depending on the input voltage polarity, both positive and negative weights can be realized.

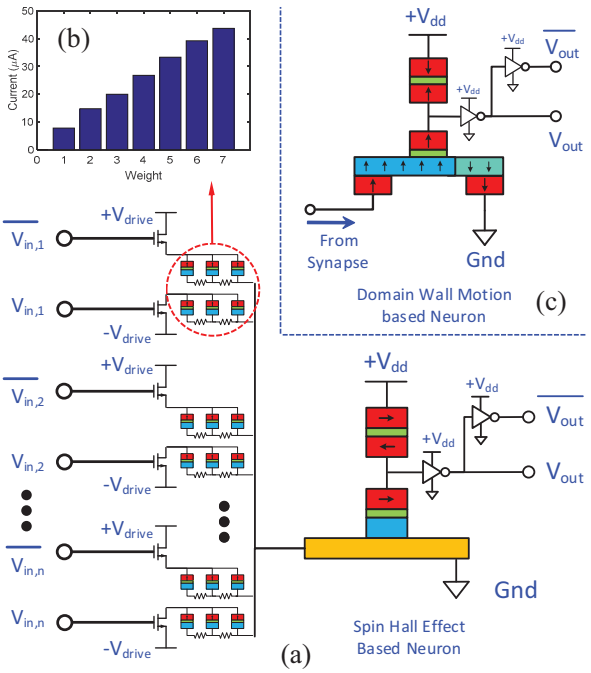

Fig. 4. Schematic of spintronic CNN implementations based on spin Hall effect and domain wall motion.

During the CNN operation, multiple synapses connecting nearby cells are connected to the neuron, and the net charge current flowing through the SHE material in the neuron is converted into spin currents due to the spin orbit coupling [21]. The spin-polarized current density is $J_s = J_c\theta$, where θ is the spin Hall angle at a value of 0.3 [22], J_c is the charge current density. The dimension of the SHE material is 150×60×2 nm^3 with a resistance of 60 Ω. To further improve the spin current received in the magnet, recent work has shown that adding an extra layer of the copper plate between the SHE material and the free magnet can enhance the spin injection through the lateral diffusion [23]. For the benchmarking results shown in Section III, an enhancement factor of 2 is considered to show the potential improvement by using the copper collector.

The read-out circuitry is identical to the spin diffusion based CNN by using two MTJs [13]. The voltage at the input of the inverter becomes low and high when the bottom MTJ is at parallel and antiparallel configurations, respectively. The complementary voltages are generated to drive the synapses in nearby cells. The dynamic of the magnet orientation of each cell is numerically solved based on (2).

3) Domain Wall Motion based Device

Another CNN implementation is based on using a domain wall device as the neuron, shown in Fig. 4 (c). By replacing the spin Hall effect material with the domain wall magnet, the resistance of the bottom MTJ depends on the position of the domain wall underneath the fixed magnets. For instance, if the input electrons flow to the right direction, the domain wall moves to the right and the bottom MTJ is at the parallel configuration, lowering the voltage at the input of the inverter, and the V_{out} rises to V_{dd}. The domain wall magnet size is 150×30×2 nm^3 with a resistance of 150 Ω, and the relation between the domain wall speed c_{dw} and the input current density are adopted from [24].

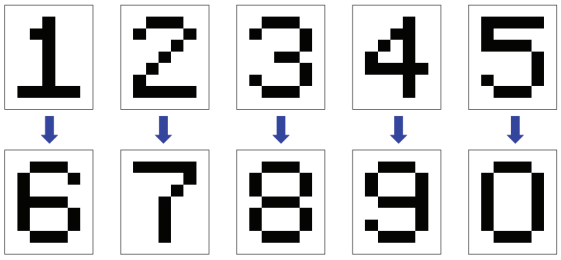

Fig. 5. Training patterns of the associative memory application, where digital numbers '1' – '5' are associated with '6' – '0'.

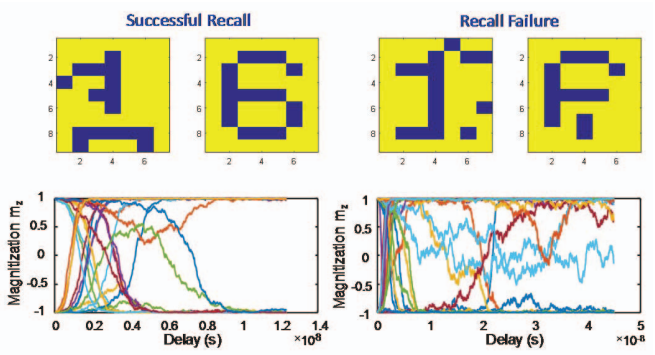

Fig. 6. Functional demonstration of a spintronic CNN using ASL devices with PMA magnets as the basic building blocks. (a) Successful recall and (b) failure recall using pattern '1' as the input with 10 % noisy pixels.

III. Benchmarking Methodology and Results

A. Functional Demonstration

The associative memory application is widely used in the tasks of voice and image recognition, which can be efficiently performed in the CNN architecture [25, 26]. In this section, three types of CNN implementations are investigated to perform the pattern recall task, shown in Fig. 5. Top five digital numbers, '1' – '5', are associated with bottom five numbers, '6' – '0'. The training method used for storing patterns is adopted from the Hebbian learning algorithm [27]. It can be applied for a large number of free parameters in a space-varying template used in the associative memory with a fair computational cost and a good convergence speed. Once the training is finished, a digital pattern with certain random noisy pixels is set as the input. Here, a spintronic CNN based on spin diffusion as the writing mechanism using PMA magnets is shown in Fig. 6 as an example to demonstrate the functionality of the application. A noisy pattern '1' is used as the input and the output is expected to be associated with the pattern '6'. The simulations are performed at the room temperature, and the thermal noise is taken into account. Depending on the random input noise and the thermal noise, the output has a probability of successful

recall. In addition, the delay per CNN operation is dependent on the input pattern, input noise, and the thermal noise.

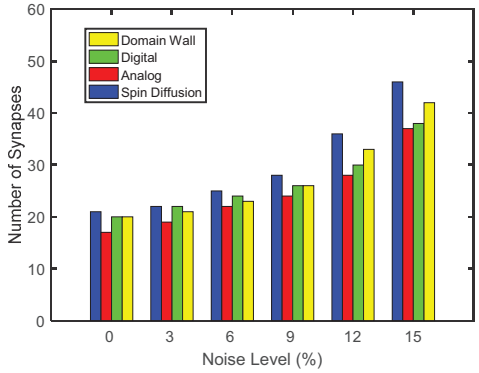

Fig. 7. Required number of synapses versus noise levels of the input patterns for four CNN implementations based on analog, digital, PMA, IMA, and domain wall devices at 80% recall accuracy.

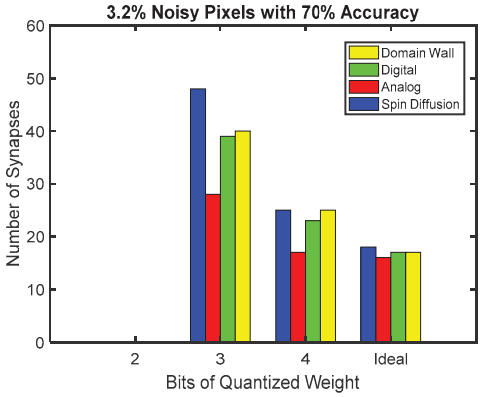

Fig. 8. Required number of synapses versus the number of bits used in quantized weights with 3% noisy pixels at the input patterns for four CNN implementations based on analog, digital, spin diffusion, and domain wall devices at 70% recall accuracy.

B. Iso-Accuracy Analysis

Since different CNN implementations may have different accuracy, iso-accuracy analyses are important to achieve a fair benchmark among various types of CNN. For each input pattern shown in Fig. 5, 100 Monte-Carlo simulations are performed for the associate memory application with a given number of random noisy pixels. The recall accuracy is defined as the number of output patterns that completely match with the associated patterns during the training. For a given recall accuracy of 80%, Fig. 7 shows the required number of synapses for four CNN implementations, and the analog CNN requires fewer synapses compared to other CNNs. By increasing the number of synapses for each neuron, the recall accuracy can be increased significantly. This is because each cell can reach and communicate with more nearby cells and improve the probability of the successful recall. This improvement comes at the cost of a larger footprint area, energy dissipation, and training cost. For a given number of synapses, the recall accuracy differs among various CNN implementations because of the differences in CNN characteristics, such as the sigmoid function, the dynamic behavior of the magnets, and the feedback of the analog integrators. The analog CNN provides a better recall accuracy for processing input patterns with few noisy pixels.

The results shown above are based on synapses with ideal weights. To quantify the impact of the finite resolution of synapse weights on the recall accuracy, the numbers of required synapses for different CNN implementations to achieve 70% recall accuracy are shown in Fig. 8. Here, the input noise level is set as 3%. One can observe that there is a trade-off between the number of synapses and the number of bits representing the weights of synapses. With a two-bit weight, no CNN implementation can reach an 70% accuracy even if all cells are connected with each other. As the number of bits representing the synapse weight increases, the required number of synapses keeps decreasing. Compared to CNNs using 4-bit synapses, ~50% more synapses are required for ones using 3-bit synapses. Therefore, using 4-bit weights provides a good trade-off and also imposes small overheads compared to the ideal weights.

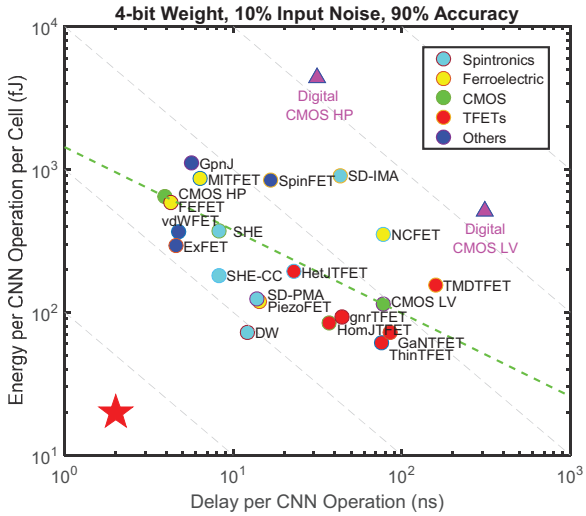

Fig. 9. Comparison of energy and delay per operation among various beyond-CMOS technologies based on analog, digital, and spintronic implementations. Triangle and circular points of charge-based devices represent the digital and analog CNN implementation, respectively. For the text labels of spintronic CNN implementation, SD, SHE, and DW stand for spin diffusion, spin Hall effect, and domain wall motion, respectively, and CC represents for the copper collector.

Fig. 9 shows the performance comparison among three types of CNN implementations using 4-bit weight synapses to achieve 90% recall accuracy for a given input noise of 10%. For the charge-based CNN implementation, CMOS HP and LV devices are employed to quantify the performance of the digital CNN and to compare against their analog counterparts. It is shown that the digital CNNs are quite power hungry and slow. This is because multiple cycles are required to read out the weights from the register and perform the summation in the adder, which is energy and time consuming. In general, the analog CNNs implemented by TFETs dissipate less energy thanks to their steep subthreshold slope and lower supply voltage. For devices with an extra ferroelectric switching time, such as FEFET, MITFET, PiezoFET, and NCFET, their relative positions in the energy-delay plot shift largely toward the preferred corner compared to the results shown in the previous

Boolean logic benchmarking [2]. The reason is that the extra polarization switching time of the ferroelectric material can be comparable or even larger than the intrinsic switching delays of FETs in a Boolean circuit; however, for the CNN application, the settling time is dominated by the product of the feedback resistance and capacitance.

Comparing the benchmarking results for Boolean and non-Boolean circuits, shown in Fig. 1 and Fig. 9, respectively, one clear trend is that spintronic devices shift much closer to the preferred corner, and they are competitive compared to charge-based devices. This is because a single magnet can mimic the functionality of a neuron, and these spintronic devices operate at a low supply voltage. For the domain wall device, it provides the best performance in terms of the EDP thanks to its low critical current requirement. The spin diffusion based CNN with IMA magnets consumes more energy due to the large critical current required to switch the magnet.

Envisioning the development of the future emerging technologies, one need to find suitable circuits to complement devices to fully take the advantage of their unique characteristics. Most likely there will be no single post-CMOS device or material that can replace CMOS in a sufficiently beneficial way while other technology aspects or design layers can remain the same. Alternative circuit platforms based on non-Boolean circuits, such as neuromorphic circuits, are attractive and have the potential of delivering more energy-efficient computing.

IV. Conclusions

In this paper, a uniform benchmarking methodology is presented for the non-Boolean computation based on the CNN architecture. A variety of beyond-CMOS device technologies, including charge-based and spintronic devices, are compared based on three types of CNN implementations, using analog, digital, and spintronic circuits. Results demonstrate that TFET-based CNNs, in general, consume less energy thanks to their steep threshold slope and low supply voltage. CNNs implemented by digital CMOS perform worse compared to their analog counterpart due to the large energy and delay from multiplying and adding synapse weights. Spintronic devices are promising candidates for implementing CNN. The results of this benchmarking are in sharp contrast to those for Boolean functions, indicating the importance of exploring alternative non-Boolean circuits for realizing the full potential of future emerging technologies.

Reference

[1] K. J. Kuhn, "CMOS scaling for the 22nm node and beyond: Device physics and technology," in *VLSI Technology, Systems and Applications (VLSI-TSA), 2011 International Symposium on*, 2011, pp. 1-2.

[2] D. E. Nikonov and I. A. Young, "Benchmarking of Beyond-CMOS Exploratory Devices for Logic Integrated Circuits," *Exploratory Solid-State Computational Devices and Circuits, IEEE Journal on,* vol. 1, pp. 3-11, 2015.

[3] S. A. Wolf, J. Lu, M. R. Stan, E. Chen, and D. M. Treger, "The promise of nanomagnetics and spintronics for future logic and universal memory," *Proceedings of the IEEE,* vol. 98, pp. 2155-2168, 2010.

[4] H. Markram, "The blue brain project," *Nature Reviews Neuroscience,* vol. 7, pp. 153-160, 2006.

[5] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, and Y. Nakamura, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science,* vol. 345, pp. 668-673, 2014.

[6] H. Aghasi, R. M. Iraei, A. Naeemi, and E. Afshari, "Smart Detector Cell: A Scalable All-Spin Circuit for Low Power Non-Boolean Pattern Recognition," *IEEE Transactions on Nanotechnology,* vol. 15, pp. 356-366, 2016.

[7] D. Monroe, "Neuromorphic computing gets ready for the (really) big time," *Communications of the ACM,* vol. 57, pp. 13-15, 2014.

[8] S. Venkataramani, A. Ranjan, K. Roy, and A. Raghunathan, "AxNN: energy-efficient neuromorphic systems using approximate computing," in *Proceedings of the 2014 international symposium on Low power electronics and design*, 2014, pp. 27-32.

[9] S. G. Ramasubramanian, R. Venkatesan, M. Sharad, K. Roy, and A. Raghunathan, "SPINDLE: SPINtronic deep learning engine for large-scale neuromorphic computing," in *Proceedings of the 2014 international symposium on Low power electronics and design*, 2014, pp. 15-20.

[10] A. Sengupta, S. H. Choday, Y. Kim, and K. Roy, "Spin orbit torque based electronic neuron," *Applied Physics Letters,* vol. 106, p. 143701, 2015.

[11] A. R. Trivedi and S. Mukhopadhyay, "Potential of Ultralow-Power Cellular Neural Image Processing With Si/Ge Tunnel FET," *Nanotechnology, IEEE Transactions on,* vol. 13, pp. 627-629, 2014.

[12] I. Palit, X. S. Hu, J. Nahas, and M. Niemier, "TFET-based cellular neural network architectures," in *Proceedings of the 2013 International Symposium on Low Power Electronics and Design*, 2013, pp. 236-241.

[13] C. Pan and A. Naeemi, "A Proposal for Energy-Efficient Cellular Neural Network based on Spintronic Devices," *IEEE Transactions on Nanotechnology,* vol. 15, pp. 1-8, 2016.

[14] L. Yang, L. Chua, and K. Krieg, "VLSI implementation of cellular neural networks," in *Circuits and Systems, 1990., IEEE International Symposium on*, 1990, pp. 2425-2427.

[15] E. Y. Chou, B. J. Sheu, and R. C. Chang, "VLSI design of optimization and image processing cellular neural networks," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on,* vol. 44, pp. 12-20, 1997.

[16] L. O. Chua and L. Yang, "Cellular neural networks: theory," *Circuits and Systems, IEEE Transactions on,* vol. 35, pp. 1257-1272, 1988.

[17] D. C. Ralph and M. D. Stiles, "Spin transfer torques," *Journal of Magnetism and Magnetic Materials,* vol. 320, pp. 1190-1216, 4// 2008.

[18] D. V. Berkov and J. Miltat, "Spin-torque driven magnetization dynamics: Micromagnetic modeling," *Journal of Magnetism and Magnetic Materials,* vol. 320, pp. 1238-1259, 2008.

[19] T. Valet and A. Fert, "Theory of the perpendicular magnetoresistance in magnetic multilayers," *Physical Review B,* vol. 48, p. 7099, 1993.

[20] S. Yuasa, T. Nagahama, A. Fukushima, Y. Suzuki, and K. Ando, "Giant room-temperature magnetoresistance in single-crystal Fe/MgO/Fe magnetic tunnel junctions," *Nature materials,* vol. 3, pp. 868-871, 2004.

[21] A. Hoffmann, "Spin Hall effects in metals," *Magnetics, IEEE Transactions on,* vol. 49, pp. 5172-5193, 2013.

[22] C.-F. Pai, L. Liu, Y. Li, H. Tseng, D. Ralph, and R. Buhrman, "Spin transfer torque devices utilizing the giant spin Hall effect of tungsten," *Applied Physics Letters,* vol. 101, p. 122404, 2012.

[23] S. Sayed, V. Q. Diep, K. Y. Camsari, and S. Datta, "Spin Funneling for Enhanced Spin Injection into Ferromagnets," *Scientific Reports,* vol. 6, 2016.

[24] D. Morris, D. Bromberg, J.-G. J. Zhu, and L. Pileggi, "mLogic: Ultra-low voltage non-volatile logic circuits using STT-MTJ devices," in *Proceedings of the 49th Annual Design Automation Conference*, 2012, pp. 486-491.

[25] M. Namba and Z. Zhang, "Cellular neural network for associative memory and its application to Braille image recognition," in *Neural Networks, 2006. IJCNN'06. International Joint Conference on*, 2006, pp. 2409-2414.

[26] D. Liu and A. N. Michel, "Cellular neural networks for associative memories," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on,* vol. 40, pp. 119-121, 1993.

[27] P. Szolgay, I. Szatmari, and K. Laszlo, "A fast fixed point learning method to implement associative memory on CNNs," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on,* vol. 44, pp. 362-366, 1997.