An Energy-efficient Classifier via Boosted Spin Channel Networks

Ameya D. Patil*, Sasikanth Manipatruni†, Dmitri Nikonov†, Ian A. Young† and Naresh R. Shanbhag*
*Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801
†Intel Corp., Hillsboro, OR 97124

***Abstract*—With diminishing energy and delay benefits via CMOS scaling, there is much interest in exploring the use of alternative state variables such as electronic spin. Multiple research efforts are underway exploring both Boolean and non-Boolean design space using spin devices in order to make their energy and delay benefits competitive to CMOS. In this paper, we propose *spin channel networks (SCNs)* – spin-based circuits that exploit exponential decay of spin current to efficiently realize multi-bit dot product computation. We show that proposed SCNs can be employed with adaptive boosting (AdaBoost) learning algorithm to efficiently realize a binary classifier for breast cancer detection. The proposed SCN implementation achieves $112\times$ and $14\times$ lower energy per decision compared to the conventional all spin logic (ASL) and 20 nm CMOS designs, respectively, for identical decision throughput.**

I. Introduction

Emerging applications, such as recognition, mining, and synthesis, in the Internet-of-Things (IoT) era require highly efficient implementations of inference algorithms on the resource-constrained edge platforms. With diminishing energy and delay benefits via CMOS scaling, there is much interest in exploring the use of alternative state variables such as spin of electron [1], [2] for computing. One promising example of spin-based devices proposed for Boolean logic computation is all spin logic (ASL) device [3]. ASL devices offer unique advantages such as non-volatility, high logic efficiency and ultra-low operating voltages However, ASL turns out to be non-competitive compared to CMOS in terms of energy consumption and delay of Boolean logic implementations [4], [5], mainly due to the large delay required to switch nanomagnet deterministically, exponential decay of spin currents along the spin channels and because ASL gates consume static power [5], [6].

There have been multiple research efforts to improve the energy-efficiency of spin-based implementations. Several research efforts have explored the neuromorphic design space using spin-devices in order to achieve energy-efficiency [7]–[11]. Multiple works have also exploited the stochastic nature of nanomagnetic switching to achieve energy-efficiency [12]–[16]. There also exists works at the architectural-level to fully exploit the advantages of emerging spin-device configurations such as racetrack memory [17]–[21].

In this paper, we propose *spin channel networks (SCNs)* – spin-based circuits that exploit exponential decay of spin current to efficiently realize multi-bit dot product computation. SCNs employ same building blocks as in ASL. However, unlike ASL, SCNs do not need nanomagnetic switching at multiple intermediate stages (as shown in Fig. 1), thus reducing energy consumption and delay significantly. We show that proposed SCNs can be employed with adaptive boosting (AdaBoost) learning algorithm to efficiently realize a binary classifier for breast cancer detection. The proposed SCN implementation achieves $112\times$ and $14\times$ lower energy per decision compared to the conventional ASL-based and 20 nm CMOS designs, respectively, for identical decision throughput.

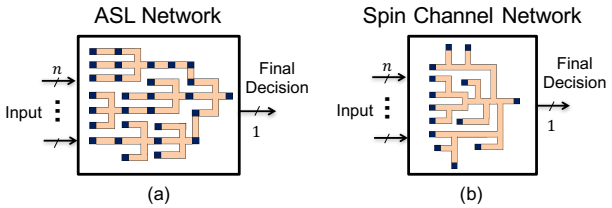

Fig. 1. Illustration of (a) an all spin logic (ASL) network, and (b) a spin channel network (SCN) implementing an N-dimensional dot product based classifier. Nanomagnets are depicted as dark squares.

II. Background

A. All Spin Logic (ASL) Devices

Figure 2 shows a diagram of a single ASL device. It consists of two nanomagnets separated by a conducting channel. As charge current is passed through the input nanomagnet, spin current propagates through the channel to exert a torque on the magnetization of the output nanomagnet, forcing it to switch. Since ASL is a current-based device, it can be operated at ultra-low supply voltages. However, charge current in the order of 10-100 μA is required to generate sufficient spin current to switch the output magnet. Hence, the charge current needs to be switched off immediately after switching to avoid static power consumption. Works in [5], [22] propose to clock these devices via a MOSFET turning on the ASL device only when it needs to compute.

Figure 3 identifies two SCN primitives (derived from ASL) from which complex SCNs will be composed in section III. The primitives are a nanomagnet with a spin channel (channelized nanomagnet) and a spin channel. Channelized nanomagnet takes an input charge current I_c and injects a proportional spin current $I_{\text{s,o}}$ in to the channel, where β_m is a proportionality constant that depends upon the material and geometry, including the channel length L_c. The input spin current $I_{\text{s,in}}$ into a spin channel of length L is attenuated by a factor of $e^{\frac{L}{\lambda}}$ to generate an output spin current $I_{\text{s,out}}$, where

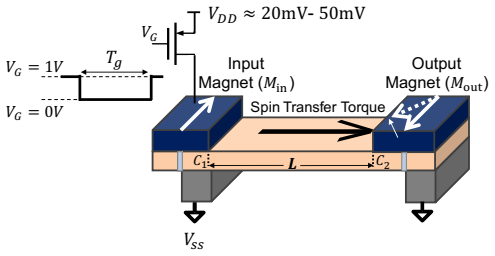

Fig. 2. All spin logic (ASL) device with a power gating transistor [5].

	Conceptual diagram	Transfer function	Symbol	Layout
Channelized nano-magnet	I_c, m, $I_{s,o}$, L_c	$I_{s,o} = \beta_m I_c m$	I_c, $I_{s,o}$	Via 1, Via 2, $L_c = 5F$, Ferromagnet, Metal 1
Spin channel	$I_{s,in}$, $I_{s,o}$, L	$I_{s,o} = I_{s,in} e^{-\frac{L}{\lambda}}$	L, $I_{s,in}$, $I_{s,o}$	L

Fig. 3. SCN primitives derived from ASL: symbol, transfer function, and layout. Each layout grid cell is of size $\frac{F}{2} \times \frac{F}{2} = 7.5\,\text{nm} \times 7.5\,\text{nm}$ [25].

λ is the spin flip length [6], [23]. The layouts are obtained by following the λ-rules defined in [4].

B. Linear Support Vector Machine (SVM)

Linear SVM [24] is a simple and popular machine learning algorithm for binary classification. The SVM employs a hyperplane to separate the training feature vectors into two regions as shown below:

$$\mathbf{w}^T\mathbf{x} + b \underset{\hat{y}=-1}{\overset{\hat{y}=1}{\gtrless}} 0 \tag{1}$$

where $\mathbf{w}$ and b denote the trained weight vector and bias representing the separating hyperplane, respectively, $\mathbf{x}$ denotes the N-dimensional input feature vector, and $\hat{y}$ denotes its label predicted by the SVM. If the true label is denoted by y, the accuracy of SVM is given by the probability of classification error $p_e = \Pr\{\hat{y} \neq y\}$, which can be empirically estimated for a given dataset.

C. Classifier Ensemble via Adaptive Boosting (AdaBoost)

A classifier ensemble consists of M weak classifiers. Each weak classifier is computationally simple but inaccurate. However, the weak classifier decisions can be combined to obtain highly accurate final decision. Adaptive boosting (AdaBoost) [26] is a technique to train these weak classifiers sequentially. Each weak classifier is specifically trained to correct the errors made by the other weak classifiers trained earlier (see [26] for the training algorithm). Let the output label of ith weak classifier be denoted as $\hat{y}_i = f_{\mathbf{w}_i}(\mathbf{x})$, where $f_{\mathbf{w}_i}(.)$ denotes the ith weak classifier function parametrized by the weight vector $\mathbf{w}_i$ computed during training. The final decision $\hat{y}_f$ is computed as shown below:

$$\sum_{i=1}^{M} \alpha_i \hat{y}_i \underset{\hat{y}_f=-1}{\overset{\hat{y}_f=1}{\gtrless}} 0 \tag{2}$$

where output weights α_is of the linear combiner are also learned during training.

III. Spin Channel Networks (SCNs)

SCNs exploit the exponential decay $e^{-(\frac{L}{\lambda})}$ of spin currents with the channel length L (see section II-A) to obtain powers-of-2 weighing of spin currents by choosing L to be the multiples of $\lambda \ln 2$. In this section, we describe the design of two components of SCNs, namely a spin channel network multiplier (SCNM) and a stochastic slicer.

A. Spin Channel Network Multiplier (SCNM)

1) Schematic: Figure 4(a) shows a schematic of a 4×4 bit spin channel network multiplier (SCNM). Each nanomagnet M_{ij} is placed at a distance of $(7-i-j)\lambda \ln 2$ from the output node, resulting in the output spin current $I_{s,o}$ as follows:

$$I_{s,o} = \frac{\beta_m I_c}{2^{-7}} \sum_{i,j=0}^{3} a_i b_j 2^{(i+j)} = \frac{\beta_m I_c}{2^{-7}} \sum_{k=0}^{6} 2^k \underbrace{\left(\sum_{\substack{i,j=0 \\ i+j=k}}^{3} a_i b_j \right)}_{p_k} \tag{3}$$

where I_c denotes the ON current of NMOS, $a_i, b_i \in \{0,1\}$ are the bits of two 4 bit unsigned binary operands A and B, respectively, β_m denotes a proportionality constant, and p_k is the sum of partial products $a_i b_j$ such that $i+j=k$. The signs of these operands can be accounted for by changing the magnetization vector direction of the corresponding nanomagnets (for A), and by using a differential supply [8] (for B). The energy consumption of SCNM is given by:

$$E_{\text{mult}}(A,B) = \sum_{i,j=0}^{3} a_i b_j \Big[I_c^2 T_g \big(R_{\text{spin}} + R_{\text{mos}}\big) + C_g V_g^2 + E_{\text{and}} \Big] \tag{4}$$

where R_{spin} denotes the series resistance of the nanomagnet and channel, E_{and} denotes the energy consumption of the AND gate, while R_{mos} and C_g denote the ON resistance and gate capacitance of the transistor, respectively. A gate voltage of V_g is applied to switch ON the NMOS, and T_g is its ON duration. The NMOS and the AND gate constitute the *CMOS driver* for the corresponding nanomagnet.

2) Layout: SCNM schematic in Fig. 4(a) only conveys the SCN functionality at a very high level. However, it does not account for spin current branching at the spin channel junctions and the physical constraint of achieving appropriate spin channel lengths while meeting all lithographic λ-rules. We account for such design constraints in the layouts. We follow the λ-rules stated in [4] and choose $F = 15\,\text{nm}$. Figure 4(b) shows the layout of a 4×4 bit channel multiplier. Each marked joint J_k generates spin current corresponding to p_k defined in (3). Thus, minimum channel length between any two consecutive joints on any branch corresponds to the spin current weighing of 2^{-3}, hence allowing sufficient spacing to satisfy lithographic constraints. The actual channel lengths in the layout are chosen via extensive simulations using SPICE-based circuit models of spin current injection and propagation in spin devices [23] with material parameters in [27], and the nanomagnet dimensions are $30 \times 30 \times 10\ \text{nm}^3$.

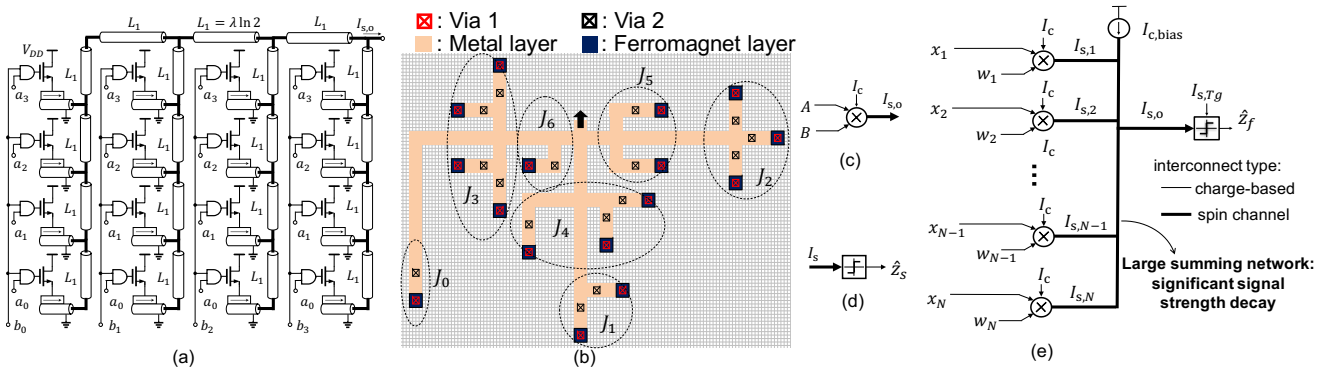

Fig. 4. Spin channel networks: (a) schematic of a 4×4 bit spin channel network multiplier (SCNM), (b) a layout of a 4×4 bit SCNM, (c) SCNM symbol, (d) stochastic slicer symbol, and (e) the SCN-based N dimensional linear SVM classifier architecture. The layout grid cell is of size $7.5\,\text{nm} \times 7.5\,\text{nm}$ [25].

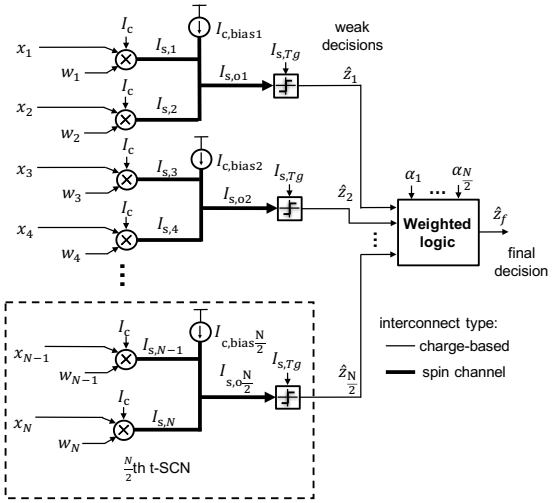

Fig. 5. Boosted SCN architecture comprising an adaptively boosted ensemble of $\frac{N}{2}$ tiny SCNs (t-SCNs), where each t-SCN consists of 2 SCNMs in parallel and one stochastic slicer to implement a 2-dimensional linear SVM classifier.

B. Stochastic Slicer

The output of SCNM is an analog spin current. We use final nanomagnet to threshold the output spin current to produce the final inference decision $\hat{y}$, represented by its magnetization vector direction. Since the nanomagnetic switching is stochastic due to thermal noise in the nanomagnet [28], we refer to the final nanomagnet as a *stochastic slicer*. For a given duration T_g, its switching probability $p_{\text{sw}}(I_s)$ is given by:

$$p_{\text{sw}}(I_s) \approx \left(\frac{1}{2}\right)^{\left[\left(\frac{\beta_1}{\ln 2}\right)^{-\left(\frac{I_s - I_{\text{th}}}{I_{\text{th}}}\right)}\right]} \tag{5}$$

where I_s denotes the input spin current, I_{th} denotes the spin current for which $p_{\text{sw}}(I_{\text{th}}) \approx 0.5$, and $\beta_1 = \frac{\pi^2 E_b}{4kT}$ with E_b denoting the energy barrier of the nanomagnet.

The stochastic slicer realizes following thresholding operation:

$$I_s \underset{\hat{y}=-1}{\overset{\hat{y}=1}{\gtrless}} I_{\text{th}} \tag{6}$$

with a probability of switching error that can be computed as a function of $p_{\text{sw}}(I_s)$ defined in (5). Thus, there exists a trade off between input spin current magnitude I_s (proportional to energy consumption) and switching accuracy, leading to a minimum energy operating point (MEOP) for a target switching accuracy.

IV. SCN-based Binary Classifiers

A. Tiny Spin Channel Networks (t-SCNs)

Figure 4(e) shows a SCN-based N-dimensional linear SVM implementation consisting of N parallel SCNMs (symbol in Fig. 4(c)) followed by a stochastic slicer (symbol in Fig. 4(d)). However, since spin current decays exponentially, such implementation suffers from significant signal degradation over the summing network. Hence, in this paper, we restrict $N = 2$, which requires only two SCNMs vertically facing each other, resulting in a small summing network. The resulting SCN-based implementation of 2-D SVM is referred to as tiny spin channel network (t-SCN). It implements following operation,

$$\left[\mathbf{w}^T(\mathbf{x} + \mathbf{d})\right] I_{\text{s,lsb}} + I_{\text{s,bias}} \underset{\hat{z}_f=-1}{\overset{\hat{z}_f=1}{\gtrless}} I_{\text{th}}, \tag{7}$$

where $\mathbf{w}$ is defined in (1), $\mathbf{d}$ denotes a constant vector with all its components equal to d, and $I_{\text{s,lsb}}$ denotes the spin current corresponding to the least significant partial product for a given ON current I_c of the NMOS. The required $I_{\text{s,bias}}$ is given as,

$$I_{\text{s,bias}} = I_{\text{th}} + bI_{\text{s,lsb}} - \left[d\Big(\sum_i w_i\Big)\right] I_{\text{s,lsb}}, \tag{8}$$

where b denotes SVM bias defined in (1). The bias current $I_{\text{s,bias}}$ is generated by having an additional nanomagnet with a supply current $I_{\text{c,bias}}$ as shown in Fig. 4(e). If the signed precision of x_i is M bit, we choose d to be 2^{M-1}. This makes $(x_i + d)$ an unsigned number, removing the need for differential supply, and reducing the magnitude of $I_{\text{s,bias}}$.

B. Classifier Dimensionality Scaling via Boosted t-SCNs

We employ AdaBoost to compose an arbitrary N-dimensional binary classifier using multiple t-SCNs. Figure 5 shows an architecture of boosted t-SCNs classifier. Each t-SCN, employed an a weak classifier, observes only two unique dimensions of input feature vector $\mathbf{x}$ and computes its local

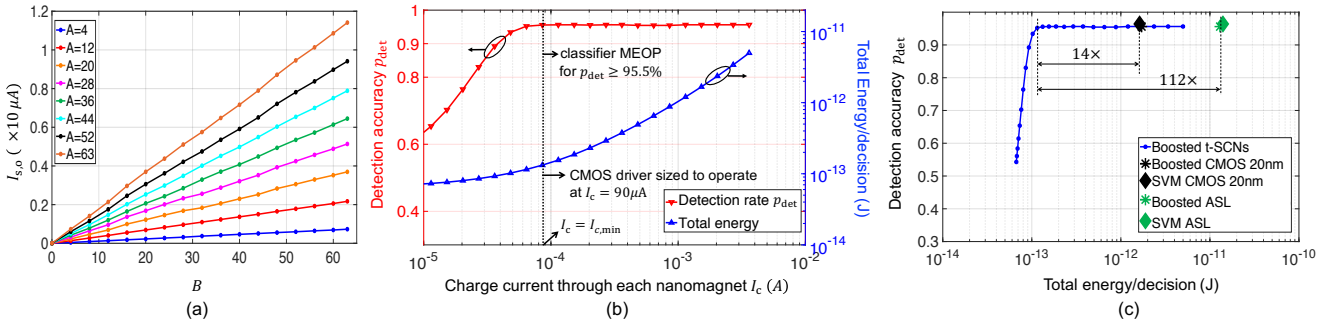

Fig. 6. Simulation results: (a) Transfer function of a 6×6 bit SCNM having A and B as its two operands, (b) detection accuracy p_{det} and total energy vs. I_c tradeoff for boosted t-SCN based breast cancer detector operating at a final decision delay of 3 ns. The minimum energy operating point (MEOP) is achieved at $I_{c,\text{min}} = 90\,\mu$A, and (c) detection accuracy p_{det} vs. energy tradeoff for different breast cancer detector implementations operating at a final decision delay of 3 ns.

(weak) decision $\hat{y}_i$. We restrict the number of weak classifiers to $\frac{N}{2}$ so that computational complexity of the boosted t-SCNs architecture is similar to the standard N dimensional linear SVM implementation (Fig. 4(e)). It is straightforward to convert the binary slicer decisions $\hat{y}_i$ $i \in \{1, \ldots, \frac{N}{2}\}$ to an equivalent voltage [13] and then route it using charge interconnects. Final weighted logic block (designed in conventional digital 14 nm CMOS) combines these weak decisions to obtain the final decision $\hat{y}_f$ as per (2). Its complexity in terms of the full adder count is less than 5% of the total complexity of the $\frac{N}{2}$ t-SCNs.

V. Simulation Results

We design and characterize a 6×6 bit SCNM using the SPICE-based spin device models [23] with the material parameters from [27]. The channel lengths between the clusters in the SCNM layouts are repeatedly adjusted until the appropriate spin current weighting is achieved and all λ-rules are satisfied. Fig. 6(a) shows the simulated SCNM transfer function observed for different A and B values. The observed $(\frac{\sigma}{\mu})$ of the deviations from linearity in output spin current is $\approx 2\%$. We use this transfer function for our system-level simulations in MATLAB. All the peripheral CMOS circuits, such as CMOS drivers for each nanomagnet in SCNM, weighted logic block in boosted t-SCNs, are designed and simulated using 14 nm HP FinFET ASU predictive technology models [29] to estimate their energy consumption and delay.

We estimate the dynamic energy consumptions of 20 nm LV CMOS [27] digital and ASL (consisting of nanomagnets with improved anisotropy in [27]) implementations using the benchmarking methodology in [25]. We assume activity factor of 33% for digital CMOS. We also consider ASL gates to be clocked using a MOSFET [5], [22] (see Fig. 2).

We demonstrate the effectiveness of proposed approach for 10-dimensional breast cancer detection (UCI repository dataset [30], [31]). We compare energy consumption and accuracy of all classifier implementations at fixed final decision delay of 3 ns. We quantify classification accuracy in terms of detection accuracy $p_{\text{det}} = 1 - p_e$, where p_e is classification error probability. For each t-SCN in the boosted architecture, I_{th} controls its weak decision delay T_g, while I_c independently controls its switching accuracy and energy. For a fixed final decision delay (of 3 ns), the trade-off between the accuracy and total energy consumption of the 10D boosted t-SCN classifier is shown in Fig. 6(b) as a function of I_c. As expected, both accuracy and total energy decrease with I_c. For accuracy of 95.5%, the classifier MEOP is achieved at $I_{c,\text{min}} = 90\,\mu$A.

Figure 6(c) shows the accuracy vs energy tradeoff for different breast cancer detector implementations. Boosted t-SCN detector achieves at least 112× lower energy per decision compared to that of the conventional boosted ASL implementation while maintaining accuracy as observed in Fig. 6(c). Such large energy savings can be attributed to the elimination of all intermediate switching nanomagnets in the spin channel network implementation. It also achieves 14× lower energy compared to boosted 20 nm LV CMOS digital implementation, while operating at the identical final decision delay. We also observe that both boosted CMOS and boosted ASL implementations achieve energy consumption close to the respective N-dimensional linear SVM implementations due to their similar computational complexity.

VI. Conclusion

In this paper, we propose spin channel networks, which exploit the exponential decay of spin current to efficiently compute multi-bit dot products. While they are composed of same building blocks as that of the ASL, they eliminate need for intermediate nanomagnetic switching, thus achieving significant energy-efficiency over both ASL and CMOS. As a part of the future work, the proposed approach can be extended to efficiently implement multi-layer deep neural networks, which mainly consist of multi-bit dot products followed by a non-linearity.

Acknowledgment

This work was supported in part by C-BRIC, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA, and SONIC, one of the six SRC STARnet Centers sponsored by MARCO and DARPA.

REFERENCES

[1] D. E. Nikonov, G. I. Bourianoff, and T. Ghani, "Proposal of a spin torque majority gate logic," *IEEE Electron Device Letters*, vol. 32, no. 8, pp. 1128–1130, 2011.

[2] S. Manipatruni, D. E. Nikonov, R. Ramesh, H. Li, and I. A. Young, "Spin-orbit logic with magnetoelectric nodes: A scalable charge mediated nonvolatile spintronic logic," *arXiv preprint arXiv:1512.05428*, 2015.

[3] B. Behin-Aein, D. Datta, S. Salahuddin, and S. Datta, "Proposal for an all-spin logic device with built-in memory," *Nature nanotechnology*, vol. 5, no. 4, p. 266, 2010.

[4] D. Nikonov and I. Young, "Benchmarking of beyond-CMOS exploratory devices for logic integrated circuits," *Exploratory Solid-State Computational Devices and Circuits, IEEE Journal on*, 2015.

[5] Z. Pajouhi, S. Venkataramani, K. Yogendra, A. Raghunathan, and K. Roy, "Exploring spin-transfer-torque devices for logic applications," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 9, pp. 1441–1454, 2015.

[6] J. Kim, A. Paul, P. A. Crowell, S. J. Koester, S. S. Sapatnekar, J.-P. Wang, and C. H. Kim, "Spin-based computing: Device concepts, current status, and a case study on a high-performance microprocessor," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 106–130, 2015.

[7] M. Sharad, C. Augustine, G. Panagopoulos, and K. Roy, "Spin-based neuron model with domain-wall magnets as synapse," *IEEE Transactions on Nanotechnology*, vol. 11, no. 4, pp. 843–853, 2012.

[8] C. Pan and A. Naeemi, "A proposal for energy-efficient cellular neural network based on spintronic devices," *IEEE Transactions on Nanotechnology*, vol. 15, no. 5, pp. 820–827, 2016.

[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*. ACM, 2014, pp. 15–20.

[10] A. Sengupta and K. Roy, "Encoding neural and synaptic functionalities in electron spin: A pathway to efficient neuromorphic computing," *Applied Physics Reviews*, vol. 4, no. 4, p. 041105, 2017.

[11] H. Farkhani, M. Tohidi, S. Farkhani, J. K. Madsen, and F. Moradi, "A low-power high-speed spintronics-based neuromorphic computing system using real time tracking method," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 2018.

[12] B. Sutton, K. Y. Camsari, B. Behin-Aein, and S. Datta, "Intrinsic optimization using stochastic nanomagnets," *Scientific Reports*, vol. 7, p. 44370, 2017.

[13] K. Y. Camsari, R. Faria, B. M. Sutton, and S. Datta, "Stochastic p-bits for invertible logic," *Physical Review X*, vol. 7, no. 3, p. 031014, 2017.

[14] R. Venkatesan, S. Venkataramani, X. Fong, K. Roy, and A. Raghunathan, "Spintastic: Spin-based stochastic logic for energy-efficient computing," in *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*. EDA Consortium, 2015, pp. 1575–1578.

[15] S. et al., "Probabilistic deep spiking neural systems enabled by magnetic tunnel junction," *Electron Devices, IEEE Tran. on*, 2016.

[16] I. Chakraborty, A. Agrawal, and K. Roy, "Proposal for a low voltage analog-to-digital converter using voltage controlled stochastic switching of low barrier nanomagnets," *arXiv preprint arXiv:1803.01431*, 2018.

[17] S. Parkin and S.-H. Yang, "Memory on the racetrack," *Nature nano.*, vol. 10, no. 3, pp. 195–198, 2015.

[18] Z. Sun, X. Bi, A. K. Jones, and H. Li, "Design exploration of racetrack lower-level caches," in *Low Power Electronics and Design (ISLPED), 2014 IEEE/ACM International Symposium on*. IEEE, 2014, pp. 263–266.

[19] J. Chung, J. Park, and S. Ghosh, "Domain wall memory based convolutional neural networks for bit-width extendability and energy-efficiency," in *Proceedings of the 2016 International Symposium on Low Power Electronics and Design*. ACM, 2016, pp. 332–337.

[20] Y. Wang, H. Yu, L. Ni, G.-B. Huang, M. Yan, C. Weng, W. Yang, and J. Zhao, "An energy-efficient nonvolatile in-memory computing architecture for extreme learning machine by domain-wall nanowire devices," *IEEE Transactions on Nanotechnology*, vol. 14, no. 6, pp. 998–1012, 2015.

[21] Q. Dong, K. Yang, L. Fick, D. Fick, D. Blaauw, and D. Sylvester, "Low-power and compact analog-to-digital converter using spintronic racetrack memory devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 3, pp. 907–918, 2017.

[22] V. Calayir, D. E. Nikonov, S. Manipatruni, and I. A. Young, "Static and clocked spintronic circuit design and simulation with performance analysis relative to cmos," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 2, pp. 393–406, 2014.

[23] P. Bonhomme, S. Manipatruni, R. M. Iraei, S. Rakheja, S.-C. Chang, D. E. Nikonov, I. A. Young, and A. Naeemi, "Circuit simulation of magnetization dynamics and spin transport," *IEEE Transactions on Electron Devices*, vol. 61, no. 5, pp. 1553–1560, 2014.

[24] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[25] D. E. Nikonov and I. A. Young, "Overview of beyond-cmos devices and a uniform methodology for their benchmarking," *Proceedings of the IEEE*, vol. 101, no. 12, pp. 2498–2533, 2013.

[26] R. E. Schapire, "Explaining adaboost," in *Empirical inference*. Springer, 2013, pp. 37–52.

[27] S. Manipatruni, D. E. Nikonov, and I. A. Young, "Material targets for scaling all-spin logic," *Physical Review Applied*, vol. 5, no. 1, p. 014002, 2016.

[28] W. H. Butler, T. Mewes, C. K. Mewes, P. Visscher, W. H. Rippard, S. E. Russek, and R. Heindl, "Switching distributions for perpendicular spin-torque devices within the macrospin approximation," *IEEE Transactions on Magnetics*, vol. 48, no. 12, pp. 4684–4700, 2012.

[29] S. Sinha, G. Yeric, V. Chandra, B. Cline, and Y. Cao, "Exploring sub-20nm finfet design with predictive technology models," in *Proceedings of the 49th Annual Design Automation Conference*. ACM, 2012, pp. 283–288.

[30] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[31] W. H. Wolberg and O. L. Mangasarian, "Multisurface method of pattern separation for medical diagnosis applied to breast cytology." *Proceedings of the national academy of sciences*, 1990.