2023 IEEE International Conference on Rebooting Computing (ICRC) | 979-8-3503-8204-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICRC60800.2023.10386729

Dendrite-inspired Computing to Improve Resilience of Neural Networks to Faults in Emerging Memory Technologies

Lizy K. John[1], Felipe M. G. França[2], Subhasish Mitra[3], Zachary Susskind[1], Priscila M. V. Lima[4], Igor D. S. Miranda[5], Eugene B. John[6], Diego L. C. Dutra[4], and Mauricio Breternitz Jr.[7],

1- UT Austin, USA, 2- Instituto de Telecomunicações, Portugal, 3- Stanford University, USA, 4- UFRJ, Brazil, 5- UFRB, Brazil, 6- UT San Antonio, USA, 7- ISCTE Instituto Universitario de Lisboa, Portugal

***Abstract*—Mimicking biological neurons by focusing on the excitatory/inhibitory decoding performed by dendritic trees offers an intriguing alternative to the traditional integrate-and-fire McCullogh-Pitts neuron stylization. Weightless Neural Networks (WNN), which rely on value lookups from tables, emulate the integration process in dendrites and have demonstrated notable advantages in terms of energy efficiency. In this paper, we delve into the WNN paradigm from the perspective of reliability and fault tolerance. Through a series of fault injection experiments, we illustrate that WNNs exhibit remarkable resilience to both transient (soft) errors and permanent faults. Notably, WNN models experience minimal deterioration in accuracy even when subjected to fault rates of up to 5%. This resilience makes them well-suited for implementation in emerging memory technologies for binary or multiple bits-per-cell storage with reduced reliance on memory block-level error resilience features. By offering a novel perspective on neural network modeling and highlighting the robustness of WNNs, this research contributes to the broader understanding of fault tolerance in neural networks, particularly in the context of emerging memory technologies.**

I. Introduction

The majority of current machine learning models today are deep neural networks based on simple weighted-sum-and threshold artificial neurons, as variants of the pioneering Threshold Logic Unit by McCullogh and Pitts [1]. The biological analogy behind this model lies on the mapping of the synaptic strength between the output produced by one neuron's axon and the input of a post-synaptic neuron into pseudo-continuous numerical weights. An important simplification happens in the way inputs to neurons are modeled: all synaptic connections terminate directly at the neuron's soma. Although such specific morphological arrangement is plausible in biological terms, the vast majority of synapses in the central nervous system terminate at the neuron's dendritic tree [2].

By simulating a state of the art detailed biophysical model of a single cortical neuron (that attempts to capture all biological details that are currently known about the inner workings of biological neurons), and trying to find the smallest deep neural network, Beniaguev et al. [3] showed that a deep neural network of 5-8 layers is necessary to faithfully capture a detailed model of a single L5 cortical pyramidal neuron (See Figure 1). Seven hidden layers consisting of 128 feature maps per layer and a history of 153 ms is necessary for capturing both AMPA (α-Amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid) and NMDA (n-methyl-d-aspartic acid) responses whereas if only AMPA synapses are modeled, it is faithfully captured by a DNN with one hidden layer. The minimal DNN size required to achieve a good fit is larger for AMPA and NMDA synapses compared to AMPA-only synapses across all tested hyperparameters. This research adds evidence to the notion that traditional neuron model may not be the most appropriate one to describe brain's computing.

The dendritic tree, a highly noticeable morphological structure of the neuron cell, is not being taken into account in mainstream neural network paradigms. As widely understood, a nerve cell consists of dendrites, soma, axons and synapses. Each nerve cell has one or more dendrites, which receive the stimulus and transmit them to soma. It was generally thought for a long time that dendrites simply passively transmit electrical impulses received from synapses to soma. However, recently there has been research to suggest that dendrites are not just passive channels [3, 5, 6, 7]. According to recent research, dendritic branches can be conceptualized as a set of spatio-temporal pattern detectors [3] (Figure 2). Dendrites process inhibitory and excitatory action which are marked *i* and *e* in the Figure 2 and dendritic integration [4] can be modeled into hardware tables. There is increased support to the notion that understanding dendritic activity and utilizing principles learned from that into computer architecture may be necessary to improve the energy-efficiency of neural network hardware.

Mimicking biological neurons by focusing on the excitatory/inhibitory decoding performed by the dendritic trees as in Figure 2 is an attractive alternative to the integrate-and-fire McCullogh-Pitts neuron stylisation. In such alternative analogy, neurons can be seen as a set of memory nodes addressed by Boolean inputs and producing Boolean outputs. This is what a class of neural networks called Weightless Neural Networks (WNNs) do [8, 9, 10]. Their operation has similarities to the integration of excitatory and inhibitory signaling performed by the neuron's dendritic tree.

The objective of this paper is to illustrate the advantage of

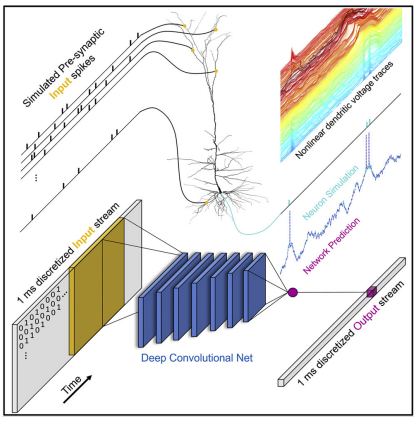


Fig. 1: *A 7-layer DNN is necessary to model an L5 cortical pyramidal neuron with AMPA and NMDA synapses [3]*

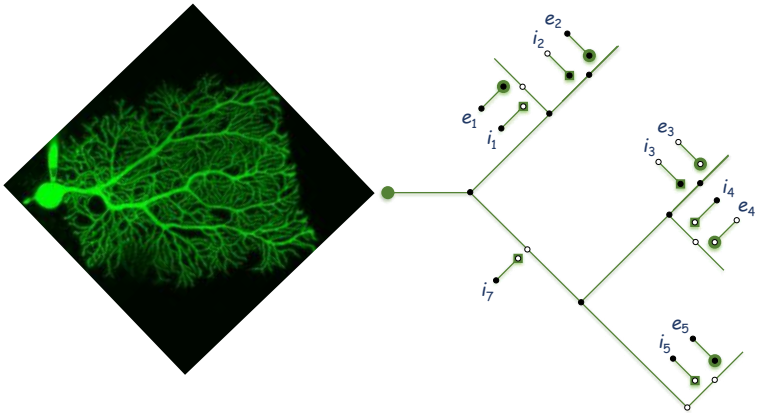


Fig. 2: *The dendritic tree (Left). Dendrite inhibitory and excitatory action marked i and e in the figure on the Right and dendritic integration [4] can be modeled into hardware tables*

WNNs from the point of view of reliability and fault tolerance, and their suitability for implementation in emerging memory technologies that may be more susceptible to errors (compared to traditional memories) for various reasons such as technology maturity or multiple bits-per-cell storage.

II. Background and Motivation

Weightless Neural Networks (WNNs) rely on value lookups implemented using RAMs or look up tables (LUTs) instead of Multiply-Accumulate (MAC) operations [8, 9, 10]. The weightless neural network architecture has shown success as pattern detectors. This neuromorphic architecture modeled after synaptic integration in dendrites has serious potential in creating energy-efficient machine learning systems. However, naive implementations of WNNs need tremendous amounts of memory, and individual RAM nodes can not generalize.

WiSARD [8] is one of the most popular and successful WNNs. It is intended for classification tasks and avoids the state explosion problem. Lookup tables can capture a variety of non-linear functions, and neural networks built with them have a great ability to learn patterns with few parameters. WiSARD has also been shown to have a large VC dimension [11]. WiSARD is best suited to classifier tasks, where inputs are partitioned into different categories. WiSARD uses a sub-model called *discriminator* for each class. A discriminator is created for each output category; these discriminators in turn are composed of many small RAM nodes, as illustrated in Figure 3. During inference, outputs of the RAM nodes in each discriminator are summed, and index of the discriminator with the strongest response is the prediction. WiSARD avoids the state explosion problem for simple WNNs, but it is impractical for large models.

The energy consumed by modern deep neural networks (DNNs) is orders of magnitude higher than equivalent biological neural activity. Recent research [13, 14, 12] has demonstrated that WNNs and their variations are effective for energy-critical edge applications.

There are also technology trends that may be opportunistic

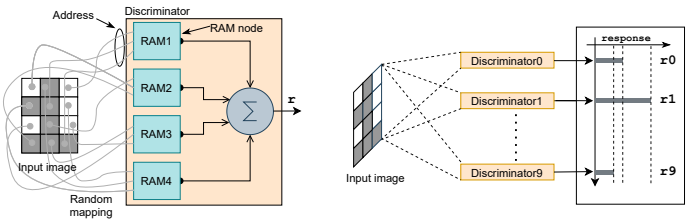
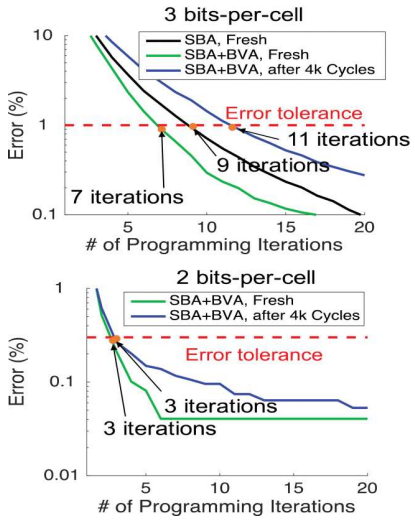


Fig. 3: *A popular weightless model, the WiSARD, doing digit recognition. The input image has a 1 and the discriminator corresponding to digit 1 has the highest response here [12].*

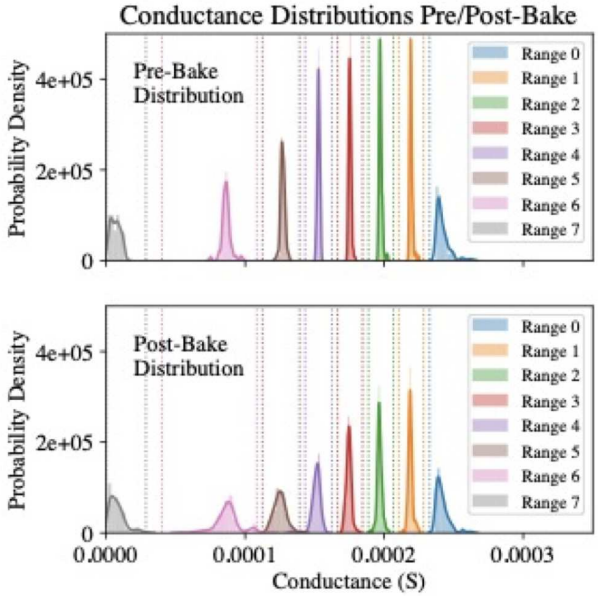
for WNNs now. We are finding some emerging memory technologies which can support high densities. However, certain features of such technologies, e.g., multiple bits-per-cell storage, might exhibit reliability challenges that may have to be coped architecturally. Such scenarios trade off capacity for reliability - an increasing number of bits can be stored in a cell in exchange for a higher error rate. Applications need very high fault tolerance in order to use these technologies at the maximum level of density.

III. Fault-tolerant WNNs for the edge

Consider the Resistive RAM (RRAM) technology used in the Chimera work [15, 16]. RRAM arrays are high-density thanks to the 1-transistor structure (vs. 6-transistor SRAM cells). Even though RRAM controller and peripheral circuits are complex, RRAM arrays provide a $> 2\times$ density increase over SRAM arrays (at the same technology node). Bit density can be further improved (2-4$\times$) by leveraging the wide resistance range of RRAM (5kOhm to 100kOhm) for multiple bits-per-cell storage [17, 18]. Moreover, RRAM is compatible with high-density 3D structures due to its back-end-of-line (BEOL) compatible fabrication temperature, leading to further density improvement via vertical integration [17, 19]. RRAM read energy (8 pJ/byte, measured) is several times lower compared to DRAM. RRAM is thus non-volatile, ultra-dense, and read-energy efficient. However, RRAM writes require high voltages resulting in a high write energy (40.3 nJ/byte, measured) and repeated pulses leading to extra write latency (421 ns/byte, measured). The limited write endurance (10k cycles [20]) is a major challenge for DNN training, however WNN accelerators



(a) *RRAM can experience programming bit errors, especially for multiple-bits-per-cell storage. Additional programming pules (e.g., iterations) can help reduce these errors at the cost of additional write latency and energy.*



(b) *Programmed resistance distributions can shift over time, leading to read errors. Here shown is a distribution of RRAM cells before baking (top) and after 30-min bake at 130C (bottom). BER after baking is 0.6% (before ECC). Dashed lines indicate read range boundaries.*

Fig. 4: *RRAM Reliability (a) Percent error (b) Resistance Distribution*

will contain very few writes due to the inherent one-shot or few-shot nature of WNN training. When used for edge inference, the write-endurance challenge can be easily handled since lookup tables are only read and not written into. In addition, prior work demonstrates that RRAM achieves fine-grained temporal power gating with up to 5,878X quicker transition from active to shutdown mode (measured) vs. on-chip Flash [21], leading to further suitability for energy-constrained edge devices which can save energy by going off, but can come to action for fast response times. The features of RRAMs are suitable for on-chip non-volatile memory with low-energy, and we expect these attributes to be ideal for WNNs as well.

RRAM arrays where each cell can store 3 bits [16] have also been demonstrated. Such full array-level demonstration was possible through special techniques (e.g., that exploit RRAM-specific characteristics of variations in cell resistances), which efficiently allocated resistance range corresponding to each bit combination (required for proper write operation) while maintaining appropriate sensing margin (required for proper read operation). These techniques are not restricted to 3 bits-per-cell only (and, in the paper [16], but 2 bits-per-cell at the array level was demonstrated as well). The measured results are based on multiple 4 Kbit arrays of 1T1R HfOx-based RRAM integrated in the back end of the line of 130-nm silicon CMOS technology and yield 3 bits-per-cell (2 bits-per-cell) RRAM with 11 (3) programming iterations on average.

While binary storage in RRAM is well-established (through hardware prototypes, RRAM macros from foundries such as TSMC, and announcements about the use of RRAM for automotive applications) [22], there are several challenges in integrating multiple bits-per-cell capabilities to systems. RRAM cell-to-cell variations mean that cell programming can yield resistances which are not reliably within the desired range for multiple bits-per-cell storage. Bit error rates can thus vary greatly with the number of programming attempts made on the cell (see Figure 4a). Bit retention can also become increasingly difficult as more bits are programmed into the cell. Accelerated retention studies such as those shown in Figure 4b are encouraging for system performance and programming robustness in face of these potential errors. WNNs have features that can potentially mask these challenges. Since WNN's output is chosen by looking at the different discriminators and choosing the one with the highest output, some bit errors may not catastrophically affect the inference.

IV. Preliminary Results

We explored the resilience of WNNs by injecting soft and permanent bit flips into the RAM nodes of pretrained WNNs. Our preliminary investigation shows that WiSARD-based WNNs exhibit a high degree of resilience to both transient (soft) errors and permanent errors. Figure 5 demonstrates that WNNs can retain useful accuracy with error rates as high as 20%. With error rates such as 5%, the accuracy deterioration is negligible. By contrast, recent studies show that a *single* bit error can cause substantial (>4%) loss of accuracy in 5.9% of cases in DNN training [23]. The source of this robustness can be traced back to the behavior of the WiSARD model itself. To restate from earlier, in WiSARD, each discriminator, or single-class predictor, produces an "activation" score by summing the outputs of its component RAMs, and the prediction of the model is the class corresponding to the discriminator with the strongest activation. Thus, the actual values of the activations do not impact the prediction, only their *relative* values. If the output of one discriminator a is larger than that of another discriminator b, then $a > b$ will still *usually* remain true so long as the error rate of the RAMs is less than 0.5. The potential for fault-tolerant WNNs considering the reliability attributes of RRAM needs to be further studied.

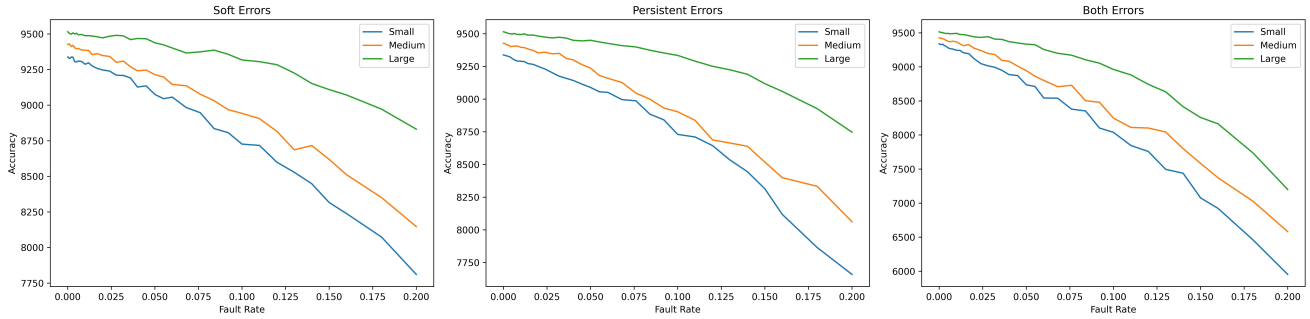


Fig. 5: *A demonstration of the robustness of WiSARD models to faults in memories. Three models were trained using the WiSARD dataset (Small: 70 KiB model size; Medium: 210 KiB; Large: 960 KiB); then, soft and persistent errors were artificially injected during inference at rates up to 20%. All models showed excellent resistance to faults up to a 5% fault rate. The larger models, which had greater redundancy, were less affected by higher fault rates.*

The source of this robustness can be traced back to the behavior of the WiSARD model itself. To restate from earlier, in WiSARD, each discriminator, or single-class predictor, produces an "activation" score by summing the outputs of its component RAMs, and the prediction of the model is the class corresponding to the discriminator with the strongest activation. Thus, the actual values of the activations do not impact the prediction, only their *relative* values. If the output of one discriminator a is larger than that of another discriminator b, then $a > b$ will still *usually* remain true so long as the error rate of the RAMs is not very high.

V. Future Outlook

There are additional reasons why WNNs can be even more resilient. They can use **Ensembles**, like the one shown in Figure 6, combining multiple weak classifiers into a single strong classifier to improve the accuracy of WNNs [12]. Ensembles have been extensively studied in other areas of machine learning, and are the driving concept behind techniques such as Bayesian averaging, boosting, and bagging [24]. In recent work [12] ensembles are trained by independently training several WNN submodels on the same training data. The response scores for each discriminator across the submodels are then summed up before performing the final prediction. In other words, if a submodel i produces response score $R_{i,j}$ for class j, then the final response score for this class will be $\sum_i R_{i,j}$. This ensemble technique is similar to but distinct from bagging. In bagging, submodels are trained using random subsets of the training data, with the objective of influencing them to learn different patterns and behaviors. On the other hand, in ensemble WNNs [12], all submodels see the same training data, but the connections from model inputs to RAM nodes are different. This sparse connectivity forces RAM nodes in different submodels to capture information distinctly.

One might reasonably expect that using ensembles of submodels would increase the size of a model, since there are more RAM nodes in total. However, prior work [12] found that in practice this is frequently not the case. The individual submodels of an ensemble can be made much smaller (and therefore individually less accurate) than a monolithic model without significantly degrading ensemble accuracy. It was seen that

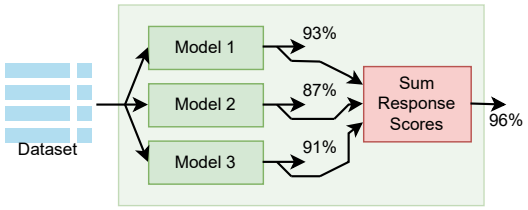


Fig. 6: *Simplified view of an ensemble model. Summing the response scores of independently-trained submodels (Model 1, Model 2, etc.) is more accurate than any of the individual submodels. From* [12].

ensembles can increase MNIST accuracy to 98.5%. Ensembles have the potential to further improve the fault-tolerance of WNNs due to inherent redundancies. Fault-tolerance of ensemble-based WNNs will be studied in the future utilizing insights from prior works on DNN reliability [25].

VI. Conclusion

In this paper, we conducted an analysis of the fault tolerance of weightless neural networks through a series of fault injection experiments. Additionally, we examined the advantages and limitations of emerging memory technologies. Our findings reveal a compelling story: WNN models exhibit exceptional resilience, experiencing minimal accuracy degradation even when exposed to fault rates of up to 5% in the RAM that holds the lookup tables. A detailed comparison of WNN resilience to the resilience of DNNs is interesting future work.

The inherent fault-tolerant nature of WNNs positions them as an ideal architecture to harness the benefits of emerging memory technologies. This work underscores the profound significance of fault tolerance in the context of neural networks and emerging memory technologies. Looking ahead, the synergy between WNNs and emerging memory technologies opens up exciting possibilities for more robust and efficient computing systems. Future research may explore the practical implementations and applications of this symbiotic relationship, pushing the boundaries of both neural network design and memory technology advancement.

Acknowledgement: This research was supported in part by National Science Foundation (NSF) Grant #2326894, #2326895, CAPES and CNPq, Brazil, by Next Generation EU, PRR Program, Project Route 25 Grant #C645463824-

00000063, and ISTAR Projects UIDB/04466/2020, UIDP/04466/2020 and DSAIPA/AI/0122/2020 Aim Health Portugal, through national funds and when applicable co-funded EU funds under the project UIDB/50008/2020. Any opinions, findings, conclusions or recommendations are those of the authors and not of the funding agencies.

REFERENCES

[1] Warren S. McCulloch and Walter Pitts. "A Logical Calculus of the Ideas Immanent in Nervous Activity". In: *The Bulletin of Mathematical Biophysics* 5.4 (1943), pp. 115–133. DOI: 10.1007/bf02478259.

[2] D. Johnston. "Foreword, in: Dendrites, G. Stuart, N. Spruston and M. Häusser, Eds, Oxford Univ. Press". In: (1999).

[3] David Beniaguev, Idan Segev, and Micahel London. "Single Cortical Neurons as Deep artificial Neuron Networks". In: 2021. URL: https://www.sciencedirect.com/science/article/abs/pii/S0896627321005018.

[4] C. Koch and T. Poggio. "Biophysics of Computation: Neurons, Synapses and Membranes, in: Synaptic Function, G. M. Edelman, W. E. Gall and W.M. Cowan, John Wiley & Sons, 1987". In: (1987).

[5] A.J. Moore et al. "Dynamics of cortical dendritic membrane potential and spikes in freely behaving rats". In: *Science*. 2017.

[6] Jyotibdha Acharya et al. "Dendritic Computing: Branching Deeper into Machine Learning". In: *Neuroscience* 489 (2022). Dendritic contributions to biological and artificial computations, pp. 275–289. ISSN: 0306-4522. DOI: https://doi.org/10.1016/j.neuroscience.2021.10.001.

[7] Hong Peng et al. "Dendrite P systems". In: *Neural Networks* 127 (2020), pp. 110–120. ISSN: 0893-6080. DOI: https://doi.org/10.1016/j.neunet.2020.04.014.

[8] I. Aleksander, W.V. Thomas, and P.A. Bowden. "WISARD·a radical step forward in image recognition". In: *Sensor Review* 4.3 (1984), pp. 120–124. ISSN: 0260-2288. DOI: 10.1108/eb007637.

[9] Alan F. Murray et al. "Analogue and Digital Neural VLSI: Duet or Duel?" In: *International Symposium on Circuits and Systems, ISCAS*. IEEE, 1994, pp. 285–288.

[10] Igor Aleksander et al. "A brief introduction to Weightless Neural Systems". In: *17th European Symp on Artificial Neural Networks (ESANN)*. 2009, pp. 299–305.

[11] Hugo Carneiro et al. "The exact VC dimension of the WiSARD n-tuple classifier". In: *Neural Computation* (Nov. 2018), pp. 1–32. DOI: 10.1162/neco_a_01149.

[12] Zachary Susskind et al. "ULEEN: A Novel Architecture for Ultra Low-Energy Edge Neural Networks". In: *ACM Trans. Archit. Code Optim.* (Oct. 2023). Just Accepted. ISSN: 1544-3566. DOI: 10.1145/3629522.

[13] Leandro Santiago et al. "Weightless neural networks as memory segmented bloom filters". In: *Neurocomputing* 416 (2020), pp. 292–304.

[14] Zachary Susskind et al. "Weightless Neural Networks for Efficient Edge Inference". In: *31st International Conference on Parallel Architectures and Compilation Techniques (PACT)*. 2022. DOI: https://doi.org/10.1145/3559009.3569680.

[15] Massimo Giordano et al. "CHIMERA: A 0.92 TOPS, 2.2 TOPS/W Edge AI Accelerator with 2 MByte On-Chip Foundry Resistive RAM for Efficient Training and Inference". In: *Symp. on VLSI Circuits*. 2021, pp. 1–2. DOI: 10.23919/VLSICircuits52068.2021.9492347.

[16] Binh Q. Le et al. "Resistive RAM With Multiple Bits Per Cell: Array-Level Demonstration of 3 Bits Per Cell". In: *IEEE Transactions on Electron Devices* 66.1 (2019), pp. 641–646. DOI: 10.1109/TED.2018.2879788.

[17] E. R. Hsieh et al. "High-Density Multiple Bits-per-Cell 1T4R RRAM Array with Gradual SET/RESET and its Effectiveness for Deep Learning". In: *2019 IEEE Intl. Electron Devices Meeting*. 2019, pp. 35.6.1–35.6.4. DOI: 10.1109/IEDM19573.2019.8993514.

[18] E. R. Hsieh et al. "Four-Bits-Per-Memory One-Transistor-and-Eight-Resistive-Random-Access-Memory (1T8R) Array". In: *IEEE Electron Device Letters* 42.3 (2021), pp. 335–338. DOI: 10.1109/LED.2021.3055017.

[19] Max M. Shulaker et al. "Three-dimensional integration of nanotechnologies for computing and data storage on a single chip". In: *Nature* 547.7661 (July 2017), pp. 74–78. ISSN: 1476-4687. DOI: 10.1038/nature22994.

[20] M. M. Shulaker et al. "Monolithic 3D integration of logic and memory: Carbon nanotube FETs, resistive RAM, and silicon FETs". In: *2014 IEEE International Electron Devices Meeting*. 2014, pp. 27.4.1–27.4.4. DOI: 10.1109/IEDM.2014.7047120.

[21] T. F. Wu et al. "14.3 A 43pJ/Cycle Non-Volatile Microcontroller with 4.7μs Shutdown/Wake-up Integrating 2.3-bit/Cell Resistive RAM and Resilience Techniques". In: *2019 IEEE International Solid- State Circuits Conference - (ISSCC)*. 2019, pp. 226–228. DOI: 10.1109/ISSCC.2019.8662402.

[22] *Infineon and TSMC to introduce RRAM technology for Automative*. 2022. URL: https://www.infineon.com/cms/en/about-infineon/press/market-news/2022/INFATV202211-031.html.

[23] Yi He and Yanjing Li. "Understanding Permanent Hardware Failures in Deep Learning Training Accelerator Systems". In: *IEEE European Test Symp.* 2023, pp. 1–6.

[24] Thomas G. Dietterich. "Ensemble Methods in Machine Learning". In: *Multiple Classifier Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 1–15. ISBN: 978-3-540-45014-6.

[25] Brunno F. Goldstein et al. "A Lightweight Error-Resiliency Mechanism for Deep Neural Networks". In: *22nd International Symposium on Quality Electronic Design (ISQED)*. 2021, pp. 311–316.