NEUROMORPHIC COMPUTING AND ITS APPLICATIONS: A SURVEY

NAME: DAVID ADESHINA ARUNGBEMI

ID. NO.: 191203012

DEPT: COMPUTER ENGINEERING

FACULTY: FACULTY OF ENGINEERING

LEVEL: 500 LVL

SUBMITTED TO:

PROF. STEVE ADESHINA

NILE UNIVERSITY OF NIGERIA

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE COURSE:

MICROCOMPUTER HARDWARE AND SOFTWARE TECHNIQUES

SUBMISSION DATE:

15TH JANUARY, 2024

Neuromorphic Computing and its Applications: A Survey

David A. Arungbemi
Nile University of Nigeria, Department of Computer Engineering
Abuja, Nigeria

***Abstract*—Beyond Von Neumann computing refers to approaches deviating from the traditional Von Neumann computing paradigm. These include its successor, the Harvard architecture and other paradigms such as quantum computing and neuromorphic computing. The neuromorphic computing approach is inspired by the human brain. It aims to mimic brain processes or behaviours in an attempt to achieve computing breakthroughs that traditional computers today cannot achieve. This paper introduces the concept of neuromorphic computing including the different fields involved, fields of research, and motivation; the factors that influence its developments including models developed, algorithms, hardware and present and potential future applications in different fields especially in the area of machine learning.**

***Index Terms*—computing architecture, neuromorphic computing, neural networks**

I. Introduction

This paper provides a comprehensive survey of neuromorphic computing, beginning with a comparison with traditional architecture, fields of research and motivating factors. It discusses the areas of research and development including models, algorithms and hardware. Additionally, the paper analyses the range of applications (both present and future) such as robotics, control and machine learning. The goal of this paper is to provide a good general understanding of what neuromorphic computing entails and to serve as a starting point for those new to the field.

A. Traditional Computing and Neuromorphic Computing

Traditional computers are based on modified versions of the Von Neumann architecture. The Von Neumann architecture is distinguished by its stored-program concept, leading to the separation of memory and CPU which is often referred to as the Von Neumann/memory bottleneck(low bandwidth between CPU and memory) [1]. For example, when the CPU requests data faster than the memory can provide it can lead to latency(slower processing). Introducing techniques such as separated memory(Harvard architecture) [2], and caching [3] help to alleviate the problem but even these still suffer from some bottlenecks [1], [4]. Hence alternative architecture is required, brain-like computing.

Neuromorphic computing was first proposed in the 1980s by Carver Mead [5] and takes inspiration from the human brain due to its low power and processing capabilities. Neuromorphic computing refers to applying biologically plausible, biologically inspired or artificial neural network models to developing non-Von Neumann architectures [6]. In addition to the Von Neumann bottleneck, other factors for the growing interest in neuromorphic computing include the imminent plateauing of Moore's law and the need to reduce power consumption with the end Dennard scaling [7], as shown in Figure 1.

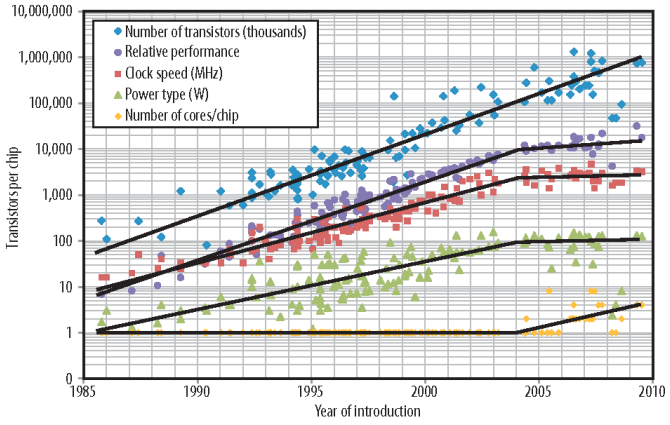

Fig. 1. Evolution of transistors, performance, clock speed, power, and core trends over time in relation to the increasing number of transistors, reflecting the trajectories of Moore's Law and Dennard Scaling. Adapted from [8].

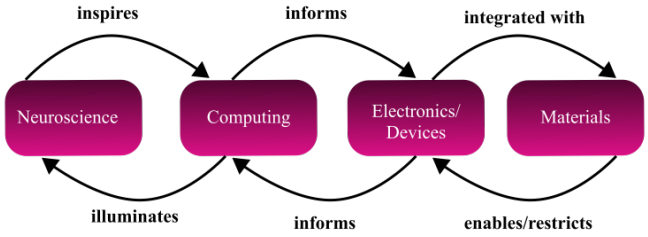

Fig. 2. Fields of research in neuromorphic computing. Adapted from [6] with modifications for better clarity.

B. Neuromorphic Computing Community

Neuromorphic computing is an interdisciplinary field involving researchers in the fields of materials science, neuroscience, electronics engineering, computer engineering, and computer science [6]. Figure 2 illustrates the interrelationship between the different fields.

Material science, an interdisciplinary field, is dedicated to the study of the properties of solid materials and how the material's composition and structure influence these properties

[9]. Materials can be selected or designed for a variety of applications such as structural steels and computer chips, among many others [9]. In neuromorphic computing, material scientists study and create new materials that demonstrate characteristics similar to biological systems [6].

Neuroscience is the study of the nervous system's structure, function, development and degeneration in states of health and disease, with the primary focus on the brain [10]. In neuromorphic computing, neuroscientists provide new results in their study for possible computational applications and also utilise neuromorphic machines to simulate and study the brain [6].

Electronics/computer engineering are both involved in the design, construction, implementation and maintenance of hardware/circuits [11], [12]. They work with analogue, digital or a combination of both to develop new devices, new systems and communication methods in neuromorphic computing.

Computer scientists including computer engineers are involved in the study of computers and computing, algorithms, modelling data, information processes and software/hardware [11], [13]. They help to develop biologically inspired/plausible or artificial neural network-inspired models, algorithms, software and supporting systems for neuromorphic systems [6].

C. Motivation

The motivations for neuromorphic computing have evolved significantly since earlier research endeavours. However, one of the popular reasons for neuromorphic implementation in early works was parallelism on a single chip, inspired by the parallel operations of the brain [14]–[18]. The distinctive focus of neuromorphic was on designing systems that highlighted simple processing components, often in the form of neurons, and establishing dense interconnections typically represented as synapses [6].

Another reason for neuromorphic implementation is computation speed [19]–[22]. Early neuromorphic developers emphasized faster neural network computation with custom chips, leveraging natural parallelism, as mentioned and specialized hardware compared to traditional von Neumann architectures [6].

The potential for very low power consumption is another key motivation [6]. The human brain, the key inspiration, operates on about 20 watts, demonstrating remarkable computational abilities within this power constraint [6]. Thus the desire to create a system with similar power requirements.

Another key motivator is real-time performance [6]. Leveraging natural parallelism and faster computation, neuromorphic systems outpaced von Neumann architectures in neural network computations. This advantage was especially valuable for real-time applications like control systems [23], real-time digital image reconstruction [24], and autonomous robot control [25].

Fault tolerance is another key motivation [6]. Early developers saw neural networks as a natural model for hardware implementation due to their inherent fault tolerance, primarily derived from their massively parallel representation [6]. The intrinsic resilience and adaptability of neural networks make them appealing for integration into physical hardware systems.

TABLE I
RANKING OF MOTIVATIONS FOR NEUROMORPHIC HARDWARE IMPLEMENTATION IN 2015

Rank	Motivation
Rank 1	Low Power
Rank 2	Parallelism
Rank 3	Fault Tolerance
Rank 4	Faster Computation
Rank 5	Scalability
Rank 6	Neuroscience
Rank 7	Von Neumann Bottleneck
Rank 8	Online Learning
Rank 9	Real-Time Performance
Rank 10	Footprint

Rank 1 denotes the most prevalent motivation, while Rank 10 corresponds to the least influential motivation.
Adapted from [6] and modified to a table for easier ranking.

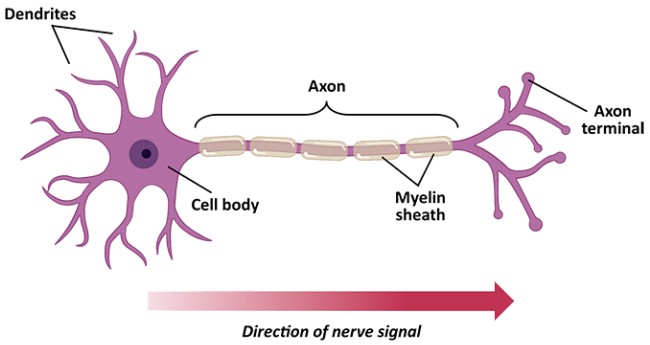

Fig. 3. An illustration showcasing the intricate structure of a biological neuron, featuring key components like dendrites, the cell body (soma), axons, and synapses. This representation, adapted from [28], emphasizes the fundamental architecture responsible for transmitting electrical signals and information in the nervous system.

Further driving the adoption of neuromorphic hardware are considerations for compact device size, neuroscience studies, scalability and the facilitation of online learning [6]. Table I provides the ranking for different motivations in the year 2015.

II. AREAS OF RESEARCH AND DEVELOPMENT

A. Models

1) Neuron Model: The neuron is acknowledged as the fundamental unit of the nervous system [26]. As described in [27], a neuron comprises dendrites, soma, and axons, as illustrated which is illustrated Figure 3. Neurotransmitters bind to dendritic receptors, causing a change in the neuron's potential [27]. The soma, or cell body, interprets this potential. If the potential change exceeds the neuron's potential threshold, the soma initiates and transmits a potential along the axon, triggering the release of neurotransmitters at the axon terminals [27]. These neurotransmitters are received by the subsequent cell [27].

Neuromorphic neuron models are classified into 5 distinct categories as shown in Table II. Some mimic the neuron operation(such as generation and propagation of action potentials), referred to as biologically plausible models or seek to replicate behaviour instead of the process, biologically inspired [6].

TABLE II
CATEGORIES OF NEURON MODELS

Model Type	Description	Existing Models/Implementations
Biologically Plausible	Neuron models that explicitly imitate the biological processes involving in the neuron	Hodgkin-Huxley, Morris Lecar
Biologically Inspired	Models focused on imitating neuron behaviour not process. Exhibit relatively lower computational demands compared to more plausible models	Izhikevich, Mihalas-Niebur, Fitzhugh-Nagumo and Hindmarsh-Rose
Neurons with Bio-Inspired Mechanisms	Introduces additional mechanisms to the neuron model for a greater level of biological detail.	Membrane dynamics, Ion-channel dynamics and Axon/dendrite model
Integrate-and-Fire	Ranges from simple to Izhikevich-level complexity, sufficient for spiking neural system despite not closely mimicking biological processes	Leaky Integrate-and-Fire, General Nonlinear Integrate-and-Fire, Quadratic Integrate-and-Fire and Adaptive Exponential Integrate-and-Fire Model.
McCulloch-Pitts	Models utilizing a linear predictor function and non-linear activation actions. Widely employed in hardware due to its adaptability	Activation functions include Sigmoid, hyperbolic tangent, piecewise linear, ramp-saturation, Radial Basis Function(RBF)

Categories in this table represent various neuromorphic neuron models, each briefly described. .
Refer to relevant references and literature in [6], [29]-[41]

2) Synapse Model: Synapses are cellular structures connecting neurons to enable the transmission of both chemical and electrical signals [42], [43]. Chemical synaptic transmission as described in [43], [44] involves the conversion of electrical signals within the presynaptic neuron into chemical signals, neurotransmitters, which are transmitted to the postsynaptic neuron. In contrast, electrical transmission relies on gap junctions that facilitate the direct flow of electrical signals from the presynaptic to the postsynaptic neuron [43], [45].

In addition to neuron models, ongoing research is focused on synaptic models, broadly categorised as biologically-inspired models and classical synaptic models designed for artificial neural networks. Synapse models tend to be simple as as possible due to their abundance in neuromorphic systems. However, they can get more complex especially when attempting to mimic biological processes such as plasticity and chemical interaction of synapses [6].

In spiking neuromorphic systems, synapse models are developed to incorporate plasticity [46], [47] and learning rules based on long-term and short-term potentiation and depression [48]. Synapse models have also been implemented in artificial neural network systems such as feed-forward networks, winner-takes-all and convolutional neural networks with learning rules such as Hebbian learning, least mean square and backpropagation with Gaussian synapse [6].

3) Network Model: Network models determine how the neurons and synapses are connected. Different factors must be considered when selecting a network model including biological inspiration and complexity, the topology of the network which is affected by available hardware, and the practicability of existing training algorithms for the model [6]. The different types of network models and implementations are shown in Table III.

TABLE III
NETWORK MODEL CATEGORIES AND EXISTING IMPLEMENTATIONS IN NEUROMORPHIC SYSTEMS

Model Category	Implementations
Spiking Models	Spiking Feed-Forward [49] Spiking Recurrent [50] Spiking Hopfield Network [51] Central Pattern Generators (CPGs) [52] Spiking Deep Belief Networks(DBN) [53] Spiking Probabilistic [54] Spiking Hebbian [55]
Feed-Forward(FF) Models	Convolutional [56] Probabilistic [57] Radial Basis Function(RBF) [58] Multilayer perceptron with delay [59] Extreme Learning Machines(ELM) [60]
Stochastic Models	Boltzmann Machines [61] Restricted Boltzmann Machines [62] Deep Belief Networks [63]
Recurrent	Hopfield [64] Reservoir Computing [65] Winner-Takes-All [66]
Unsupervised	Hebbian [67] Self Organising Maps(SOM) [68]
Visual	Convolutional Neural Networks [6] Cellular Neural Networks [69] Pulse-coupled Neural Networks [70]
Others	Cellular Automata [71] Fuzzy Neural Networks [72] Hierarchical Temporal Memory [73]

B. Algorithms

Algorithms refer to the specific computational processes used for learning, adaptation, and information processing within the neural network. These algorithms define how neurons, synapses, and networks interact, update weights, and undergo training. They determine the rules by which the system learns and operates. In neuromorphic systems, algorithm choices are influenced by key factors. These include the selected neuron, synapse, and network models, the decision to train on-chip or off-chip, and the choice between online, unsupervised, and offline, supervised learning. Recognizing online learning's potential as a post-Moore's law complementary architecture adds significance [6].

1) Supervised: In neuromorphic systems programming, back-propagation, a supervised learning method, is a widely used offline and on-chip for various neural network architectures [74], [75]. Despite its adaptability, constraints on neuron models, network topologies, and hardware implementation challenges are considerations [6]. On-chip alternatives like least-mean-squares and weight perturbation offer flexibility [76], [77], especially in convolutional neural networks [78]. Specialized mechanisms cater to specific model types, including Boltzmann machines and hierarchical temporal mem-

ory [73], [79]. Nature-inspired algorithms (e.g., differential evolution, genetic algorithms, particle swarm optimization) are favoured for hardware adaptability [80], [81]. Chip-in-the-loop methods facilitate hardware evaluations during training, considering device characteristics [6]. These approaches signify ongoing efforts to address challenges and enhance programming efficiency in neuromorphic systems.

2) Unsupervised: In neuromorphic systems programming, on-chip, on-line unsupervised training mechanisms are crucial for unlocking the full potential of these systems. Early implementations utilized self-organizing maps [82], evolving into recent advancements. Hebbian-type learning rules [83], accommodating both supervised and unsupervised aspects, are widely adopted. Spike-timing dependent plasticity (STDP) [84], a prevalent on-line, unsupervised learning mechanism, adjusts synapse weights based on the timing of pre-synaptic and post-synaptic neuron firing. Custom circuits enhance synapse depression and potentiation in biologically-inspired implementations [85]. Notably, the broad applicability of STDP across various applications is still under exploration.

C. Hardware Implementation

1) Digital, Analog and Mixed Neuromorphic: In the domain of digital neuromorphic systems, two key categories, Field Programmable Gate Arrays (FPGAs) and Full Custom/Application-Specific Integrated Circuit (ASIC) chips, play crucial roles [6]. FPGAs are often utilized for their programmability, offering versatility in network topologies and algorithms, though with limitations in small, low-power systems [6]. In contrast, ASIC chips like IBM's TrueNorth and SpiNNaker showcase different approaches. TrueNorth optimizes for efficiency in a fixed spiking neural network model [86], while SpiNNaker prioritizes flexibility at the cost of higher energy consumption [87]. Comparative analysis reveals TrueNorth's 25 pJ per connection efficiency and SpiNNaker's 10 nJ per connection flexibility with increased energy usage [6]. The choice between FPGAs and ASICs hinges on project goals, with FPGAs providing versatility and ease of use, and ASICs delivering optimized solutions for specific neuromorphic models.

Analog systems are classified into programmable and custom chip implementations, featuring Field Programmable Analog Arrays (FPAAs) [88] and dedicated circuits like the field programmable neural array (FPNA) [89] and NeuroFPAA [90] for neuromorphic applications. Shared characteristics with biological systems, such as charge conservation and amplification, drive the synergy between custom analogue integrated circuits and neuromorphic systems [6]. These implementations, inspired by biological neural systems, offer solutions to analogue circuit challenges like global asynchrony and unreliable components [6]. Operating in the subthreshold mode for enhanced power efficiency [91], these systems exhibit versatility in addressing challenges related to noise and asynchrony in analogue circuits.

In neuromorphic systems, a prevalent approach involves combining analogue and digital components to leverage their respective strengths. Analogue circuitry, mirroring biological systems, is commonly used for processing neurons and synapses, while digital components address challenges like unreliability, offering reduced noise and increased reliability. Some platforms primarily employ analogue components but incorporate digital communication within and between chips, typically using digital spikes for communication. Projects like Neurogrid [92] and BrainScaleS [3] exemplifies the mixed analog/digital category, with Neurogrid operating in subthreshold mode and BrainScaleS in superthreshold mode, achieving a remarkable 10,000x speed-up [94]. These instances highlight the versatility and benefits of mixed analogue/digital neuromorphic systems.

2) Circuit Components: Neuromorphic systems incorporate diverse circuit-level components, including memristors, known for their plasticity-like properties in synapse implementations [6]. Challenges such as memristor behaviour and geometry variations persist [95], [96]. CBRAM and atomic switches offer non-volatile memory solutions, demonstrating synaptic plasticity. Phase change memory elements enable high-density synapse implementations with STDP capabilities. Spintronics presents beyond-CMOS technology, offering tunable functionalities compatible with CMOS. Floating gate transistors, adapted from digital storage, serve as analogue memory cells and synapse implementations, often incorporating learning mechanisms. Optical and photonic components, once less popular, are gaining attention for their potential in ultrafast operation, moderate complexity, and programmability, contributing to optical synapses and proposed photonic neuron implementations in recent neuromorphic systems [6].

III. Applications

Neuromorphic computing has emerged as a transformative technology with wide-ranging applications, showcasing its adaptability across various domains. From sensory systems to robotics, control mechanisms, and beyond, neuromorphic systems demonstrate their prowess in tackling complex tasks with efficiency.

In sensory systems, neuromorphic implementations extend beyond the popular vision-based models, encompassing auditory, olfactory, and somatosensory/touch-based systems. Vision-based systems, with architectures inspired by biological visual systems, have seen widespread adoption, illustrating their capability in tasks such as image processing, classification, and segmentation [97], [98].

Medical interfaces benefit from the compact size and low power requirements of neuromorphic devices. Brain-machine interfaces and implantable devices leverage these characteristics, providing innovative solutions for medical treatment and monitoring. This represents a significant stride towards integrating neuromorphic technology into the healthcare sector [99].

Robotics applications also stand out as a common arena for neuromorphic systems. The need for small, power-efficient systems aligns well with the capabilities of neuromorphic implementations. Tasks such as behavior learning, locomotion control, and autonomous navigation showcase the effectiveness of these systems in real-world applications [100].

Control tasks, which often demand real-time performance, small form factors, and low power consumption, find suitable

solutions in neuromorphic systems. The cart-pole problem, a classic control test case, exemplifies the system's aptitude for tasks with temporal processing components [101].

The field of image processing witnesses the widespread adoption of neuromorphic systems in tasks like edge detection, image compression, and classification. These systems prove particularly adept at image-based applications, demonstrating their versatility across datasets like MNIST, CIFAR10, and ImageNet [102].

Sound-based recognition tasks, including speech and music recognition, find a natural fit with neuromorphic systems. Their inherent temporal processing capabilities make them suitable for tasks with real-time constraints, providing efficient solutions for tasks such as speaker recognition and noise filtering [103].

Video applications, ranging from object recognition to motion detection and activity recognition, highlight the adaptability of neuromorphic systems in processing visual data. Natural Language Processing (NLP) tasks, which often require recurrent networks, also witness successful applications of neuromorphic technology [104].

Real-time sensing applications, including smart sensors and anomaly detectors in domains such as traffic, biological data, and industrial processes, showcase the practical utility of neuromorphic systems. Their ability to operate with a small footprint and low power consumption positions them as valuable tools in these domains [6].

In data classification, neuromorphic systems address a diverse set of applications, from accident diagnosis and stock price prediction to medical diagnostics. Benchmarking tasks, such as the two spirals problem and N-bit parity tasks, provide insights into the behavioural characteristics of neuromorphic systems [105].

Beyond neural network applications, neuromorphic systems extend their influence to non-neural network domains. Graph algorithms, flock motion simulation, and optimization problems serve as examples of their applicability in diverse computational tasks [6]. As neuromorphic computers become more accessible, their potential for a broad spectrum of applications continues to unfold, marking a paradigm shift in computing architectures.

IV. Conclusion

In summary, the review on neuromorphic computing provides a comprehensive overview of its motivations and diverse architectural models. The persistent need for a non-Von Neumann architecture, characterized by low power consumption, massive parallelism, real-time performance, and online learning capability, remains a driving force.

The paper also delved into the variety of neuron, synapse, and network models, highlighting the flexibility in designing neuromorphic systems at an abstract level. The absence of a singular encompassing model underscores the ongoing diversity in approaches, ranging from feed-forward neural networks to detailed biological neural network emulators.

The paper also examined learning algorithms both supervised and unsupervised. Some supervised learning methods included backpropagation, least mean squares and genetic algorithms. Unsupervised learning methods such as self-organising maps, Hebbian learning rules and STDP.

Lastly, the applications of neuromorphic computing across various domains were highlighted such as the medical field, robotics, image processing and control systems.

References

[1] P. S. Pacheco and M. Malensek, An Introduction to Parallel Programming, 2nd ed. Elsevier, 2021. Available: https://www.sciencedirect.com/book/9780128046050/an-introduction-to-parallel-programming

[2] L. Wanhammar, DSP Integrated Circuits. Burlington: Academic Press, 1999, pp. 357–385. Available: https://www.sciencedirect.com/science/article/abs/pii/B9780127345307500088. [Accessed: Dec. 25, 2023]

[3] Chen Ding and K. Kennedy, "The memory of bandwidth bottleneck and its amelioration by a compiler," Proceedings 14th International Parallel and Distributed Processing Symposium. IPDPS 2000, Cancun, Mexico, 2000, pp. 181-189, doi: 10.1109/IPDPS.2000.845980.

[4] R. G. Brown, "Bottlenecks," Duke.edu, May 24, 2004. Available: https://webhome.phy.duke.edu/~rgb/Beowulf/beowulf_book/beowulf_book/node24.html. [Accessed: Dec. 26, 2023]

[5] J. Jiang and L. Gu, Encyclopedia of Materials: Electronics, vol. 3. Oxford: Academic Press, 2023, pp. 658–675. Available: https://www.sciencedirect.com/science/article/abs/pii/B9780128197288001364. [Accessed: Dec. 26, 2023]

[6] C. D. Schuman et al., "A Survey of Neuromorphic Computing and Neural Networks in Hardware," Neural and Evolutionary Computing, May 2017, doi: https://doi.org/10.48550/arXiv.1705.06963.Available:https://arxiv.org/abs/1705.06963

[7] D. Monroe, "Neuromorphic computing gets ready for the (really) big time," Communications of the ACM, vol. 57, no. 6, pp. 13–15, 2014.

[8] S. H. Fuller and L. I. Millett, "Computing Performance: Game Over or Next Level?," in Computer, vol. 44, no. 1, pp. 31-38, Jan. 2011, doi: 10.1109/MC.2011.15.

[9] L. A. Girifalco, J. D. Venables, R. E. Marchant, C. K. N. Patel, R. L. McCullough, and D. S. Kukich, "Materials science - Oil platforms," www.britannica.com. Britannica, Dec. 12, 2023. Available: https://www.britannica.com/technology/materials-science/Oil-platforms. [Accessed: May 03, 2023]

[10] King's College London, "What is neuroscience?," www.kcl.ac.uk. Available: https://www.kcl.ac.uk/neuroscience/about/what-is-neuroscience#:~:text=At%20its%20most%20basic%2C%20neuroscience. [Accessed: Dec. 26, 2023]

[11] ACM CCECC, "Computer Engineering - ACM CCECC," ccecc.acm.org. Available: https://ccecc.acm.org/guidance/computer-engineering#:~:text=Computer%20engineering%20is%20defined%20as. [Accessed: Dec. 26, 2023]

[12] New World Encyclopedia, "Electronic engineering," www.newworldencyclopedia.org. New World Encyclopedia. Available: https://www.newworldencyclopedia.org/entry/Electronic_engineering. [Accessed: Dec. 26, 2023]

[13] G. G. Belford and A. Tucker, "computer science," Encyclopædia Britannica. Mar. 14, 2019. Available: https://www.britannica.com/science/computer-science. [Accessed: Dec. 26, 2023]

[14] A. F. Murray and A. V. Smith, "Asynchronous vlsi neural networks using pulse-stream arithmetic," Solid-State Circuits, IEEE Journal of, vol. 23, no. 3, pp. 688–697, 1988.

[15] F. Blayo and P. Hurat, "A vlsi systolic array dedicated to hopfield neural network," in VLSI for Artificial Intelligence. Springer, 1989, pp. 255–264.

[16] F. Salam, "A model of neural circuits for programmable vlsi implementation," in Circuits and Systems, 1989., IEEE International Symposium on. IEEE, 1989, pp. 849–851.

[17] S. Bibyk, M. Ismail, T. Borgstrom, K. Adkins, R. Kaul, N. Khachab, and S. Dupuie, "Current-mode neural network building blocks for analog mos vlsi," in Circuits and Systems, 1990., IEEE International Symposium on. IEEE, 1990, pp. 3283–3285.

[18] F. Distante, M. Sami, and G. S. Gajani, "A general configurable architecture for wsi implementation for neural nets," in Wafer Scale Integration, 1990. Proceedings.,[2nd] International Conference on. IEEE, 1990, pp. 116–123.

[19] J. B. Burr, "Digital neural network implementations," Neural networks, concepts, applications, and implementations, vol. 3, pp. 237– 285, 1991.

[20] M. Chiang, T. Lu, and J. Kuo, "Analogue adaptive neural network circuit," IEE Proceedings G (Circuits, Devices and Systems), vol. 138, no. 6, pp. 717–723, 1991.

[21] K. Madani, P. Garda, E. Belhaire, and F. Devos, "Two analog counters for neural network implementation," Solid-State Circuits, IEEE Journal of, vol. 26, no. 7, pp. 966–974, 1991.

[22] A. F. Murray, D. Del Corso, and L. Tarassenko, "Pulse-stream vlsi neural networks mixing analog and digital techniques," Neural Networks, IEEE Transactions on, vol. 2, no. 2, pp. 193–204, 1991.

[23] P. Hasler and L. Akers, "Vlsi neural systems and circuits," in Computers and Communications, 1990. Conference Proceedings., Ninth Annual International Phoenix Conference on. IEEE, 1990, pp. 31– 37.

[24] J.-C. Lee and B. J. Sheu, "Parallel digital image restoration using adaptive vlsi neural chips," in Computer Design: VLSI in Computers and Processors, 1990. ICCD'90. Proceedings, 1990 IEEE International Conference on. IEEE, 1990, pp. 126–129.

[25] L. Tarassenko, M. Brownlow, G. Marshall, J. Tombs, and A. Murray, "Real-time autonomous robot navigation using vlsi neural networks," in Advances in neural information processing systems, 1991, pp. 422–428.

[26] Wikipedia Contributors, "Neuron," Wikipedia, Feb. 28, 2019. Available: https://en.wikipedia.org/wiki/Neuron

[27] Neuroscientifically Challenged, "2-Minute Neuroscience: The Neuron," YouTube. Jul. 22, 2014. Available: https://www.youtube.com/watch?v=6qS83wD29PY

[28] National Institute of Neurological Disorders and Stroke, "Brain Basics: Know Your Brain — National Institute of Neurological Disorders and Stroke," www.ninds.nih.gov, Sep. 26, 2022. Available: https://www.ninds.nih.gov/health-information/public-education/brain-basics/brain-basics-know-your-brain. [Accessed: Dec. 30, 2023]

[29] Wikipedia Contributors, "Hodgkin–Huxley model," Wikipedia, Oct. 12, 2023. Available: https://en.wikipedia.org/wiki/Hodgkin%E2%80%93Huxley_model. [Accessed: Dec. 30, 2023]

[30] L. Wei and D. Li, "Stochastic Morris–Lecar model with time delay under magnetic field excitation," Chaos, Solitons & Fractals, vol. 173, p. 113715, Aug. 2023, doi: https://doi.org/10.1016/j.chaos.2023.113715. Available: https://www.sciencedirect.com/science/article/abs/pii/S0960077923006161. [Accessed: Dec. 30, 2023]

[31] Wikipedia Contributors, "Morris–Lecar model," Wikipedia, Oct. 12, 2023. Available: https://en.wikipedia.org/wiki/Morris%E2%80%93Lecar_model. [Accessed: Dec. 30, 2023]

[32] E. M. Izhikevich, "Simple model of spiking neurons," IEEE Transactions on neural networks, vol. 14, no. 6, pp. 1569–1572, 2003.

[33] S. Mihalas , and E Niebur, "A generalized linear integrate-and-fire neural model produces diverse spiking behaviors" Neural computation, vol. 21, no. 3, pp. 704–718, 2009.

[34] X. Fang, S. Duan, and L. Wang, "Memristive FHN Spiking Neuron Model and Brain-Inspired Threshold Logic Computing," Neurocomputing, vol. 517, Aug. 2022, doi: https://doi.org/10.1016/j.neucom.2022.08.056

[35] M. Nouri, Gh. R. Karimi, A. Ahmadi, and D. Abbott, "Digital multiplierless implementation of the biological FitzHugh–Nagumo model," Neurocomputing, vol. 165, pp. 468–476, Oct. 2015, doi: https://doi.org/10.1016/j.neucom.2015.03.084

[36] M. Hayati, M. Nouri, D. Abbott, and S. Haghiri, "Digital multiplierless realization of two-coupled biological hindmarsh–rose neuron model," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 63, no. 5, pp. 463–467, 2016.

[37] R. Wang, T. J. Hamilton, J. Tapson, and A. van Schaik, "A generalised conductance-based silicon neuron for large-scale spiking neural networks," in Circuits and Systems (ISCAS), 2014 IEEE International Symposium on. IEEE, 2014, pp. 1564–1567.

[38] K. M. Hynna and K. Boahen, "Neuronal ion-channel dynamics in silicon," in Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on. IEEE, 2006, pp. 4–pp.

[39] P. Hasler, S. Kozoil, E. Farquhar, and A. Basu, "Transistor channel dendrites implementing hmm classifiers," in Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on. IEEE, 2007, pp. 3359–3362.

[40] W. Gerstner and R. Brette, "Adaptive exponential integrate-and-fire model," Scholarpedia, vol. 4, no. 6, p. 8427, 2009, doi: https://doi.org/10.4249/scholarpedia.8427

[41] Wikipedia Contributors, "Perceptron," Wikipedia, Dec. 31, 2023. Available: https://en.wikipedia.org/wiki/Perceptron. [Accessed: Jan. 02, 2024]

[42] M. J. Caire and M. Varacallo, "Physiology, Synapse," Nih.gov, Nov. 13, 2018. Available: https://www.ncbi.nlm.nih.gov/books/NBK526047/

[43] Wikipedia Contributors, "Synapse," Wikipedia, Jan. 06, 2024. Available: https://en.wikipedia.org/wiki/Synapse. [Accessed: Jan. 08, 2024]

[44] Britannica, "Synapse — anatomy — Britannica," Encyclopædia Britannica. 2020. Available: https://www.britannica.com/science/synapse. [Accessed: Jan. 08, 2024]

[45] A. E. Pereda, "Electrical synapses and their functional interactions with chemical synapses," Nature Reviews Neuroscience, vol. 15, no. 4, pp. 250–263, Mar. 2014, doi: https://doi.org/10.1038/nrn3708. Available: https://www.nature.com/articles/nrn3708

[46] M. Stampanoni Bassi, E. Iezzi, L. Gilio, D. Centonze, and F. Buttari, "Synaptic Plasticity Shapes Brain Connectivity: Implications for Network Topology," International Journal of Molecular Sciences, vol. 20, no. 24, p. 6193, Dec. 2019, doi: https://doi.org/10.3390/ijms20246193

[47] N. Du et al., "Synaptic Plasticity in Memristive Artificial Synapses and Their Robustness Against Noisy Inputs," Frontiers in Neuroscience, vol. 15, Jul. 2021, doi: https://doi.org/10.3389/fnins.2021.660894

[48] M. Migliore and P. Lansky, "Long-Term Potentiation and Depression Induced by a Stochastic Conditioning of a Model Synapse," Biophysical Journal, vol. 77, no. 3, pp. 1234–1243, Sep. 1999, doi: https://doi.org/10.1016/s0006-3495(99)76975-4

[49] A. Bofill-i-Petit and A. F. Murray, "Synchrony detection and amplification by silicon neurons with STDP synapses," in IEEE Transactions on Neural Networks, vol. 15, no. 5, pp. 1296-1304, Sept. 2004, doi: 10.1109/TNN.2004.832842.

[50] P. Camilleri, M. Giulioni, M. Mattia, J. Braun and P. Del Giudice, "Self-sustained activity in attractor networks using neuromorphic VLSI," The 2010 International Joint Conference on Neural Networks (IJCNN), Barcelona, Spain, 2010, pp. 1-6, doi: 10.1109/IJCNN.2010.5596342.

[51] J. Dungen and J. . -J. Brault, "Simulated control of a tracking mobile robot by four aVLSI integrate-and-fire neurons paired into maps," Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005., Montreal, QC, Canada, 2005, pp. 695-699 vol. 2, doi: 10.1109/IJCNN.2005.1555936.

[52] E. Donati, G. Indiveri and C. Stefanini, "A novel spiking CPG-based implementation system to control a lamprey robot," 2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob), Singapore, 2016, pp. 1364-1364, doi: 10.1109/BIOROB.2016.7523822.

[53] E. Stromatias, D. Neil, M. Pfeiffer, F. Galluppi, S. B. Furber, and S.-C. Liu, "Robustness of spiking deep belief networks to noise and reduced bit precision of neuro-inspired hardware platforms," Frontiers in neuroscience, vol. 9, 2015.

[54] H. -Y. Hsieh and K. -T. Tang, "An on-chip learning, low-power probabilistic spiking neural network with long-term memory," 2013 IEEE Biomedical Circuits and Systems Conference (BioCAS), Rotterdam, Netherlands, 2013, pp. 5-8, doi: 10.1109/BioCAS.2013.6679626.

[55] A. Bofill, D. Thompson, and A. F. Murray, "Circuits for vlsi implementation of temporally asymmetric hebbian learning," in Advances in Neural Information processing systems, 2001, pp. 1091–1098.

[56] F. Conti and L. Benini, "A ultra-low-energy convolution engine for fast brain-inspired vision in multicore clusters," in Design, Automation & Test in Europe Conference & Exhibition (DATE), 2015. IEEE, 2015, pp. 683–688.

[57] N. Aibe, M. Yasunaga, I. Yoshihara, and J. H. Kim, "A probabilistic neural network hardware system using a learning-parameter parallel architecture," in Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on, vol. 3. IEEE, 2002, pp. 2270–2275.

[58] S. M. Fakhraie, H. Farshbaf, and K. C. Smith, "Scalable closed-boundary analog neural networks," Neural Networks, IEEE Trans- actions on, vol. 15, no. 2, pp. 492–504, 2004.

[59] J. Van der Spiegel, C. Donham, R. Etienne-Cummings, S. Fernando, P. Mueller, and D. Blackman, "Large scale analog neural computer with programmable architecture and programmable time constants for temporal pattern analysis," in Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on, vol. 3. IEEE, 1994, pp. 1830–1835.

[60] C. Merkel and D. Kudithipudi, "Neuromemristive extreme learning machines for pattern classification," in VLSI (ISVLSI), 2014 IEEE Computer Society Annual Symposium on. IEEE, 2014, pp. 77–82.

[61] M. N. Bojnordi and E. Ipek, "Memristive Boltzmann machine: A hardware accelerator for combinatorial optimization and deep learning," 2016 IEEE International Symposium on High Performance Computer Architecture (HPCA), Barcelona, Spain, 2016, pp. 1-13, doi: 10.1109/HPCA.2016.7446049.

[62] C. C. Lu, C. Y. Hong and H. Chen, "A Scalable and Programmable Architecture for the Continuous Restricted Boltzmann Machine in

VLSI," 2007 IEEE International Symposium on Circuits and Systems, New Orleans, LA, USA, 2007, pp. 1297-1300, doi: 10.1109/ISCAS. 2007.378409.

[63] B. Ahn, "Computation of deep belief networks using special-purpose hardware architecture," in Neural Networks (IJCNN), 2014 International Joint Conference on. IEEE, 2014, pp. 141–148.

[64] X. Guo, F. Merrikh-Bayat, L. Gao, B. D. Hoskins, F. Alibart, B. Linares-Barranco, L. Theogarajan, C. Teuscher, and D. B. Strukov, "Modeling and experimental demonstration of a hopfield network analog-to-digital converter with hybrid cmos/memristor circuits," Frontiers in neuroscience, vol. 9, 2015.

[65] D. Kudithipudi, Q. Saleh, C. Merkel, J. Thesing, and B. Wysocki, "Design and analysis of a neuromemristive reservoir computing architecture for biosignal processing," Frontiers in Neuroscience, vol. 9, p. 502, 2015.

[66] P. Hylander, J. Meader, and E. Frie, "Vlsi implementation of pulse coded winner take all networks," in Circuits and Systems, 1993., Proceedings of the 36th Midwest Symposium on. IEEE, 1993, pp. 758–761.

[67] T. Shima, T. Kimura, Y. Kamatani, T. Itakura, Y. Fujita, and T. Iida, "Neuro chips with on-chip backprop and/or hebbian learning," in Solid-State Circuits Conference, 1992. Digest of Technical Papers. 39th ISSCC, 1992 IEEE International. IEEE, 1992, pp. 138–139.

[68] H. Tamukoh and M. Sekine, "A dynamically reconfigurable platform for self-organizing neural network hardware," in Neural Information Processing. Models and Applications. Springer, 2010, pp. 439–446.

[69] J. Secco, M. Farina, D. Demarchi, and F. Corinto, "Memristor cellular automata through belief propagation inspired algorithm," in 2015 International SoC Design Conference (ISOCC). IEEE, 2015, pp. 211–212.

[70] H. Torikai, H. Hamanaka, and T. Saito, "Novel digital spiking neuron and its pulse-coupled network: spike position coding and multiplex communication," in Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on, vol. 5. IEEE, 2005, pp. 3249–3254.

[71] T. Matsubara and H. Torikai, "Asynchronous cellular automaton-based neuron: theoretical analysis and on-fpga learning," Neural Networks and Learning Systems, IEEE Transactions on, vol. 24, no. 5, pp. 736– 748, 2013.

[72] Y.-H. Kuo, C.-I. Kao, and J.-J. Chen, "A fuzzy neural network model and its hardware implementation," Fuzzy Systems, IEEE Transactions on, vol. 1, no. 3, pp. 171–183, 1993.

[73] T. Ibrayev, A. P. James, C. Merkel, and D. Kudithipudi, "A design of htm spatial pooler for face recognition using memristor-cmos hybrid circuits," in 2016 IEEE International Symposium on Circuits and Systems, IEEE, May, 2016.

[74] S. Haykin, "Neural networks: A comprehensive foundation," 2004.

[75] E. Zamanidoost, M. Klachko, D. Strukov, and I. Kataeva, "Low area overhead in-situ training approach for memristor-based classifier," in Nanoscale Architectures (NANOARCH), 2015 IEEE/ACM International Symposium on. IEEE, 2015, pp. 139–142.

[76] D. Christiani, C. Merkel, and D. Kudithipudi, "Invited: Towards a scalable neuromorphic hardware for classification and prediction with stochastic no-prop algorithms," in Quality Electronic Design (ISQED), 2016 17th International Symposium on. IEEE, 2016, pp. 124–128.

[77] Y. Maeda and T. Tada, "Fpga implementation of a pulse density neural network with learning ability using simultaneous perturbation," Neural Networks, IEEE Transactions on, vol. 14, no. 3, pp. 688–695, 2003.

[78] J. Fieres, J. Schemmel, and K. Meier, "Training convolutional net- works of threshold neurons suited for low-power hardware imple- mentation," in Neural Networks, 2006. IJCNN'06. International Joint Conference on. IEEE, 2006, pp. 21–28.

[79] A. M. Sheri, A. Rafique, W. Pedrycz, and M. Jeon, "Contrastive divergence for memristor-based restricted boltzmann machine," Engineering Applications of Artificial Intelligence, vol. 37, pp. 336–342, 2015.

[80] D. Maliuk, H.-G. Stratigopoulos, and Y. Makris, "An analog vlsi multilayer perceptron and its application towards built-in self-test in analog circuits," in On-Line Testing Symposium (IOLTS), 2010 IEEE 16th International. IEEE, 2010, pp. 71–76.

[81] D. Braendler and T. Hendtlass, "The suitability of particle swarm optimisation for training neural hardware," in Developments in Applied Artificial Intelligence. Springer, 2002, pp. 190–199.

[82] J. Mann, R. Lippmann, B. Berger, and J. Raffel, "A self-organizing neural net chip," in Custom Integrated Circuits Conference, 1988., Proceedings of the IEEE 1988. IEEE, 1988, pp. 10–3.

[83] B. Linares-Barranco, E. Sanchez-Sinencio, A. Rodriguez-Vazquez, and J. Huertas, "Modular analog continuous-time vlsi neural networks with on-chip hebbian learning and analog storage," in Circuits and Systems, 1992. ISCAS'92. Proceedings., 1992 IEEE International Symposium on, vol. 3. IEEE, 1992, pp. 1533–1536.

[84] J. Schemmel, D. Bruderle, A. Grubl, M. Hock, K. Meier, and S. Millner, "A wafer-scale neuromorphic hardware system for large- scale neural modeling," in Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on. IEEE, 2010, pp. 1947– 1950.

[85] S. R. Schultz and M. A. Jabri, "A silicon basis for synaptic plasticity," Neural Processing Letters, vol. 2, no. 6, pp. 23–27, 1995.

[86] F. Akopyan, J. Sawada, A. Cassidy, R. Alvarez-Icaza, J. Arthur, P. Merolla, N. Imam, Y. Nakamura, P. Datta, G.-J. Nam et al., "Truenorth: Design and tool flow of a 65mw 1 million neuron programmable neurosynaptic chip."

[87] R. Araújo,N. Waniek,and J. Conradt,"Development of a Dynamically Extendable SpiNNaker Chip Computing Module" in Artificial Neural Networks and Machine Learning–ICANN 2014. Springer, 2014, pp. 821–828.

[88] S. Shapero, C. Rozell, and P. Hasler, "Configurable hardware integrate and fire neurons for sparse approximation," Neural Networks, vol. 45, pp. 134–143, 2013.

[89] E. Farquhar, C. Gordon, and P. Hasler, "A field programmable neural array," in Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on. IEEE, 2006, pp. 4–pp.

[90] M. Liu, H. Yu, and W. Wang, "Fpaa based on integration of cmos and nanojunction devices for neuromorphic applications," in Nano- Net. Springer, 2009, pp. 44–48.

[91] ——, "An experimentation platform for on-chip integration of analog neural networks: A pathway to trusted and robust analog/rf ics," 2014.

[92] B. V. Benjamin, P. Gao, E. McQuinn, S. Choudhary, A. R. Chandrasekaran, J. Bussat, R. Alvarez-Icaza, J. V. Arthur, P. A. Merolla, and K. Boahen, "Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations," Proceedings of the IEEE, vol. 102, no. 5, pp. 699–716, 2014.

[93] T. Pfeil, T. C. Potjans, S. Schrader, W. Potjans, J. Schemmel, M. Diesmann, and K. Meier, "Is a 4-bit synaptic weight resolution enough?–constraints on enabling spike-timing dependent plasticity in neuromorphic hardware," Frontiers in neuroscience, vol. 6, 2012.

[94] S. Furber, "Large-scale neuromorphic computing systems," Journal of Neural Engineering, vol. 13, no. 5, p. 051001, 2016.

[95] A. Serb, R. Berdan, A. Khiat, S. Li, E. Vasilaki, C. Papavassiliou, and T. Prodromakis, "Memristors as synapse emulators in the context of event-based computation," in Circuits and Systems (ISCAS), 2014 IEEE International Symposium on. IEEE, 2014, pp. 2085–2088.

[96] R. E. Pino, H. Li, Y. Chen, M. Hu, and B. Liu, "Statistical memristor modeling and case study in neuromorphic computing," in Design Automation Conference (DAC), 2012 49th ACM/EDAC/IEEE. IEEE, 2012, pp. 585–590.

[97] C. Markan, P. Gupta, and M. Bansal, "An adaptive neuromorphic model of ocular dominance map using floating gate ?synapse?" Neural Networks, vol. 45, pp. 117–133, 2013.

[98] A. Rios-Navarro, E. Cerezuela-Escudero, M. Dominguez-Morales, A. Jimenez-Fernandez, G. Jimenez-Moreno, and A. Linares-Barranco, "Real-time motor rotation frequency detection with event-based visual and spike-based auditory aer sensory integration for fpga," in Event-based Control, Communication, and Signal Processing (EBCCSP), 2015 International Conference on. IEEE, 2015, pp. 1–6.

[99] S.Harrer,I.Kiral-Kornek,R.Kerr,B.Mashford,J.Tang,A.J.Yepes, and H. Deligianni, "From wearables to thinkables-deep learning, nano-biosensors and the next generation of mobile devices," White Paper, ICONN, 2016.

[100] S. Pande, F. Morgan, S. Cawley, T. Bruintjes, G. Smit, B. McGinley, S. Carrillo, J. Harkin, and L. McDaid, "Modular neural tile architecture for compact embedded hardware spiking neural network," Neural processing letters, vol. 38, no. 2, pp. 131–153, 2013.

[101] T. Fukuda, T. Shibata, M. Tokita, and T. Mitsuoka, "Neuromorphic control: adaptation and learning," Industrial Electronics, IEEE Transactions on, vol. 39, no. 6, pp. 497–503, 1992

[102] T. Serrano-Gotarredona, T. Masquelier, T. Prodromakis, G. Indiveri, and B. Linares-Barranco, "Stdp and stdp variations with memristors for spiking neuromorphic learning systems," Frontiers in neuro- science, vol. 7, 2013.

[103] A. Amir, P. Datta, W. P. Risk, A. S. Cassidy, J. Kusnitz, S. K. Esser, A. Andreopoulos, T. M. Wong, M. Flickner, R. Alvarez-Icaza et al., "Cognitive computing programming paradigm: a corelet language for composing networks of neurosynaptic cores," in Neural Networks (IJCNN), The 2013 International Joint Conference on. IEEE, 2013, pp. 1–10.

[104] M. Suri and B. DeSalvo, "Phase change memory and chalcogenide materials for neuromorphic applications: Emphasis on synaptic plasticity," in Advances in Neuromorphic Memristor Science and Applications. Springer, 2012, pp. 155–178.
[105] M. M. Syiam, H. Klash, I. Mahmoud, and S. Haggag, "Hardware implementation of neural network on fpga for accidents diagnosis of the multi-purpose research reactor of egypt," in Microelectronics, 2003. ICM 2003. Proceedings of the 15th International Conference on. IEEE, 2003, pp. 326–329.