NEUROMORPHIC COMPUTING

SHIV TIKOO, IIT Hyderabad, India

VISHAL VIJAY DEVADIGA, IIT Hyderabad, India

1 ABSTRACT

In the landscape of computational advancements, a new frontier has emerged, known as neuromorphic computing. This innovative approach, inspired by the intricate workings of the human brain, promises to transcend the boundaries of traditional Von Neumann architecture. This paper embarks on a journey through the recent developments in neuromorphic computing.

The advent of neuromorphic computing is a response to the escalating need for high-performance computing systems capable of managing complex tasks with superior efficiency. Conventional computing architectures are grappling with challenges related to speed, power consumption, and scalability. Neuromorphic computing, by emulating the neural structure of the brain, presents a potential resolution to these challenges.

At the heart of neuromorphic computing lie Spiking Neural Networks (SNNs). These networks deviate from traditional artificial neural networks by communicating through spikes, discrete events that occur at specific moments. This mode of communication mirrors the way neurons in the brain interact, leading to more efficient information processing. Another pivotal aspect of neuromorphic computing is Memristor technologies. Memristors, or memory resistors, are passive circuit elements that maintain a relationship between the time integrals of current and voltage across a device. Their unique properties make them ideal for implementing synaptic weights in neuromorphic systems.

As we look towards the future, neuromorphic computing is poised to play an increasingly significant role in the next generation of computing technologies. This abstract serves as a snapshot of the current state of neuromorphic computing and aims to pique interest in the detailed discussions that ensue in the rest of the paper. This paper is intended to be a valuable resource for those intrigued by the development and potential of neuromorphic computing.

Authors' Contact Information: Shiv Tikoo, IIT Hyderabad, , India; Vishal Vijay Devadiga, IIT Hyderabad, , India.
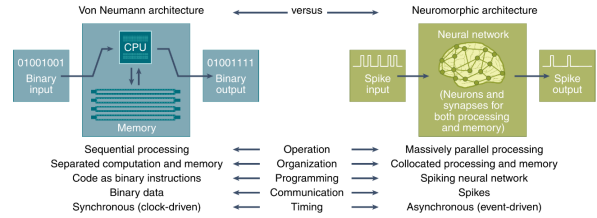
Fig. 1. Comparision of Von Neumann Architecture with Neuromorphic Architecture [3]

2 INTRODUCTION

The journey of computing hardware development has been a testament to human ingenuity, transitioning from the room-sized mainframes of the past to the compact, high-performance devices that are now commonplace. Yet, as we continue to push the envelope of miniaturization, we are approaching the inherent limits of traditional computing architectures. The relentless pursuit of Moore's Law, which foresaw a doubling of transistor density approximately every two years, is now confronting the challenges of diminishing returns as feature sizes near the atomic scale.

2.1 Need for Neuromorphic Computing

The evolution of computing hardware has been a cornerstone of technological progress over the past several decades. Reflecting on the fact that the first computer was the size of a room and had less processing power than a modern calculator, the strides made are truly extraordinary and have revolutionized the way we live and work. However, the current paradigm of computing is nearing its limits in terms of speed, power consumption, and scalability. Coupled with the exponential growth of data generation, there is a pressing need for a new computing paradigm that can keep pace with the demands of the modern world. Neuromorphic computing is anticipated to be this new paradigm and holds the potential to revolutionize our approach to computing.

The human brain, a highly complex organ, is capable of processing vast amounts of information in parallel. It is estimated that the human brain comprises around 100 billion neurons and 100 trillion synapses. The brain is also remarkably energy efficient, operating at a low frequency of around 100 Hz and consuming only about 20 watts of power. In stark contrast, modern supercomputers consume megawatts of power and operate at frequencies of several gigahertz. The human brain's ability to learn and adapt to new situations remains a challenge for traditional computers. Neuromorphic computing, inspired by the human brain, seeks to emulate the way the brain processes information and is expected to be more efficient than traditional computing.

2.2 Background

The concept of Neuromorphic computing was first introduced by Carver Mead in the 1980s. The idea is based on using Very-Large-Scale Integration (VLSI) circuits with analog components to simulate the behavior of neurons in the brain. This stands in stark contrast to traditional computing, which is based on digital logic gates. Such systems use transistors to simulate the behavior of neurons and synapses and communicate with each other using impulses of electrical activity. This inspired the development of Spiking Neural Networks (SNNs) and Hybrid Neural Networks (HNNs).

SNNs are a type of artificial neural network that are inspired by the way the brain processes information. They are made up of spiking neurons that communicate with each other using spikes of electrical activity. These neurons do not have a fixed firing rate and only fire when their membrane potential reaches a certain threshold. SNNs have several advantages and disadvantages compared to traditional artificial neural networks. One of the main advantages is that they are more energy efficient and can achieve higher levels of performance. However, they are also more difficult to train and require more complex hardware. HNNs were first developed to retain the respective advantages of both SNNs and ANNs. In computer vision, for example, researchers use ANNs to extract features from images and then use SNNs to process these features.

There is a tremedous amount of research being done in developing new types of materials and devices that can be used to implement neuromorphic systems. Silicon-based devices have been the mainstay of the semiconductor industry for several decades, but they are reaching their physical limits in terms of size and power consumption. New materials such as phase change memory, memristors, ferroelectric materials and channel-doped biomembrances are being developed. Each of these materials has its own advantages and disadvantages, and researchers are working on developing new devices that can take full advantage of their capabilities.

Over the years, researchers have developed several different types of neuromorphic hardware. Some of the most notable include Neurogrid by Stanford University, SpiNNaker by the University of Manchester, TrueNorth by IBM, and Loihi by Intel. All of these systems are based on the idea of using analog components to simulate the behavior of neurons and synapses. They are also highly parallel and can process vast amounts of information in real time. Loihi, for example, is capable of simulating the behavior of 130,000 neurons across its 128 neuromorphic cores and 3 x86 cores. Extended clusters of up to 768 Loihi Chips operate at 500 watts of power and can simulate the behavior of 100 million neurons.

Tianjic is another neuromorphic chip developed by Tsinghua University in China. Released in 2014, it offered a peak internal storage bandwidth of 610 GBps. It generated a lot of interest in the field of neuromorphic computing due to its high performance, energy efficiency, and software toolchain. The university also developed a robotic system that was controlled by Tianjic and was capable of performing tasks such as object recognition, navigation, and manipulation.

3 SURVEY

Neuromorphic computing is still in its infancy and there are many challenges that need to be overcome before it can be widely adopted. This paper will discuss some of the challenges, solutions and potential area of research in neuromorphic computing that include hardware design, architecture, algorithms, and applications.
Below are some of the key challenges in neuromorphic computing:

- Hardware design: Neuromorphic hardware is still in the early stages of development and there are many challenges to be overcome in terms of design and fabrication. For example, memristors are a promising new technology that could be used to implement neuromorphic systems, but there are still many challenges to be overcome in terms of fabrication and integration with existing technologies.
- Architecture: Neuromorphic systems are highly parallel and require specialized architectures to take full advantage of their capabilities. There are still many challenges to be overcome in terms of designing efficient and scalable architectures for neuromorphic systems.
- Algorithms: Neuromorphic systems require specialized algorithms to take full advantage of their capabilities. There are still many challenges to be overcome in terms of developing algorithms that are efficient and scalable for neuromorphic systems.
- Applications: Neuromorphic systems have the potential to revolutionize many different fields, but there are still many challenges to be overcome in terms of developing applications that can take full advantage of their capabilities.

3.1 Spiking Neural Networks

Spiking Neural Networks (SNNs) are a type of artificial neural network that takes inspiration from the functioning of biological neural networks. Unlike traditional neural networks that transmit information as continuous values, SNNs transmit information as discrete spikes, mimicking the "fire-and-forget" manner of real neurons. This event-driven nature of SNNs opens up novel research directions and has the potential to achieve lower power consumption.

The relevance of SNNs to neuromorphic computing is profound. Neuromorphic computing, by design, seeks to mimic the human brain's structure and function – and SNNs provide a computational model that aligns with this goal. By leveraging the discrete, spike-based communication of SNNs, neuromorphic computing can potentially achieve high computational efficiency, low power consumption, and robust performance, much like the human brain. The advent of SNNs in neuromorphic computing marks a significant shift in the computational paradigm. Traditional computing systems, with their clock-driven synchronous operations and energy-intensive processes, are reaching their physical limits. In contrast, neuromorphic computing systems with SNNs operate asynchronously, leading to potential advantages in speed, power efficiency, and real-time processing capabilities.

This shift towards SNN-based neuromorphic computing is not just a technological evolution; it's a paradigm shift that could redefine what is possible in the realm of artificial intelligence and machine

learning. As we delve deeper into the survey, we will explore how this paradigm is being realized in current research and applications.

In the following subsection, we delve into a curated selection of papers that exemplify the advancements and applications of SNNs in the field of neuromorphic computing. Through these papers, we aim to gain insights into the current state-of-the-art techniques, challenges, and future directions in this rapidly evolving domain, having gained an understanding of SNNs, let us now delve deeper into the field with the examination of a recent and pivotal paper titled *Exploring Neuromorphic Computing Based on Spiking Neural Networks: Algorithms to Hardware.*

This work[2] takes a holistic approach, delving into the intricate interplay between algorithms and hardware architectures in the context of SNNs. At its core, the paper advocates for a paradigm shift in computational design, aiming to emulate the efficiency and adaptability of the human brain.

The cornerstone of this paper lies in the notion of multi-stack optimization, a concept that defies conventional compartmentalization. Here, optimization extends across devices, circuits, and algorithms, intertwining their functionalities in a harmonious symphony. This unified approach endeavors to emulate the brain's natural prowess in information processing, striving for unparalleled efficiency. A particularly intriguing aspect of the paper is its exploration of data representation in spike-trains. Drawing inspiration from neurobiology, the authors propose encoding data as sequences of discrete spikes, mirroring the asynchronous communication observed in biological neurons. This innovative approach holds promise for enhancing the efficiency and scalability of neuromorphic systems.

Moreover, the paper elucidates novel neuron models tailored to capture the nuanced temporal dynamics of biological neurons. By incorporating temporal processing capabilities, these models enable SNNs to more faithfully emulate the brain's intricate information processing mechanisms, unlocking new frontiers in cognitive computing. The authors also tackle the formidable challenge of designing learning algorithms optimized for event-driven dynamical systems. By embracing the inherently asynchronous nature of SNNs, these algorithms adapt dynamically to environmental stimuli, paving the way for robust and adaptive neuromorphic systems. Beyond technical innovations, the paper provides a thought-provoking exploration of the broader challenges and opportunities in neuromorphic computing. It underscores the pressing need for interdisciplinary collaboration and continued research across algorithmic development, hardware design, and system optimization to realize the full potential of SNN-based architectures. In essence, it offers a captivating glimpse into the future of computing. By seamlessly integrating insights from neuroscience, computer science, and engineering, the paper charts a course towards a new era of intelligent computing systems that rival the capabilities of the human brain.

Embarking on a journey from theory to reality, with the next paper we explored how SNNs team up with neuromorphic architectures[1]. These neuromorphic architectures, meticulously crafted to mimic the intricate neural structures of the human brain, provide an ideal substrate for the integration of SNNs. Unlike their traditional counterparts, SNNs operate by transmitting discrete spikes of information, echoing the firing patterns observed in biological neurons. Within this intricate ecosystem, system software frameworks
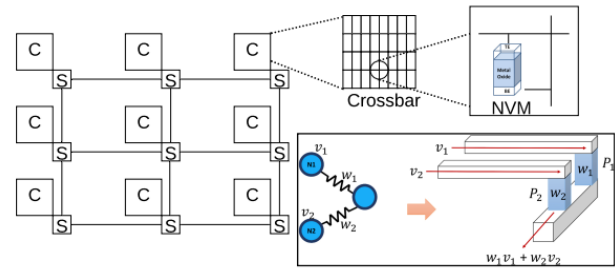

Figure 2: A representative tile-based neuromorphic hardware [47].

Fig. 2. A representative tile-based neuromorphic hardware [1]

emerge as indispensable enablers, furnishing the requisite infrastructure for deploying machine learning applications on neuromorphic hardware. These frameworks serve as orchestrators, seamlessly facilitating the intricate interplay between algorithms and hardware components, thereby streamlining the design, training, and deployment of SNNs on neuromorphic architectures.

The paper under scrutiny embarks on a meticulous exploration of the multifaceted challenges inherent in integrating SNNs with neuromorphic architectures. Navigating through the intricate landscape of system software frameworks, the authors adeptly navigate issues related to performance optimization, energy efficiency, and reliability. Through a nuanced examination, they underscore the urgent imperative for robust frameworks capable of accommodating the ever-mounting complexity of machine learning applications on neuromorphic systems. At the core of their discourse lies the meticulous delineation of system software frameworks tailored to cater to both platform-based design and hardware-software co-design paradigms. By meticulously dissecting the diverse array of existing frameworks, the authors meticulously lay out a roadmap for future endeavors in system software technology for neuromorphic computing. They meticulously outline the contours of prospective investigations, beckoning towards innovative solutions that seamlessly integrate software sophistication with hardware prowess.

In essence, this paper serves as a guiding beacon, illuminating the practical pathways towards realizing the symbiotic relationship between SNNs and neuromorphic architectures. It underscores the indispensable role of robust system software frameworks and charts a definitive course towards the future of system software technology in the realm of neuromorphic computing.

Neuromorphic systems, with their intricate integration of neurons and synapses, are susceptible to faults that can compromise system reliability. As the complexity of these systems grows, so does the vulnerability of their hardware implementations. The intricacies hidden within these accelerators can be exploited by attackers through various side-channel information, posing a significant security risk. However, the inherent spike-based computation in neuromorphic systems offers resilience against transient faults, as the impact of faulty spikes can often be mitigated. Yet, permanent and intermittent faults can still disrupt the functionality of modules within neuromorphic systems, leading to inaccuracies in results. Given these challenges, the development of robust fault-tolerant mechanisms becomes paramount for ensuring the practical implementation and

reliability of neuromorphic systems. This area of research is actively pursued within the field of neuromorphic computing, aiming to address the growing concerns surrounding system reliability and security.

The final paper for this subsection is directly aimed at this pitfall and presents a significant advancement in the field of neuromorphic computing by introducing a groundbreaking scalable neuromorphic fault-tolerant context-dependent learning (FCL) hardware framework[4]. This framework represents a significant advancement, offering resilience against faults while enabling effective learning in context-dependent tasks, even in the presence of hardware failures. Addressing the critical issue of fault-tolerance in neuromorphic systems, the authors provide insights into the development of a framework capable of maintaining robust learning capabilities despite potential hardware faults. This breakthrough not only mitigates the challenges associated with fault-tolerant on-chip learning but also opens up new possibilities for the design and implementation of resilient neuromorphic systems.

Furthermore, the paper delves into the broader landscape of neuromorphic computing, emphasizing the need for continued research in fault-tolerant learning mechanisms and hardware design paradigms. It advocates for the development of robust learning frameworks that can adeptly manage faults, ensuring the reliable operation of neuromorphic systems in real-world applications. It not only addresses present challenges but also sets the stage for future advancements, propelling the development of more dependable and efficient neuromorphic systems.

3.2 Algorithms for Neuromorphic Systems [3]

Considering that neuromorphic architecture are highly parallelizable and inherently scalable, there is a need for specialized algorithms that can take full advantage of their capabilities. One of the key challenges in neuromorphic computing is developing algorithms that are efficient and scalable. Traditional algorithms that are used in artificial neural networks may not be well suited for neuromorphic systems.

Below are some examples of specialized algorithms that have been developed for neuromorphic systems:

- Backpropagation, which is a popular algorithm for training artificial neural networks, may not be well suited for neuromorphic systems due to its high computational complexity and memory requirements. Also the fact that SNNs are inherently stochastic and non-deterministic, makes it difficult to train them using traditional backpropagation algorithms. *Spike-based quasi-backpropagation* adapt the backpropagation algorithm to work with SNNs by using surrogate gradients and smoothed actiobation functions to compute the gradients while maintaining the spiking behavior of the neurons.
- Since ANNs have a long history of research, there are many algorithms and optimizations that have been developed for them. Performing a *mapping* process from ANNs to SNNs is a challenging task, but inherently leads to better performance with respect to energy efficiency and speed. However this mapping process can lead to loss of accuracy due to the conversion of continuous values to spikes.

- *Resevoir Computing* is a popular algorithm used in neuromorphic systems that is based on the idea of using a large network of recurrently connected neurons to perform computations. The network does not involve any training of the SNN components. It uses sparse connectivity and random weights to cast inputs to a higher dimensiontality space. The output is then read out from the network using a linear classifier. Resevoir computing has been shown to be highly efficient and scalable and has been used in a wide range of applications.
- *Evolutionary Approaches* is another popular algorithm used in neuromorphic systems that is based on the idea of using multiple generations of populations and selecting the best individuals to evolve the population. This approach can be used to optimize the parameters of the SNNs and is highly efficient and scalable as it does not require any differentiablity in activation functions nor rely on any particular network structure

Neuromorphic Computing can be used in a wide range of applications, including non-ML tasks such as signal processing, graph theory, and optimization. For example, during the pandemic in 2020, researchers used neuromorphic computing to develop a model that could predict the spread of COVID-19.

However there are still many challenges to be overcome in terms of developing algorithms that are efficient and scalable for neuromorphic systems. There is not yet a ML algorithm that substantially outperforms traditional ML algorithms in terms of accuracy, speed, and energy efficiency. This leads to the argument that neuromorphic computing is useful for its low power computing abilities.

Another key challenge is the lack of a standard benchmark for evaluating neuromorphic algorithms. There are many different neuromorphic systems available, each with its own architecture and capabilities. This makes it difficult to compare the performance of different algorithms across different systems. There is a need for a standard benchmark that can be used to evaluate the performance of neuromorphic algorithms and systems.

3.3 CerebelluMorphic[6] and BiCoSS[5]

BiCoSS and CerebelluMorphic are two neuromorphic systems that have been developed in recent years. Considering both these papers were published in 2022 and by the same group, they share some common design principles.

BiCoSS uses scalable hierachical heterogeneous multicore architecture with localized memory. By determining neuron models for multiple cognitive organs such as the cerebellum, hippocampus, and basal ganglia, and many more, BiCoSS can construct biological cognitive systems. It has 3 main design principles:

- Independency Reconfigurable Population Process: Using FPGAs, the system can be reconfigured to perform different tasks of different complexity. This allows for the system to be used in a wide range of applications.
- Stochastic Neural Heterogenity: BiCoSS presents an efficient alogrithm for optimizing temporal and spatial heterogenity in the neural network, meaning that the response of the neurons can be optimized with respect to the intensity of the input. This allows for the system to be more energy efficient and
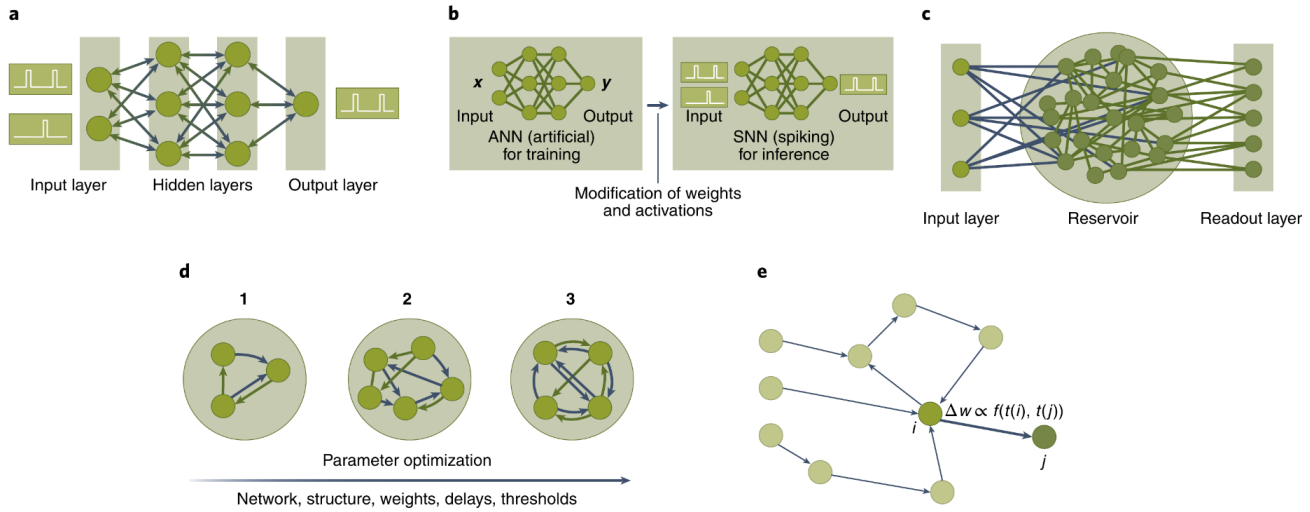
Fig. 3. [3] (a): Spike-based Quasi-backpropgation (b): Mapping ANNs to SNNs (c): Resevoir Computing (d): Evolution of SNNs over time (e): Time Parameter in spikes
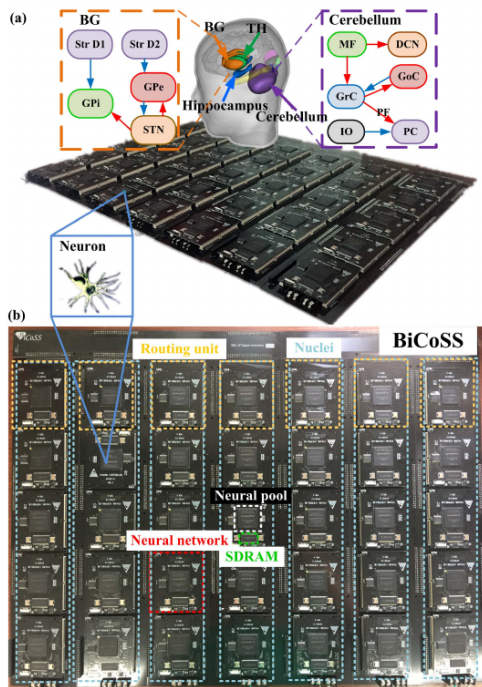

Fig. 4. [5] (a): Brain Cognitive Functions implemented in BiCoSS (b): BiCoSS System

faster. It also allows for the system to be more robust to noise and other disturbances.

- Routing Scheme for Hybrid Neural information: Different synapses have different spiking activities, and synaptic dynamics. 1 bit calculations are realized by multiplexer. This

allows for the system to be mimic the behavior of various cognitive organs.

CerebelluMorphic is a neuromorphic system that is inspired by the cerebellum, which is a part of the brain that is responsible for motor control and coordination. The cerebellum is highly efficient and is capable of processing vast amounts of information in real time. CerebelluMorphic is based on the idea of using a large network of spiking neurons to simulate the behavior of the cerebellum.

A key challenge in neuromorphic computing is acheiving further brain complexity. The human brain is highly complex and is capable of processing vast amounts of information in parallel. These two architecture are just the beginning of realizing large-scale brain models and lay the groundwork for further research in the field of neuromorphic computing.

3.4 Memristors[7]

Memristance is a property of an electronic component that is capable of changing its resistance in response to the amount of current that flows through it. Materials that exhibit memristance are called memristors. A memristor is a two-terminal electronic device whose resistance depends on the voltage applied to it and the length of time that voltage has been applied. This property allows memristors to retain information even when the power is turned off, making them ideal for memory storage. Memristors have been proposed as a possible solution to the challenges of neuromorphic computing as they can adjust their resistance based on the amount and timing of the voltage applied across them, mimicking the plasticity of biological synapses and enabling learning and adaptation considering that transistor technology, as it is currently implemented, is reaching its physical limits in terms of size and power consumption.

Memristors were first proposed by Leon Chua in 1971, but it wasn't until 2008 that the first practical memristor was developed by HP Labs. Since then, memristors have been the subject of intense
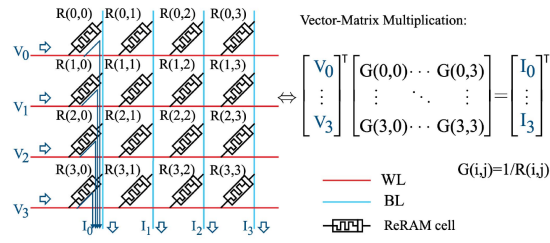
Fig. 5. Memristor Crossbars - based Matrix Multiplication [7]

research and development, with many companies and research institutions working on developing memristor-based devices.

Memristors have several advantages over traditional transistors.

- **Non-volatile memory**: Memristors retain their state even when the power is turned off. This makes them ideal for use in memory devices.
- **Low power consumption**: Memristors consume less power than traditional transistors. This is because they do not require a constant flow of current to maintain their state.
- **High density**: Memristors can be packed more densely than traditional transistors. Memristors can also store multiple bits of data per cell, further increasing their density. Multiple sturctures have been proposed for memristors, such as the crossbar array, which allows for high density and parallel processing, or cubes of memristors, which allow for 3D integration.

However, memristors also have some disadvantages.

- **Reliability issues**: Memristors can be prone to errors in conductance leading to data corruption. This can be a problem in applications that require high reliability. Thus, effective solutions to mitigate these errors are needed.
- **Fabrication challenges**: Memristors are still in the early stages of development and there are still many challenges to be overcome in terms of fabrication and integration with existing technologies.

Memristor Crossbars are of much relevance in the field of neural networks. Due to their structure, they can be used to implement vector matrix multiplication, which is a key operation in neural networks. Figure 5 shows a simple crossbar array with memristors. On applying a voltage to the input lines, the memristors change their resistance based on the voltage applied. The output lines then read the resistance of the memristors, which is proportional to the dot product of the input and weight vectors. This allows for parallel processing of multiple inputs and weights, making it ideal for neural networks.

Using memristors in neuromorphic computing instead of traditional transistors has several advantages.

- State retention: Memristors retain their state even when the power is turned off. This makes them ideal for use in memory devices.
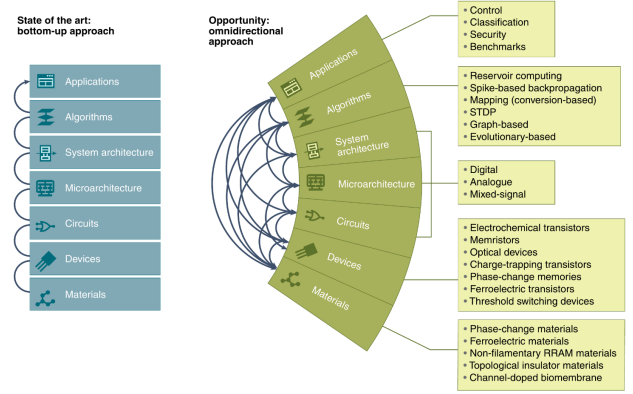

Fig. 6. Opportunities in Neuromorphic Computing [3]

- State representation: Conductance of a memristor can be used to represent the synaptic weight between two neurons. This allows for the implementation of synapses in neuromorphic systems.
- As mentioned earlier, memristors can be used to implement vector matrix multiplication, which is a key operation in neural networks. This allows for the implementation of neural networks in hardware, which can be much faster and more energy efficient than software implementations.

Thus, memristors have the potential to revolutionize the field of neuromorphic computing and enable the development of more efficient and powerful neuromorphic systems once the challenges of fabrication and reliability are overcome.

4 CONCLUSION

Neuromorphic computing is a promising new computing paradigm that has the potential to revolutionize the way we think about computing. By mimicking the way the brain processes information, neuromorphic systems are expected to be more efficient and powerful than traditional computing systems.

Over the years, researchers have developed several different types of neuromorphic hardware, architecture, and algorithms. Considering that the human brain is capable of processing vast amounts of information in parallel, neuromorphic systems are highly parallel and can process vast amounts of information in real time. Several large-scale neuromorphic systems have been developed and are available for research and development.

However, there are still many challenges that need to be overcome before neuromorphic computing can be widely adopted. Various functions of the brain are still not fully understood, and reverse engineering the brain is a complex and challenging task.

There are also many challenges in software-hardware co-design. Currently most neuromorphic research is focused on individual aspects of neuromorphic systems, such as hardware design, architecture, algorithms, and applications. There is a need for more research that integrates these different aspects and develops holistic solutions that take full advantage of the capabilities of neuromorphic systems.

Integration of such systems is quite difficult because they encompass a wide range of technologies and disciplines. There is a need for more collaboration between researchers in different fields and more interdisciplinary research that can address the challenges of neuromorphic computing.

By addressing these challenges, researchers can help to unlock the full potential of neuromorphic computing and usher in a new era of computing.

5 ACKNOWLEDGEMENTS

REFERENCES

[1] Phu Khanh Huynh, M. Lakshmi Varshika, Ankita Paul, Murat Isik, Adarsha Balaji, and Anup Das. Implementing spiking neural networks on neuromorphic architectures: A review, 2022.

[2] Nitin Rathi, Indranil Chakraborty, Adarsh Kosta, Abhronil Sengupta, Aayush Ankit, Priyadarshini Panda, and Kaushik Roy. Exploring neuromorphic computing based on spiking neural networks: Algorithms to hardware. *ACM Comput. Surv.*, 55(12), mar 2023.

[3] Catherine D. Schuman, Shruti R. Kulkarni, Maryam Parsa, J. Parker Mitchell, Prasanna Date, and Bill Kay. Opportunities for neuromorphic computing algorithms and applications. *Nature Computational Science*, 2(1):10–19, 2022.

[4] Shuangming Yang, Jiang Wang, Bin Deng, Mostafa Rahimi Azghadi, and Bernabe Linares-Barranco. Neuromorphic context-dependent learning framework with fault-tolerant spike routing. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):7126–7140, 2022.

[5] Shuangming Yang, Jiang Wang, Xinyu Hao, Huiyan Li, Xile Wei, Bin Deng, and Kenneth A. Loparo. Bicoss: Toward large-scale cognition brain with multigranular neuromorphic architecture. *IEEE Transactions on Neural Networks and Learning Systems*, 33(7):2801–2815, 2022.

[6] Shuangming Yang, Jiang Wang, Nan Zhang, Bin Deng, Yanwei Pang, and Mostafa Rahimi Azghadi. Cerebellumorphic: Large-scale neuromorphic model and architecture for supervised motor learning. *IEEE Transactions on Neural Networks and Learning Systems*, 33(9):4398–4412, 2022.

[7] Xiaoxuan Yang, Brady Taylor, Ailong Wu, Yiran Chen, and Leon O. Chua. Research progress on memristor: From synapses to computing systems. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 69(5):1845–1857, 2022.