A Review of Spiking Neural Networks

Junyi Wang*

School of Communication Engineering, Jilin University- Lambton College, Changchun, Jilin Province, China, 130012

ABSTRACT: Spiking neuron network (SNN) attaches much attention to researchers in neuromorphic engineering and brain-like computing because of its advantages in Spatio-temporal dynamics, diverse coding mechanisms, and event-driven properties. This paper is a review of SNN in order to help researchers from other areas to know and became familiar with the field of SNN or even became interested in SNN. Neuron models, coding methods, training algorithms, and neuromorphic computing platforms will be introduced in this paper. This paper analyzes the disadvantages and advantages of several kinds of neural models, coding methods, learning algorithms, and neuromorphic computing platforms, and according to these to propose some expected development, such as improving the balance between bio-mimicry and cost of computing for neuron models, compounding coding methods, unsupervised learning algorithms in SNN, and digital-analog computing platform.

1. INTRODUCTION

Due to Moore's Law, the performance of traditional computers increased exponentially. However, after the failure of Dennard's Law, it was thought that Moore's Law would fail, so it is a must for researchers to rethink how to develop new computer architectures [4]. Traditional computer systems utilizing the Von Neumann architecture have limitations in power consumption: the data transmission between the separated central processing units and the memory contributes approximately 50% of the power consumption of the system [1]. Also, conventional computers can't generalize and learn themselves [1]. There are many landmark neural network models appeared since MP artificial neuron model was proposed by McCulloch and Pitts in 1943, such as the Perceptron, the Hopfield network, the Boltzmann Network, the Back Propagation (BP) Network, Convolutional Neural Networks (CNN), Spiking Neural Network (SNN), Deep Belief network, and Deep Neural Network (DNN) [5,6,7,8,9,10,11]. Spiking neural networks are a new generation of artificial neural network models inspired by biology. Spiking neural networks has the following advantages: 1) fast speed and better real-time performance, 2) high performance in computing with low power consumed, 3) more bio-mimetic, 4) enabling neuromorphic hardware systems utilizing SNN's own a better robustness [1, 2].

This paper focuses on the introduction of neuron models, coding methods, training algorithms, and neuromorphic platforms, and discussed their advantages and disadvantages. This paper aims to attract researchers from different fields, which will promote the development of this field. Also, it wishes to help researchers who are unfamiliar with SNN to have a basic understanding of SNN. This paper first analyzes the bio-mimicry and cost of compute of different kinds of neuron models, and then explains the commonly used coding methods. After that, it discusses the supervised and unsupervised learning methods of SNN, and at the end of the discussion, it mentions the digital and digital-analog neuromorphic computing platform. This paper can help people who are unfamiliar with this field to know the basic knowledge quickly and help them be interested in this area.

2. BIOLOGICAL NEURONAL PRINCIPLES AND MAJOR NEURON MODELS

2.1. Biological Neuronal Principles

A neuron is constituted by a dendrite, soma, and axon. Dendrite transmits the signals received from other dendrites to soma which can be seen as a central processor, and then soma transmits the signals along with the axon when the membrane potential accumulates to its threshold potential to the synapse without any attenuation. The synapse is at the end of the axon and connected to other neurons [12].

2.2. Neuron Models

In SNN, there are some models that are commonly used: Hodgkin-Huxley model (HH model), integrate and fire model (IF model, leaky integrate and fire model (LIF model), Izhikevich model, and spike response model (SRM). HH model, LIF model, and Izhikevich model will be introduced in this section.

*Corresponding author. Email: 1151890419@qq.com

2.2.1. HH Model

Despite the fact that the dynamic changes of iron channels like Na+ and K+ can be expressed by HH model in a precise way, researchers find the expression of HH model is complex, which leads the Cumbersome calculations. Researchers also find the fact that stimulating the behaviors of HH model in 1ms needs 1200 floating-point operations [12,13,14,15].

2.2.2. LIF Model

LIF model simplifies the process of action potentials by ignoring the dynamic changes of ion channels. LIF only focuses on the changes of membrane potential at a macroscopic level and its computational procedure is easier. However, LIF can't stimulate more neuronal behaviors except for leakage, accumulation, and threshold excitation [12,13,14].

2.2.3. Izhikevich Model

The Izhikevich model tries to balance biological rationality and computational efficiency. Izhikevich model can stimulate more than 20 neuronal behaviors and can stimulate the behaviors of neurons in 1ms only with 13 times floating-point operations, which is much less than HH model.

All in all, it is important for neuron model to have a balance between the degree of bio-mimicry and the low computational cost which determines whether the neuron can be integrated on a large scale. Finding an appropriate neuron model is also a current research problem [12,14,15].

3. CODING METHODS

There are two processes in neural information encoding: extracting characteristics and generating pulse sequences. The receiving information is extracted or represented by the perceptual neural system first, which is corresponding to the extracting feature in machine learning [16]. In the pulse sequence generation, there are some widely used coding methods: rate coding, temporal coding, population coding and address event representation.

3.1. Rate Coding

The frequency of pulse delivery was influenced by the strength of the stimulus, and the two were positively correlated [1,12]. An application of rate coding is the conversion of a well-trained Artificial neural network (ANN) to an SNN where the pulse release rate can be equated to the continuous output value of the ANN. However, the information that varies fast is not accurately described by the rate encoding [16].

3.2. Temporal Coding

In temporal coding, information is carried by impulse timed precisely. Researchers proposed an algorithm called latency coding. The generating time of impulse is negatively correlated with the intensity of the stimulus [16,17]. However, using temporal coding directly can lead to extracting a sheer volume of unimportant information [16].

3.3. Population Encoding

In the neuron system, individual neurons are susceptible to interference, while population encoding can increase accuracy. In population encoding, the information is expressed by multiple neurons together. Researchers proposed a population vector model in which the actual moving direction of primates is determined by a weighted sum of the intensity of multiple neuron activity to a specific direction [12,18].

3.4. Address Event Representation (AER)

In AER, Each neuron has an address that will be transmitted to a shared asynchronous bus when the neuron is ready to send a pulse [1].

There is no consensus on the method of neuron coding, and actually, it's possible for neurons to take different coding methods in cooperation or use different coding methods on various occasions and parts of the brain. Therefore, the mix-use of code methods should be supported in a generic SNN [12].

4. LEARNING ALGORITHMS

Core training algorithms and techniques haven't been proposed currently.

Whether or not using tags in training, the algorithms, on the whole, can be classified into unsupervised and supervised learning algorithms. The main principles of unsupervised learning are Hebb principles and Spiking Timing Dependent Plasticity (STDP). For supervised algorithms, challenges exist in application of BP algorithms where the weight transport problem leads to a contrary between the direction of the information transmitted in synaptic and the adjustment of weights in BP algorithm by feedback. Also, the discrete binary signals transmitted in SNN can't be differentiated, and the activation function in pulse form brings difficulties to applying the gradient-based algorithms.[12].

4.1. Unsupervised Learning Algorithms

4.1.1. Biological Principles

Hebb proposed that when the B cell is stimulated repeatedly by A, the efficiency for B being stimulated successfully will increase. In STDP, the coupling strength increases when the presynaptic neuron release pulse before postsynaptic, otherwise the coupling strength decreases [12, 13].

4.1.2. The Disadvantages of STDP

Despite the fact that STDP has a higher degree of bionic, it is currently not suitable for multi-layer structures. Researchers proposed utilizing traditional ANN to train the SNN because the performances of CNN are much better than the SNN in classification tasks [13, 16, 19].

4.2. Supervised Learning Algorithms

Researchers combined the STDP with other weight adjustment methods including gradient descent and Widrow-Hoff rules to develop other supervised algorithms [16]. However, researchers have challenges with backpropagation algorithms (BP algorithms). On the one hand, It is a contrary between the direction of the information transmitted in synaptic and the adjustment of weights in BP algorithm by feedback, which is called the weight transport problem [12].

4.2.1. SpikeProp

With the error back-propagation in ANN, researchers proposed the SpikeProp where the gradient descent is calculated according to the error minimization principle. However, back-propagation gives rise to low learning efficiency. It is a must to increase the accuracy of the algorithm by adding the number of hidden-layer, which in turn leads to the learning process not being roust anymore [13, 16, 20].

4.2.2. Remote Supervised Method (ReSuMe)

STDP and remote supervision are combined by ReSuMe to minimize the gap between the output and the target spiking in a situation where the gradient is not concerned. Signals being supervised remotely can't be transmitted directly to trained neurons, and it determines the variation of synaptic changes with the presynaptic neurons [12, 13].

4.2.3. ANN-Converted SNN

Due to the algorithm in SNN is not mature as it in ANN, researchers proposed transforming ANN to SNN. That means Utilizing firing rate encoding to transform the well-trained deep neuron network in ANN into SNN. ANN-converted Algorithm utilizes back-propagation algorithms to avoid the challenges in training SNN directly, and there is no huge gap between the current ANN and final SNN. Although the experience and achievement in traditional ANN can be transformed into SNN directly by Utilizing ANN-Converted SNN, it composing constraints to initial ANN and make SNN spends longer time to stimulate forward reasoning. Such changes decrease the performance of the original ANN, and also give rise to a contrary that utilizing SNN can't reduce the delay and energy consumed [12, 16].

5. NEUROMORPHIC COMPUTING PLATFORM

Neuromorphic computing platforms usually can be expanded from computational cores and chips to systems. The data transmission between computing cores and chips is supported by a routing network, and the transmission can take place in the core. Compared with the traditional Van Neumann, each computing cores have its own space, and such a decentralization operation mode provides high parallelism and access efficiency. In the neural computing platform, neurons and their corresponding synaptic functional behaviors are simulated by the computing core. To change the network function to complete the learning process, the weight of the synaptic is needed to be changed by the update of synaptic learning [12].

5.1. Offline Learning Mixed Platform

The problem that the transmission of long-range data between cores and chips is tough to be done accurately exists in routing networks with analog circuits. Therefore, the computing cores often utilize analog circuits, and the routing network uses digital circuits [12].

5.1.1. Neurogrid

In Neurogrid, each chip contains 256×256 neurons, and the 16 chips in the circuit board are connected by a tree routing network. Neurogrid consumes 3.1 W with millions of neuron networks simulated [12, 21].

5.1.2. DYNAPS

DYNAPS uses a 2D grid routing topology between the chips and a tree-structured routing topology between the computing cores in chips. In computing cores, packet transmission is achieved by multicast and tag matching, and the content addressing memory (CAM) is responsible for the matching process. Low latency of tree topology and low bandwidth requirements of mesh topology can be combined in such structure. In DYNASP, each circuit board contains 9 chips which have 4 computing cores with 256 neurons per core [12].

5.1.3. Neuromorphic Chips Utilizing Memristor

The computation and storage can be integrated by memristor array because of its high-density nature, so the speed and the efficiency of matrix vector can be improved significantly. However, due to the leakage of the controllability, it is tough to integrate the memristor array currently [12].

5.2. Offline Digital Platform

Although analog circuit makes implementation of neural dynamics easily, it is defective in stability, programmability, and simulation. Therefore, digital neuromorphic computing platforms have become popular

in the industry [12].

Four thousand and ninety-six computing cores are contained in a chip of TrueNorth, and the cores are connected by 2D grid routing. Every computational core contains 256 neurons and a 256 × 256 size synaptic array. It only computes when there are pulse events as input because it utilizes both synchronous event-driven and asynchronous event-driven, which enables it only consumes tens or hundreds of milliwatts when it simulates the SNN with millions of neurons [12].

5.3. Online Learning Mixed Platform

Update of parameters is supported in SNN during the operation of the SNN [12]. In BrainScales, super-threshold properties are used for neural dynamics simulation, and BrainScales can operate at a 1,000 to 100,000 times more speed than the biological brain. A wafer of BrainScales contains 352 chips with 352 × 512 neurons, and Field Programmable Gate Array (FPGA) is equipped for routing communication between digital boards. The system consumes approximately 1 kw [12, 22].

5.4. Online Learning Digital Platform

5.4.1. SpiNaker

In SpiNaker, there are 18 dynamic random access memory (ARM) processor cores with approximately 1000 neurons simulated and 182 megabytes off-chip Dynamic Random Access Memory (DRAM) where synaptic parameters are stored [12].

5.4.2. Loihi

In Loihi, there are 128 computing cores with up to 1024 neurons and 16 megabytes of synaptic capacity, and the entire chip integrates 33 megabytes of on-chip SRAM memory [12, 23].

6. CONCLUSION

The destination and pointer of Artificial intelligence is always human brain. This review aims to help readers know the SNN, and its neuron models, coding methods, training algorithms, and neuromorphic computing platform. For Neuron models, it is important to balance the degree of bio-mimicry and the low computational cost. For coding methods, it is significant to find generic SNN coding methods. For algorithms, researchers could pay more attention to direct training algorithms andexplore more characteristics of SNNs. This paper only mentions the representative neuron models, coding methods, algorithms, and platforms of SNN, and doesn't introduce much theory and formula of SNN. Therefore, it can be improve by adding information mentioned above. Compared to the digital platform, the analog neuromorphic computing platform has lower programmability, so researchers can try to improve it by

optimizing the hardware. Brain-like technology will push development in some hot areas such as image recognition, speech recognition, pilotless, and more filed. It is possible to drive a new round industry revolution.

REFERENCES

1. WANG XQ, ZENG H, HAN DM, LIU Y, LU F, Spiking neural network-based brain-like computing, JOURNAL OF BEIJING UNIVERSITY OF TECHNOLOGY,

2. ZHANG TL, XU B. Research Advances and Perspectives on Spiking Neural Networks. CHINESE JOURNAL OF COMPUTERS, 2021, 44(9): 1767-1785. [DOI: 10.11897/SP.J.1016.2021.01767]. 45(12): 1278-1286. [DOI: 10.11936 /bjutxb2018100018].

3. Yu S. From Brain Networks to Artificial Intelligence--Opportunities and Challenges of Brain-Like Computing. SCIENCE AND TECHNOLOGY REVIEW, 2016, 34(7): 75-77.

4. Huang TJ, Yu ZF, Liu YJ. Brain G like Machine: Thought and Architecture. Journal of Computer Research and Development, 2019, 56(6): 1135-1148. [DOI:10.7544/issn1000-1239.2019.20190240]

5. Mcculloch W S, Pitts W. A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 1943, 5(4): 115-133.

6. Rosenblatt F. The perceptron, a perceiving and recognizing automaton project para. Cornell Aeronautical Laboratory, 1957.

7. Hinton G E, Sejnowski T J. Optimal perceptual inference//Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition. Washington, D.C, USA, 1983.

8. Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors. Nature, 1986, 323(6088): 533-536.

9. Lecun Y, Boser B, Denker J S, et al. Backpropagation applied to handwritten zip code recognition. Neural Computation, 1989, 1(4): 541-551

10. Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets. Neural Computation, 2006, 18(7): 1527-1554.

11. Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks. Communications of the Acm, 2017, 60(6): 84-90.

12. Hu YF,Li GQ,Wu YJ,et al. A review of research progress on impulsive neural networks[J]. Control and Decision Making, 2021, 36(1): 1-26.

13. ZHANG HG, XU GZ1, GUO JR, GUO L. Brain-like pulsed neural networks and their neuromorphic chips Research Review. Journal of Biomedical Engineering, 2021, 38(5): 986-994.

14. Xu LF, Li CD, Chen L. Contrastive analysis of neuron model. Acta Physica Sinica, 2016, 65(24): 240701-1-240701-12. [DOI: 10.7498/aps.65.240701].

15. Hodgkin A L, Huxley A F. A quantitative description of membrane current and its application to conduction and excitation in nerve. J Physiol, 1952, 117(4): 500-544.

16. Huang T J, Shi L P, Tang H J, Pan G, Chen Y J, Yu J Q. Research on multimedia technology 2015—advances and trend of brain-like computing [J]. Journal of Image and Graphics, 2016, 21(11): 1411-1424. [DOI: 10. 11834 /jig. 20161101].

17. Hopfield J J. Pattern recognition computation using action potential timing for stimulus representation [J]. Nature, 2002, 376 (6535): 33-36. [DOI: 10.1038 /376033a0]

18. Georgopoulos A P, Schwartz A B, Kettner R E. Neuronal population coding of movement direction[J]. Science, 1986, 233(4771): 1416-1419.

19. Cao Y Q, Chen Y, Khosla D. Spiking deep convolutional neural networks for energy-efficient object recognition [J]. International Journal of Computer Vision, 2015, 113(1): 54-66. [DOI: 10.1007 /s11263-014-0788-3].

20. Bohte S M, Kok J N, Poutr'e H L. SpikeProp: backpropagation for networks of spiking neurons [C]/ /The of 8th European Symposium on Artificial Neural Networks, Bruges, Belgium, 200

21. Benjamin B V, Gao P, McQuinn E, et al. Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations[J]. Proceedings of the IEEE, 2014, 102(5): 699-716

22. Schemmel J, Brüderle D, Grübl A, et al. A wafer-scale neuromorphic hardware system for large-scale neural modeling[C]. 2010 IEEE International Symposium on Circuits and Systems. Paris: IEEE, 2010: 1947-1950.

23. Davies M, Srinivasa N, Lin T H, et al. Loihi: A neuromorphic manycore processor with on-chip learning[J]. IEEE Micro, 2018, 38(1): 82-99.