xBrain: Brain-Like Computing for Explainable Brain-Computer Interfaces

***Abstract*—The extraction and classification of repeated brain activity patterns using brain-computer interfaces (BCI) could be the key to advancing our understanding of the brain. Conventional approaches to BCI incorporate deep neural networks (DNNs) to yield effective classification results in a black box. However, this approach ignores the three stages of BCI processing, including preprocessing electrical signals from the brain, spike-sorting, and template-matching, and makes explainability challenging. This paper proposes an end-to-end neuromorphic-driven framework for efficient, transparent, and explainable BCI processing. Preliminary tools for generating rate-based and temporal-based spiking neural networks (SNNs) implementations drive this neuromorphic-based BCI framework, yielding efficient and explainable BCI processing methodology. Future research directions are also outlined.**

***Index Terms*—brain-computer interface, unary computing, temporal neural networks, neuromorphic computing**

I. Introduction

Brain-computer interfaces (BCI) are systems that measure specific patterns of brain activity to translate them into device control signals, advancing our understanding of the human brain [2]. Progress in BCI research has enabled various applications, including human-machine interactions [19], personal entertainment [13], restoring lost motor functions [5] [9], and more. State-of-the-art applications require very low-power BCI systems (limited to a few milliwatts of power) to be surgically implanted as they collect far higher fidelity neural signals than wearable BCIs [18]. BCI hardware generally consists of three main stages - (i) *spike sampling*: the continuous sampling and analog processing of electrophysiological spikes using neuroprobes [1], (ii) *spike sorting*: converting the analog signals into time-ordered digital binary data, and (iii) *template-matching*: pattern matching of the sorted signals with indicator signals stored in memories.

While DNN-specific approaches encapsulate both the spike-sorting and template-matching stages in the processing, yielding promising results [7] [10], they convert the pipeline into a black box with no explainability for gaining valuable insights into the systems. Further, neuromorphic-based approaches have already been proposed, but they target individual process stages [3] [24] [21]. Our approach contrasts with the previous works by proposing a fully end-to-end framework, xBrain, for explainable BCI processing driven by neuromorphic-based approaches across the entire BCI pipeline.

II. Background

A. Unary Computing

Unary computing is a paradigm that substitutes the need for multiple parallel bits to represent data with just a single-bit stream. It manifests in two encoding forms: *rate-based unary* and *temporal-based unary*. Data representation in rate-based unary coding is based on the frequency of 1s and 0s within a bitstream, with the number of 1s in the bitstream proportional to the data value. In contrast, in temporal-based unary encoding, the data is represented in a sequence of 1s followed by a sequence of 0s, with the number of consecutive 1s representing the value. Rate-coding results in approximate computation for arithmetic operations, while temporal-coding is precise for relational operations [22].

B. Neuromorphic Computing

a) Rate-coded: Traditionally rate-coded neural networks like ANNs operate on continuous activation values, however, this approach has the drawback of increased computational energy. The framework of SNNs reflects neuronal communication, basing spiking potential on temporal patterns [23]. Their merits yield in their use as applications for interpreting and preparing electrophysiological (EP) and electroencephalogram (EEG) for template matching. The accuracy of SNNs without the addition of training algorithms are not as accurate as ANNs. Backpropagation (BP) algorithm used to train SNNs increases the computational cost and are not reflective of biological systems. Hence, we include the use of unsupervised spike-timing-dependent plasticity (STDP) algorithm which are biologically plausible to increase the accuracy of the SNNs [8]. STDP trains the convolution-pooling layers of SNNs, so the most pertinent features from the dataset are learned [12].

b) Temporal-coded: Temporal-coded SNNs are inspired by the brain's compute paradigm. Recent research on temporal neural networks (TNNs) [16], [17] [14], a special class of SNNs, use space-time algebra [15] and precise spike timings to represent and process information. Unlike DNNs, TNNs employ simple feed-forward processing based on spikes and their timing relationships. Furthermore, TNNs are capable of online

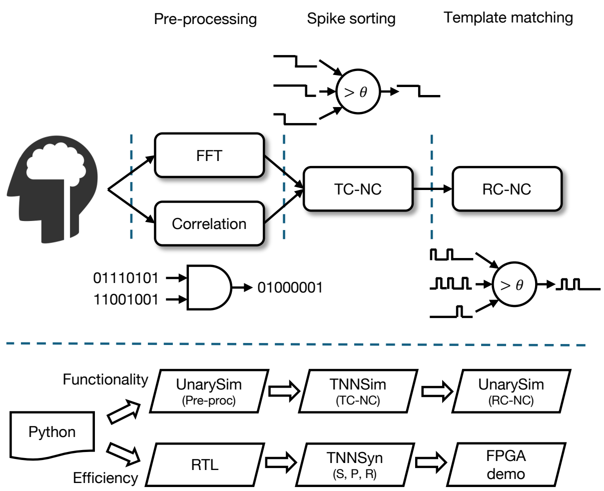

Figure 1: xBrain framework overview. TC-/RC-NC means temporal-/rate-coded neuromorphci computing.

continuous learning using biologically plausible local learning algorithms called Spike Timing Dependent Plasticity (STDP), unlike backpropagation-fueled DNNs that have a strict bifurcation between training and inference phases. TNNs have been demonstrated to be highly feasible for applications such as time-series clustering [6]. Recently, an automated toolchain, TNNGen, for realizing highly efficient TNN columns [20] has been proposed and implemented. We wish to leverage these developments for BCI processing due to the high synergy between the two research domains and their explainability compared to DNN-specific approaches.

III. Proposed Framework

A. BCI Pipeline

Similar to traditional three-stage BCI processing pipelines, as mentioned in Section I, we propose incorporation of neuromorphic-driven approaches for each stage. We show the proposed BCI pipeline at the top half of the Figure 1, including pre-processing, spike-sorting, and template matching stages. These stages are sequential and can be selectively disabled if the applications allow so. The first stage uses rate-coded unary computing to pre-processing the raw brain signals. The candidate algorithms can be FFT or correlation, which are popular in real-world chips [11]. Unary computing computes on bitstream data with extremely simple hardware (e.g., an AND gate for multiplication) [22]. The processed data are fed to TNN to perform spike sorting, which cluster relevant signals together. The TNN will offer an ultra low-power execution environment due to minimized switching activity in the signals. Finally, rate-coded SNN will perform template matching with much higher efficiency than conventional approaches, which require hundreds of MB data storage [1].

B. End-to-End Framework

We propose a neuromorphic-driven explainable approach for our BCI pipeline. The end-to-end toolchain will encompass rapid software design-space exploration for various BCI applications, as well as the hardware generation for the software models. A Python-to-RTL converter using Python libraries such as PyVerilog [] will be utilized to convert the software models to their equivalent RTLs, with custom TCL scripts automating the process flow invoking relevant EDA tools. Such an approach has already been implemented for generating TNNs in the form of TNNGen [20], leading to highly energy-efficient hardware implementations for TNNs. We will extend the tool to further support rate-based SNN generation as well, in both the software and hardware stacks. Further, UnarySim [22], a publically available simulation framework for unary computing and rate-based neuromorphic computing can be leveraged and incorporated into TNNGen, and hardware support can be extended for it.

IV. Evaluation Methodology

A. Metric

The evaluation metrics include accuracy, latency, and power efficiency, which are pivotal in BCI applications, especially in mobile settings. As we target explainability as a distinct feature, we need to evaluate the result metrics per stage. As multiple novel computing schemes are used across multiple stages, the design space increases exponentially. We will explore the design space and later narrow down to the Pareto front.

B. Baseline

We will use existing BCI frameworks for spike sorting as baselines, e.g., SpikeInterface [4]. Existing frameworks are mainly running on CPU or GPUs, from which we will measure the baseline results. Though there exist works for partial BCI stages, it is impossible to reimplement them and organize them into an end-to-end design, and we are not using these works as baselines. For our xBrain, we will extract the results for functionality based on both software and hardware simulation. To evaluate the efficiency with in the gigantic design space, we will design analytical models. A small scale of RTL implementations will be used to validate our analytical model.

V. Future Research Direction

A complete xBrain automated pipeline, spanning software to hardware, driven by neuromorphic principles incorporating unary computing, rate-based and temporal-based SNNs, is being developed. Researchers can leverage it to create highly efficient and explainable BCIs targeting various applications.

REFERENCES

[1] A. M. Abdelhadi, E. Sha, C. Bannon, H. Steenland, and A. Moshovos, "Noema: Hardware-efficient template matching for neural population pattern detection," in *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture*, 2021, pp. 522–534.

[2] R. A. Andersen, T. Aflalo, L. Bashford, D. Bjånes, and S. Kellis, "Exploring cognition with brain–machine interfaces," *Annual Review of Psychology*, vol. 73, pp. 131–158, 2022.

[3] J. M. Antelis, L. E. Falcón *et al.*, "Spiking neural networks applied to the classification of motor tasks in eeg signals," *Neural networks*, vol. 122, pp. 130–143, 2020.

[4] A. P. Buccino, C. L. Hurwitz, S. Garcia, J. Magland, J. H. Siegle, R. Hurwitz, and M. H. Hennig, "Spikeinterface, a unified framework for spike sorting," *eLife*, vol. 9, p. e61834, nov 2020.

[5] S. Chandrasekaran, M. Fifer, S. Bickel, L. Osborn, J. Herrero, B. Christie, J. Xu, R. K. Murphy, S. Singh, M. F. Glasser *et al.*, "Historical perspectives, challenges, and future directions of implantable brain-computer interfaces for sensorimotor applications," *Bioelectronic medicine*, vol. 7, no. 1, pp. 1–11, 2021.

[6] S. Chaudhari, H. Nair, J. M. Moura, and J. P. Shen, "Unsupervised clustering of time series signals using neuromorphic energy-efficient temporal neural networks," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021.

[7] F. Fahimi, Z. Zhang, W. B. Goh, T.-S. Lee, K. K. Ang, and C. Guan, "Inter-subject transfer learning with an end-to-end deep convolutional neural network for eeg-based bci," *Journal of neural engineering*, vol. 16, no. 2, p. 026007, 2019.

[8] W. Guo, M. E. Fouda, A. M. Eltawil, and K. N. Salama, "Neural coding in spiking neural networks: A comparative study for robust neuromorphic systems," *Frontiers in Neuroscience*, vol. 15, p. 638474, 2021.

[9] L. Huang and G. van Luijtelaar, "Brain computer interface for epilepsy treatment," *Brain-Computer Interface Systems-Recent Progress and Future Prospects*, 2013.

[10] G. Huve, K. Takahashi, and M. Hashimoto, "Brain-computer interface using deep neural network and its application to mobile robot control," in *2018 IEEE 15th International Workshop on Advanced Motion Control (AMC)*. IEEE, 2018, pp. 169–174.

[11] I. Karageorgos, K. Sriram, J. Veselý, M. Wu, M. Powell, D. Borton, R. Manohar, and A. Bhattacharjee, "Hardware-software codesign for brain-computer interfaces," in *International Symposium on Computer Architecture*, 2020, pp. 391–404.

[12] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, "Stdp-based spiking deep convolutional neural networks for object recognition," *Neural Networks*, vol. 99, pp. 56–67, 2018.

[13] C. Mühl, B. Allison, A. Nijholt, and G. Chanel, "A survey of affective brain computer interfaces: principles, state-of-the-art, and challenges," *Brain-Computer Interfaces*, vol. 1, no. 2, pp. 66–84, 2014.

[14] H. Nair, J. P. Shen, and J. E. Smith, "A microarchitecture implementation framework for online learning with temporal neural networks," in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, 2021.

[15] J. Smith, "Space-time algebra: A model for neocortical computation," in *International Symposium on Computer Architecture (ISCA)*, 2018.

[16] J. E. Smith, "Space-time computing with temporal neural networks," *Synthesis Lectures on Computer Architecture*, vol. 12, 2017.

[17] J. E. Smith, "A temporal neural network architecture for online learning," *arXiv preprint arXiv:2011.13844*, 2020.

[18] K. Sriram, R. P. Pothukuchi, M. Gerasimiuk, M. Ugur, O. Ye, R. Manohar, A. Khandelwal, and A. Bhattacharjee, "Scalo: An accelerator-rich distributed system for scalable brain-computer interfacing," in *Proceedings of the 50th Annual International Symposium on Computer Architecture*, 2023, pp. 1–20.

[19] D. Tan and A. Nijholt, *Brain-computer interfaces and human-computer interaction*. Springer, 2010.

[20] P. Vellaisamy, H. Nair, R. Vamsikrishna, D. Gupta, and J. P. Shen, "Tnngen: Automated design of neuromorphic sensory processing units for time-series clustering," in *2024 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2024.

[21] D. Wu, J. Li, Z. Pan, Y. Kim, and J. S. Miguel, "uBrain: a unary brain computer interface," in *International Symposium on Computer Architecture*, 2022.

[22] D. Wu, J. Li, R. Yin, H. Hsiao, Y. Kim, and J. San Miguel, "uGEMM: Unary Computing Architecture for GEMM Applications," in *International Symposium on Computer Architecture*, 2020.

[23] K. Yamazaki, V.-K. Vo-Ho, D. Bulsara, and N. Le, "Spiking neural networks and their applications: A review," *Brain Sciences*, vol. 12, no. 7, p. 863, 2022.

[24] B. Zhang, Z. Jiang, Q. Wang, J.-S. Seo, and M. Seok, "A neuromorphic neural spike clustering processor for deep-brain sensing and stimulation systems," in *2015 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*. IEEE, 2015, pp. 91–97.