TU Delft

Delft University of Technology

Benchmarking of hardware-efficient real-time neural decoding in brain-computer interfaces

Hueber, Paul; Tang, Guangzhi; Sifalakis, Manolis; Liaw, Hua Peng; Micheli, Aurora; Tomen, Nergis; Liu, Yao Hong

DOI
10.1088/2634-4386/ad4411

Publication date
2024

Document Version
Final published version

Published in
Neuromorphic Computing and Engineering

Citation (APA)
Hueber, P., Tang, G., Sifalakis, M., Liaw, H. P., Micheli, A., Tomen, N., & Liu, Y. H. (2024). Benchmarking of hardware-efficient real-time neural decoding in brain-computer interfaces. *Neuromorphic Computing and Engineering*, *4*(2), Article 024008. https://doi.org/10.1088/2634-4386/ad4411

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Neuromorphic Computing and Engineering

PAPER • OPEN ACCESS

Benchmarking of hardware-efficient real-time neural decoding in brain–computer interfaces

To cite this article: Paul Hueber *et al* 2024 *Neuromorph. Comput. Eng.* **4** 024008

View the article online for updates and enhancements.

You may also like

- Neural decoding of semantic concepts: a systematic literature review
Milan Rybá and Ian Daly
- A spiking neural network with continuous local learning for robust online brain machine interface
Elijah A Taeckens and Sahil Shah
- Generalized neural decoders for transfer learning across participants and recording modalities
Steven M Peterson, Zoe Steine-Hanson, Nathan Davis et al.

This content was downloaded from IP address 154.59.124.113 on 31/05/2024 at 12:13

NEUROMORPHIC
Computing and Engineering

PAPER

Benchmarking of hardware-efficient real-time neural decoding in brain–computer interfaces

Paul Hueber[1,2], **Guangzhi Tang**[1], **Manolis Sifalakis**[1], **Hua-Peng Liaw**[1], **Aurora Micheli**[2], **Nergis Tomen**[2] and **Yao-Hong Liu**[1,2,*]

[1] Imec-Netherlands, Eindhoven, The Netherlands
[2] Delft University of Technology, Delft, The Netherlands
[*] Author to whom any correspondence should be addressed.

E-mail: yao-hong.liu@imec.nl

Keywords: neural decoding, brain computer interfaces, closed-loop neuromodulation, spiking neural networks, neuromorphic computing

OPEN ACCESS

RECEIVED
15 November 2023

REVISED
22 February 2024

ACCEPTED FOR PUBLICATION
26 April 2024

PUBLISHED
17 May 2024

Original content from this work may be used under the terms of the Creative Commons Attribution 4.0 licence.

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

Abstract

Designing processors for implantable closed-loop neuromodulation systems presents a formidable challenge owing to the constrained operational environment, which requires low latency and high energy efficacy. Previous benchmarks have provided limited insights into power consumption and latency. However, this study introduces algorithmic metrics that capture the potential and limitations of neural decoders for closed-loop intra-cortical brain–computer interfaces in the context of energy and hardware constraints. This study benchmarks common decoding methods for predicting a primate's finger kinematics from the motor cortex and explores their suitability for low latency and high energy efficient neural decoding. The study found that ANN-based decoders provide superior decoding accuracy, requiring high latency and many operations to effectively decode neural signals. Spiking neural networks (SNNs) have emerged as a solution, bridging this gap by achieving competitive decoding performance within sub-10 ms while utilizing a fraction of computational resources. These distinctive advantages of neuromorphic SNNs make them highly suitable for the challenging closed-loop neural modulation environment. Their capacity to balance decoding accuracy and operational efficiency offers immense potential in reshaping the landscape of neural decoders, fostering greater understanding, and opening new frontiers in closed-loop intra-cortical human-machine interaction.

1. Introduction

Brain–computer interfaces (BCIs) have revolutionized the fields of neuroscience and medicine by enabling individuals with disabilities to interact with external devices and restore lost sensory [1], motor [2], or cognitive [3] functions. Intra-cortical BCIs (iBCIs), a type of invasive BCI that involves placing electrodes directly into the cortex of the brain, have great potential for closed-loop neuromodulation (CLN). CLN alters neural activity using personalized and responsive therapeutic electrical neural modulation based on the subject's neural activity. CLN has higher efficacy, [4] and lower risk of side effects [5] than fixed stimulation, that is open-loop neuromodulation as shown in figure 1. CLN requires neural decoders that interpret neural activity, such that BCI can provide real-time feedback stimulation or control external devices based on the subject's neural activity.

Designing iBCIs for CLN is challenging because of the highly resource-constrained environment of the implants. Even a slight temperature increase of one degree can cause damage to neural cells [6]. Moreover, a decoding time of a few milliseconds is required for CLN aimed at inter-areal interactions [7, 8]. This requires the development of energy-efficient and low-latency neural decoders that can overcome the constraints of low latency and energy consumption.

Benchmarking neural decoders for online *in vivo* iBCIs is crucial to ensure their optimal performance within the resource-constrained environment of implantable systems. By evaluating various decoders based

© 2024 The Author(s). Published by IOP Publishing Ltd

on their fidelity, latency, and power consumption, researchers can identify the most suitable options that satisfy clinical safety requirements and ensure effective real-time operation, ultimately improving the efficacy of CLN.

Traditional benchmarks [9–11] predominantly emphasize the accuracy and fidelity aspects of decoding methods. A recent addition, NeuroBench [12], expanded this focus to assess algorithm-hardware co-optimization, incorporating fidelity, efficiency, and performance metrics. While NeuroBench is well-suited for evaluating the operational cost of neural decoders, its algorithmic benchmark provides limited insights into power and latency, primarily relying on the effective operational cost as a proxy for hardware metrics. This paper, presents methods to extrapolate algorithmic-to-hardware metrics, addressing the gap encompassing all the essential constraints required to evaluate and compare the suitability of neural decoders for iBCIs in the context of CLN. Any benchmark designed to compare neural decoders for iBCI within the context of CLN must consider the co-optimization between hardware and software. Only then can we benchmark effectively and accurately evaluate power consumption, latency, and the fidelity of neural decoders, providing a holistic assessment of decoder suitability for real-time, low-energy applications.

This paper will introduce metrics that researchers can use to avoid the complex co-optimization process by evaluating neural decoders algorithmically while incorporating hardware considerations. Section 3 presents metrics designed to suit the energy and hardware-constrained environment of iBCIs suitable for CLN. Section 4 introduces six neural decoders benchmarks their ability to predict a primate's finger movements. Finally, future directions and implications of this research will be discussed.

2. Background

Intra-cortical neuronal recordings from the motor cortex have been pioneered by Delgado *et al* in 1952 [13] and Evarts conducted further groundbreaking work capturing extracellular neural activity from single recording units in conscious primates engaged in diverse motor tasks [14]. Today, almost 60 years later, neural recording has undergone a revolutionary evolution owing to innovative technologies such as high-density probes [15], high-density microelectrode array [16], and carbon nanotube yarn biosensors [17]. These technologies have made it possible to record the activities of more neurons with a higher spatial resolution and coverage and have paved the way for more clinically viable and high-performance iBCIs. BCIs help subjects with disabilities to interact with external devices, such as neuroprosthetics [18, 19], or restore lost sensory [1], motor [2], or cognitive [3] functions by translating neural activity from the brain into control commands through neural decoding. In addition to therapeutic applications, iBCIs advance our understanding of the complex neural processes that underlie behavior [20–23], cognition [24], and perception [25].

Two types of BCIs can be distinguished: non-invasive and invasive [26]. Invasive BCIs involve implanting electrodes into or on the cortex. iBCIs are a specific subtype of invasive BCIs, in which electrodes are inserted into the cortex, which is the outermost layer of the brain. They provide the finest spatial and temporal resolutions and excellent signal quality [27, 28]. Although iBCIs carry a higher risk owing to surgical implantation, their superior spatiotemporal resolution is crucial for high-precision neural decoding.

2.1. CLN

One promising field for iBCI is neuromodulation. Traditionally, neuromodulation described the physiological processes by which neurons use neurotransmitters to regulate neural activity [5]. More recently, neuromodulation has been adapted to refer to the process of altering neural activity via electrical stimulation to restore normal neurological functions or study intra-cortical interaction [5]. Neuromodulation can be classified into open and closed-loop systems, as shown in figure 1. Open-loop neuromodulation involves delivering neural stimulation without real-time feedback from the targeted neural system with predefined and fixed stimulation parameters, such as strength or timing. In contrast, CLN with iBCI uses bi-directional communication between the brain and the computing devices, providing adaptive feedback to adapt and adjust the parameters of interventions, enabling personalized and responsive therapeutic neural modulation based on the subject's neural activity [5, 29]. The adaptive and interactive nature of CLN enhances efficacy [4], i.e. maximizing the therapeutic impact and leading to more successful treatments, and reduces the side effects of neural stimulations [5] (e.g. discomfort, headache, or worst case seizure). For the remainder of this paper, CLN refers exclusively to neuromodulation via iBCI.

2.2. Constraints of closed-loop neuromodulation

CLN typically requires a powerful external computer to decode complex neural activities [30]. More sophisticated neural tasks (e.g. sensory and motor cortex interaction) require high channel counts of neural recording with fine spatial and temporal resolutions [18, 19], generating vast amounts of recording data,

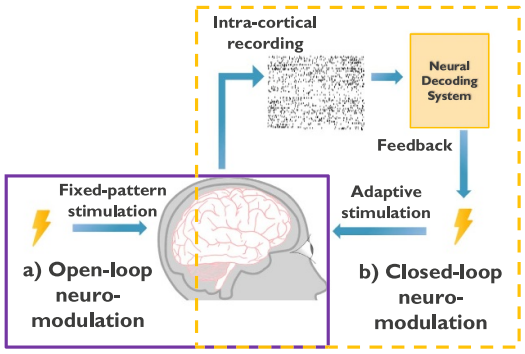

Figure 1. Paradigm of a closed-loop and open-loop neuromodulations. (a) During open-loop neuromodulation, the subject receives predefined stimulation. (b) During closed-loop neuromodulation (CLN), the subject receives adaptive stimulation based on the recorded and decoded neural activities. CLN enables individualized, responsive therapeutic treatment improving the effectiveness of the treatment and reducing side effects.

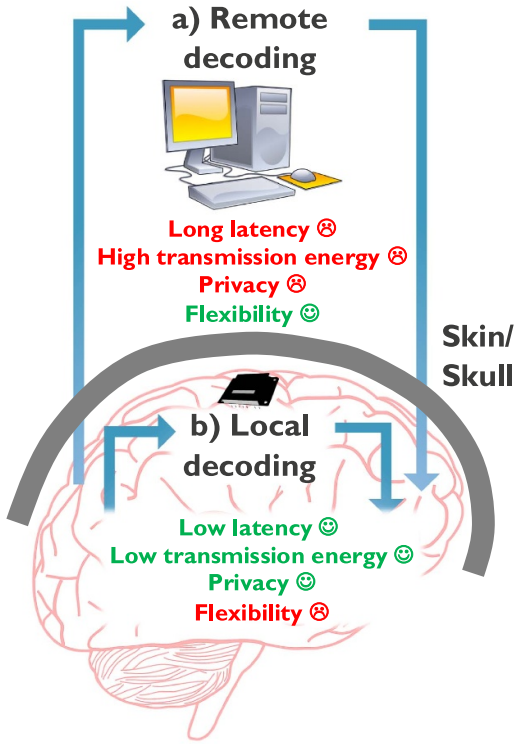

Figure 2. Advantages and disadvantages of external and local processing. (a) External processing requires transferring neural data or extracted features to an external computing device. Wireless data transfer suffers from long latency, high transmission energy, and privacy concerns. (b) Local neural processing has the potential to be significantly faster, with low transmission energy, at the cost of little flexibility.

which impose significant limitations on the real-time applicability of neural decoders, see figure 2. Transferring data from the intra-cortical neural sensors to an external system requires energy-intensive wireless transmission [31, 32], and limited wireless transmission bandwidth can increase the system's latency [6, 33–35]. Moreover, the transfer of neural data for processing to an external computer raises privacy concerns. Many neural decoders for iBCI and CLN have been implemented in application-specific integrated circuit (ASICs), achieving low power consumption and miniature form factor, which demonstrates the feasibility of *in vivo* neural decoding.

To address these concerns, data transmission can be avoided by eliminating the need for external computing and decoding neural signals locally on the implant. This eliminates the necessity for intensive data communication collectively except for programming the implant [30] or diagnostics [33]. Valencia and Alimohammad [30] highlighted the need for local processing in a fully implantable iBCI for *in vivo* closed-loop neural decoding, eliminating data transmission during inference and significantly reducing energy consumption, latency [33], and privacy concerns.

Designing iBCIs for CLN is challenging because of the highly resource-constrained environment of implants. The implant volume should be minimized to reduce the invasiveness and risks of the surgery, and low power consumption is required to prevent tissue damage due to even a one degree temperature increase [36]. However, iBCIs are required to process an exponentially growing amount of neural data [37], which adds to the complexity of the decoding task. Minimal heat diffusion is required to ensure safe local processing, without causing tissue damage. While Wolf [38] initially cautiously recommended limiting the temperature of implant electronics to below 40 available at: heat flux and 2 °C, a much lower bound of 1 °C is vital to maintain long-term neural cell health [6]. This means that a 10 mm^2 of implant electronics cannot exceed a power dissipation of 400 μW [38], which is $\sim$10$\times$ lower than that of smartphone processors (e.g. the microprocessor in iPhone concerns 100's mW of power with an ASIC area of approximately 100 mm^2). Given the conservative nature of these recommendations [38], minimizing power consumption beyond the stated limits is paramount. However, traditional processors cannot satisfy this power requirement [6], and low-power implants performing neural decoding algorithms is a major challenge to the successful clinical adoption of real-time closed-loop iBCIs [36].

Real-time processing is another crucial requirement for closed-loop iBCIs, in which continuous neural recording, decoding, and feedback occur in real time. This necessitates fast and efficient neural decoding algorithms to ensure timely and seamless interaction between the nervous system and the iBCI. Controlling a robotic prosthesis requires a latency of less than 150 ms between neural activity and arm movement for smooth and natural control [39], which is close to the biological delay for signal propagation between the brain and the arm [21]. However, immediate feedback is crucial for the neural decoder to adjust the control system in real-time, allowing subjects to adapt and refine their actions in real-time. In this closed-loop iBCI, much lower latency of less than 10 ms is necessary for decoding inter-areal interactions (e.g. sensory and motor) in the brain [7, 8].

3. Metrics

Traditionally, neural decoders have been the primary evaluated on decoding performance. However, in hardware-constrained iBCIs, other factors, such as latency and power consumption, are crucial to ensure the tractability of neural decoders for applications such as CLN. Evaluating decoders solely on task performance often fails to capture their true potential but also limitations. This chapter introduces the need for comprehensive metrics for algorithmic evaluation that encompass fidelity (i.e. accuracy), latency, power consumption, and memory size for neural decoders. Table 1 presents an overview of the proposed evaluation metrics.

3.1. Model fidelity

The fidelity of a neural decoder in an iBCI is its ability to correctly classify or predict. In closed-loop iBCI, making accurate predictions is crucial for effective and reliable control of external devices or neural stimulations, which should closely reflect the subject's intention or state.

Classification accuracy has traditionally been used to assess neural decoding performance by determining the percentage of correct classifications of stimuli, such as odors [1], faces [40], or speech [2, 41]. However, these metrics do not consider the temporal continuity of neural data, which is critical for many neural decoding applications. Neural decoding tasks require metrics that incorporate the correction of the temporal regression to evaluate decoding accuracy. In such cases, the coefficient of determination (R^2) [3, 11, 20, 30, 42–44] and the coefficient of correlation (Pearson's r) [42, 44–47] are widely used to assess the performance of neural decoding algorithms.

The R^2 measures the proportion of variance in the dependent variable explained by the model's prediction, while Pearson's r evaluates the temporal alignment of the predictions and labels via a linear relationship. Both metrics should be reported to assess the temporal regression performance of the neural decoder comprehensively.

3.2. Latency

The latency of the neural decoder, defined as the time delay between the first input stimulus and the output response [48], comprises two architecture-specific subcomponents: the *binning latency* and the *processing latency* (see figure 3(c)). This allows for a platform- and architecture-agnostic evaluation of neural decoders.

The *binning latency* corresponds to the time span of the input data required for each prediction. This equates to the binning time window or the history of binning windows in the case of multiple windows. Minimizing binning window size is crucial for limiting total latency.

The *processing latency* is caused by the processing time for a neural decoder to produce a prediction. This combines the operational delay of preprocessing, network inference, and postprocessing. The processing

Table 1. Overview over the proposed closed-loop iBCI metrics.

	Metric	Explanation
Fidelity	R^2	Proportion of explainable data variance
	r	Temporal alignment of label and prediction
Latency	Binning latency	Timespan of input data needed per inference
	Processing latency	Operational delay optimized by the algorithm designer
Power consumption	Total eff. operations	Number of effective operations needed per inference
	Memory access	Number of effective memory access needed per inference
Size	Memory footprint	Number of bits required to store the decoder

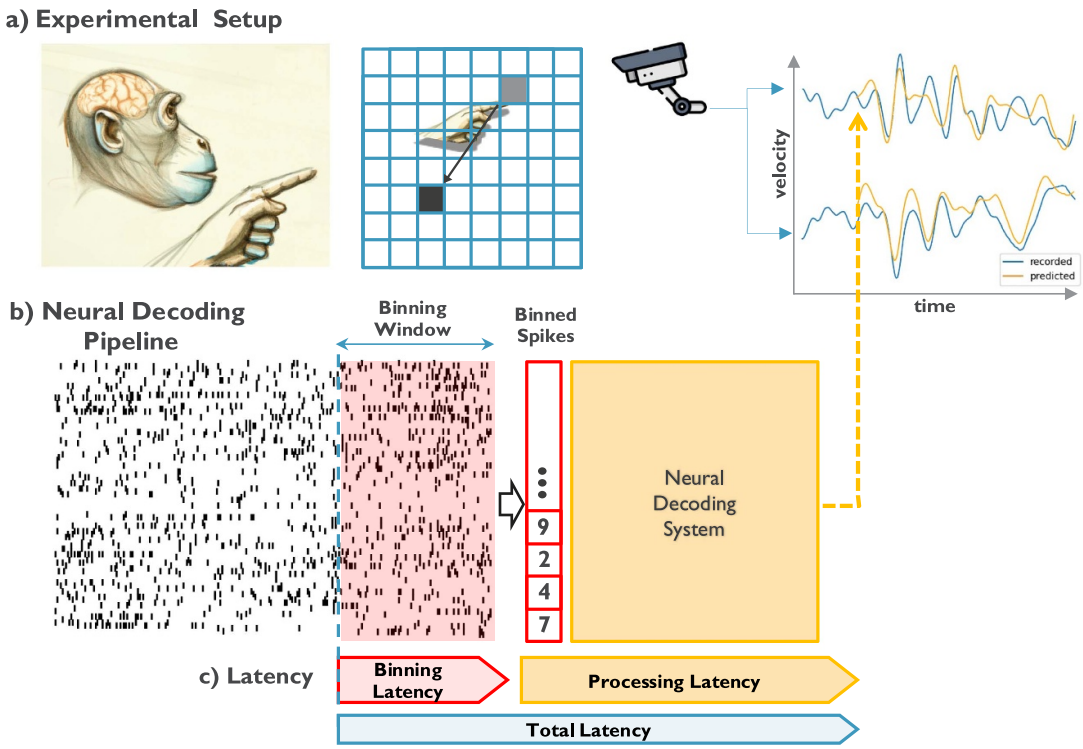

Figure 3. Experimental pipeline for evaluating closed-loop capability of neural decoders. (a) A primate consecutively reaches for black boxes on a grid. Neural activity is recorded using one or two UTAH arrays placed at the primary motor and sensorimotor cortices, while finger position is tracked through an electromagnetic position sensor. Finger velocity is computed and used as a decoding target. (b) Neural activity is processed as a spike train, binned, and processed by the neural decoding system to predict finger velocity. (c) The visualization illustrates the latency of the entire decoding system, composed of binning latency and processing latency.

time is bound by a function of the system's required effective operations per inference, which can be assessed by computing the effective multiply-and-accumulate (MAC) operations of neural decoder algorithms. This definition ignores a potential speed-up of parallel processing, which would require binding the algorithm to hardware. Latency can be reported in wall time, such as absolute SI units [48], or relative system time, as the total number of clock cycles per inference. This paper reports latency in milliseconds, providing a more intuitive and user-centric perspective and allowing the reader to assess the system's ability to deliver timely and accurate responses. Converting the *processing latency* into seconds requires platform-specific assumptions regarding the required clock cycles per operation, the clock frequency, and a system's capability for parallel processing. For the remainder of this paper, a clock frequency of 1 MHz and 3 MAC operations per clock cycle were assumed [49], to provide a more intuitive comparison of the latency of the evaluated decoders. For simplicity, one addition corresponding to the sparse synaptic operation of neurons in the SNN is assumed to be equivalent to one MAC in terms of required clock cycles.

3.3. Power consumption

Power consumption is vital to evaluating neural decoders, particularly in resource-constrained iBCIs suitable for CLN. Local neural processing, which is implemented directly on an embedded device, is usually preferable to external processing due to latency, communication bandwidth, and privacy issues. However, local neural processing requires low energy consumption to minimize tissue heating.

To compare the energy efficiency of different neural decoders in a hardware- and architecture-independent manner, one can benchmark with two hardware-agnostic metrics: total effective operations and memory accesses. This considers only algorithmic optimizations, such as reducing effective operational costs. Although algorithm-hardware co-optimizations can further improve latency, performance, and energy efficacy, they require binding the neural decoder to specific hardware and, thus, are not considered in this study.

The total effective operations are reported as the number of non-zero operations required per inference. This combines two relevant operations that dominate neural network operations: multiplication and addition. To estimate the energy consumption of a neural decoder, however, total operations should be reported instead of MAC and ACC since these operations are assumed to be optimized by vector accelerators, which is the bottleneck for latency but does not reflect well on the energy cost of a neural decoder. To reduce energy consumption, neural decoders can exploit the sparsity of the spiking data by distinguishing between effective and ineffective operations. This accounts for only non-zeros contributing to the products and, consequently, the accumulation, which can be leveraged by specialized hardware. Reporting the effective computational cost as the number of non-zero operations allows for hardware-agnostic comparison of different networks, considering computational primitives in neuromorphic neural networks. In the remainder of this paper, MAC denotes effective MAC operations.

Reporting memory access is crucial for comprehensively estimating the energy consumption of neural decoders. Because a read-and-write operation to the memory requires one to two orders of magnitude more energy than operations of arithmetic linear units [48, 50], it is insufficient to report only effective operations without including the memory access. Furthermore, Liao *et al* [46] report ∼10× more reads than effective operations during inference, highlighting that most of the energy consumption of an architecture comes from the memory read-and-write operations. Following their approach, the number of memory accesses in a network is conservatively estimated by assuming that a MAC operation consists of three loads and one store. By contrast, an ACC consists of two loads and one store, which both, similarly to before, need to be combined with the sparseness of activity of the network [46].

3.4. Memory footprint

Due to space volume constraints of iBCIs, the memory size, which in comparison to other function blocks consumes far more ASIC area, should be reported. If the memory requirements of the neural decoder are too large, external memory is required, which can consume more than 100× more energy than on-chip memory [51]. Estimating the memory footprint of a neural decoder involves calculating the sum of the network's parameters and variables, as well as the memory requirements of input data from binning windows. The memory size should be reported in bits to provide credit to architectures that limit the precision of weights.

4. Methods

The methodology is structured into five subsections, addressing how this study evaluates and benchmarks various neural decoders within the context of iBCI for CLN. The first subsection provides an overview of the neural data utilized in this benchmark, which comprises neural activity recorded from non-human primates, representing a relevant scenario for closed-loop iBCI applications. Decoders are categorized into three main groups: traditional neural decoders, artificial neural networks (ANN)-based neural decoders, and neuromorphic spiking neural network (SNN)-based decoders. The selection of traditional and ANN decoders was based on existing literature, while the long short-term memory (LSTM) and SNN-based decoders were optimized for this study. The presented decoders are non-exhaustive (the interested reader is referred to recently proposed models [52–56]), yet the decoders aim to represent commonly used neural decoding methods. The final subsection outlines the experimental setup and details the conditions and parameters under which the benchmark was conducted. These methodological components provide a detailed pipeline for evaluating neural decoding methods for closed-loop iBCI applications.

4.1. Neural recording dataset

The ANN and SNN neural decoders were trained and benchmarked on the 'primate reaching task' of the neuromorphic benchmark NeuroBench [12], as visualized in figures 3(a) and (b). This benchmark employs the same data as Makin *et al* [20], which previously were used to evaluate their traditional decoders. The R^2

value published by Makin *et al* [20] is reported here, and the metrics for power consumption and latency are calculated conservatively based on the most time-consuming and energy-intensive matrix operations, as defined in Chapters 4.2 and Chapter 4.3.

The dataset is a subset of 6 out of 33 recording sessions of 2 Rhesus primates during subsequent reaching tasks [20] as shown in figure 3(a), which was released by Dyer *et al* [57]. These sessions encompassed two non-human primates (NHPs) and the entire recording period. Three NHP 'I' sessions were recorded using a 96-channel Utah array from Blackrock Neurotech implanted in the primary motor cortex. The three NHP 'L' sessions had an additional Utah array in the sensorimotor cortex, enabling the simultaneous recording of 192 channels. Each recording session comprises 354–819 individual reaches, and those longer than 8 s were discarded as they indicate the primate's inattention.

Spikes were detected from the raw neural data using a threshold of 3.5–4 times the root-mean-square (RMS) noise. The finger position was recorded using an electromagnetic position sensor at 250 Hz, and the velocity was computed as the discrete gradient of the position. Predicting the translation invariant finger velocity better aligns with the natural dynamics of limb movements and corresponds to how neural activity encodes kinematics [58]. Makin *et al* [20], and Dyer *et al* [57] report the detailed experimental setup and the data acquisition.

4.2. Traditional neural decoders

Classic neural decoding methods have been extensively used in brain-computer interfacing. Of the six decoders Makin *et al* [20] presented, the three best-performing models were selected to represent traditional neural decoders. Those decoders are the 'unscented' Kalman Filter (UKF), the static, and the dynamic 'recurrent exponential-family harmonium' (rEFH), and they are evaluated due to their established performance and versatility in handling various neural data modalities. Makin *et al* also considered linear regression, conventional KF, and Wiener filter. However, since these decoders had worse R^2 performance and scaled poorly with decreasing binning windows, they were not considered suitable for this neural decoding benchmark for closed-loop iBCIs suitable for CLN.

The UKF approximates the state distribution by applying a full nonlinearity to a minimal set of carefully selected representative points. It was designed as an extension of the KF to address the problem of its exploding residuals on the true posterior mean and covariance. The UKF solves this by replacing the normal sampled state distribution with a deterministic sampling of this distribution [59]. The UKF, as described by Makin *et al* [20] and Wan *et al* [59], has a state space of deterministically sampled 40 variables, which requires $O\left(n^3/6\right)$ operations due to Cholesky decomposition. As a conservative estimate of the *processing latency* and the operational cost, only matrix multiplications required during inference and the computationally intensive matrix inversion were considered. This large-scale matrix inversion was assumed to require at least 200 ms. [60].

The rEFH, instead of assuming Gaussian state variables, models the state variables as a variant of a restricted Boltzmann Machine (RBM) [61] and explicitly samples the spike count from a Poisson distribution. The static variant converts the latent space of the RBM into kinematics via static mapping, that is, a matrix multiplication, whereas the dynamic version uses a KF. For the static and the dynamic rEFH models, only the forward pass of the RBM uses higher-dimensional matrix multiplications, and thus is considered. There are four times as many hidden neurons in the RBM than there are input channels and 1800 output neurons mapped to the kinematic output via either a matrix multiplication or a KF [20]. All traditional decoders were evaluated using binning windows of 16 ms, 32 ms, 64 ms, and 128 ms.

4.3. ANNs

This study used two ANN-based decoders as baselines owing to their established history of high-accuracy predictions. Previous studies have demonstrated that ANN-based decoders perform on par or better than traditional decoders [11, 46]. One fully connected ANN, published as a baseline for the NeuroBench benchmark [12], and one LSTM network were evaluated.

The ANN is a conventional 3-layer feedforward network with 32 and 48 hidden and two output neurons, respectively. This implementation uses a history of multiple non-overlapping binning windows that are flattened and processed as input data. The default implementation, including the history of 7 binning windows, is shown in figure 4(a). To explore the latency versus fidelity trade-off, a history of 4, 7, and 14 binning windows of 28 ms was considered, and the number of neurons in the hidden layers halved and doubled.

The LSTM, as shown in figure 4(b), uses a fully connected layer to reduce the input dimensionality to 16, followed by a single LSTM cell with 16 hidden neurons. A feedforward layer returns the predicted kinematics. Similar to the ANN, binning windows of 4 ms, 8 ms, and 16 ms and wider networks, with 64 and

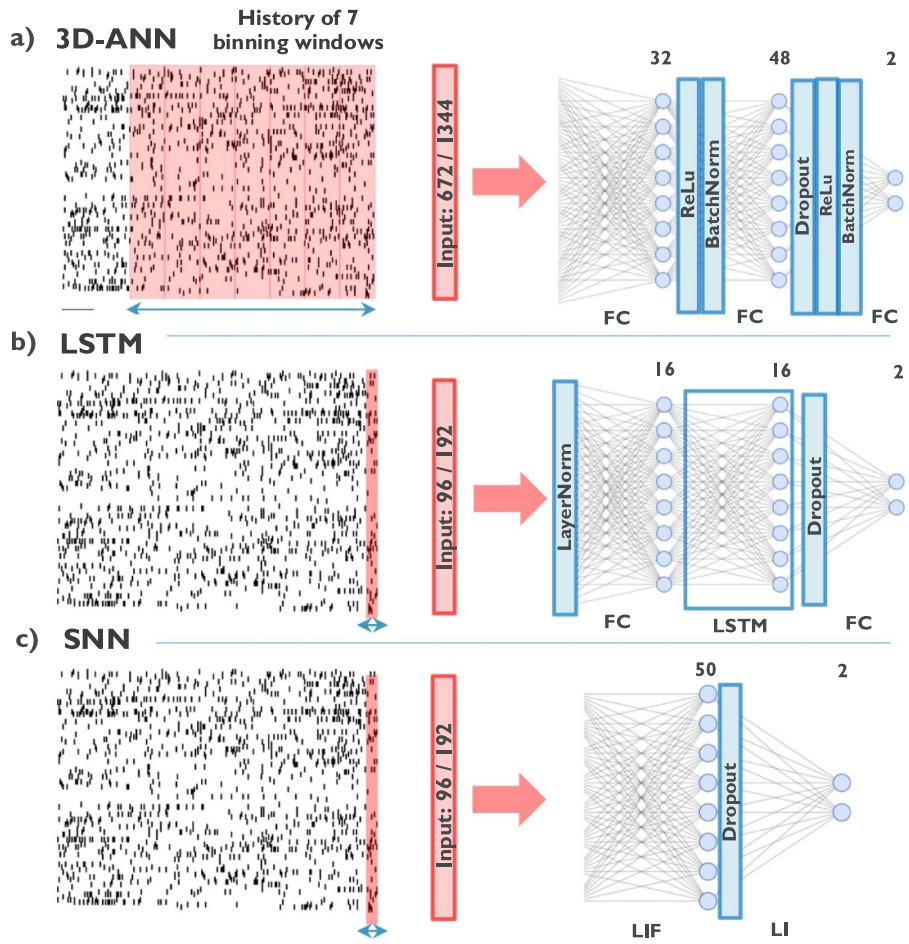

Figure 4. Architecture of three neural network-based decoders. The data extraction and binning are visualized in red, and the network architecture in blue. Each layer's dimensions and type are stated above and below, respectively. (a) The ANN uses seven binning windows spanning 28 ms as input. These extracted windows are flattened and processed by two hidden layers with 32 and 48 Neurons, respectively. (b) The LSTM extracts spikes with a temporal resolution of 4 milliseconds, effectively representing the neural data as a spike train. Then, the data undergoes dimensionality reduction via a fully connected (FC) layer with 16 neurons, after which a long short-term memory (LSTM) cell is employed. (c) Similarly, the SNN decoder extracts the spike train and processes it through a network featuring 50 hidden Leaky Integrate-and-Fire (LIF) neurons. The final output of this network comprises the membrane potential of two LIF neurons.

128 hidden neurons, were examined. The networks use batch normalization (BatchNorm) and layer normalization (LayerNorm), respectively, and use Dropout.

4.4. SNN

SNNs are a variant of neural networks that attempt to mimic the properties, processes, and functions of biological neurons. This makes them inherently recurrent and allows them to exploit sparsity to achieve lower latency [22, 62–69] and power consumption [3, 46, 62–66]. SNNs, at their core consist of stateful spiking neurons, a more bio-plausible variant of the Perceptron, with the leaky integrate-and-fire (LIF) being the most widely used neuron model. As the Perceptron, the LIF weighs and accumulates the input, but instead of returning this weighted accumulation, it is added to the neuron's membrane potential. If the membrane potential exceeds a threshold, the neuron produces a binary output, that is, a spike, and the membrane potential is reset; otherwise, it decays per time step, as conceptually illustrated in figure 5. This enables the neuron to combine information over multiple time steps, making it inherently recurrent. In addition to their binary output, LIF neurons exploit binary input data, enabling sparsity in networks built around these neurons. The binary input separates the operations into effective and ineffective because multiplications by zero do not contribute to the accumulation. Therefore, the multiplication of weights times input can be forgone by adding only non-zero weights.

The implemented neuromorphic SNN decoder is a simplified version of the model proposed by Liao *et al* [46] with fewer learnable parameters. In a preliminary exploration, reducing the complexity of the network only marginally impacted the accuracy while significantly improving the operational cost. Each hidden layer is decreased to 50 LIF-neurons without a bias term and a fixed decay ($\tau = 0.96$). The BatchNorm layer is

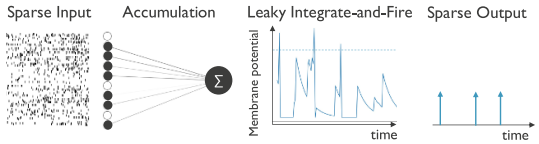

Figure 5. A scheme of the LIF neuron. In this neuron model, incoming neural signals are integrated into the membrane potential, and if the accumulated value surpasses a specified threshold, the neuron generates a binary output signal. The 'leaky' property allows for the gradual decay of the accumulated charge, further contributing to the model's simplicity. The intrinsic recurrent nature of the LIF neuron originates in the updated membrane potential.

removed because combining BatchNorm and Dropout leads to models with different feature variances inside the network during training and testing [70]. The threshold of the LIF neurons is set to one. To further push for lower latency and power consumption and explore the various tradeoffs against accuracy, the number of hidden layers is decreased from three to two and one (SNN3, SNN2, and SNN1, respectively). SNN1 is a baseline model for the 'primate reaching task' of NeuroBench [12] and is depicted in figure 4(c).

4.5. Experimental setup

The spike train was binned according to the window requirements of the respective network, and a sliding window with a stride of 4 ms extracted the spiking data. The RNNs and the SNNs used a sliding window of 50 bins to extract temporal information from the data and the non-recurrent ANNs extracted the bins individually. The neural decoders were trained to predict the finger velocity as visualized in figure 3.

RNNs and SNNs notoriously suffer from difficulties when learning long-term dependencies from data [71, 72]. To improve the gradient flow, the SNN was trained to predict the time window of the primate's finger kinematics and, for consistency, was tested to predict individual finger velocities as the ANNs. The Loss for this setup was a linearly weighted mean squared error (MSE) from zero to one to account for warm-up steps. The ANN and LSTM were trained using the conventional MSE Loss.

Each session was divided into the first 75% of the reaches shuffled and, in contrast to the Neurobench-proposed evaluation pipeline, was used for model selection using 10-fold group cross-validation with early stopping and the last 25% for testing. This allows for finding the optimal hyperparameters and the number of training epochs while reporting more robust performance.

5. Results

We considered six decoders trained on reconstructing a primate's finger velocity given binned neural activity with metrics that allow assessing a decoder's latency and power consumption. The results in table 2 offer a complete overview of the performance of all decoders. The performance of the decoders is in line with previous literature given the reduced dataset, different training paradigms and different metrics [12, 20].

5.1. Latency vs. fidelity

A higher R^2 performance can typically be achieved using deeper and more complex architectures or by better estimating the neural firing rate with a longer binning window. Both approaches negatively impact the latency and, thus, the capability of the neural decoder to facilitate real-time closed-loop feedback. The performance tradeoff of the six decoders is visualized in figure 6.

The UKF requires computationally intense matrix operations and a matrix inverse, making it significantly slower than the other decoders and achieving lower fidelity. Contrary to the rEFH filters, the R^2 score improved with decreasing binning windows. The best-performing rEFH filter had an R^2 of 63.19% (std = 0.17) with a latency of 129 ms, significantly outperforming the UKF with an R^2 of 45.10% (std = 0.12) and a latency of 270 ms both in terms of fidelity and latency.

The rEFH filters can achieve comparable R^2 scores to NN-based decoders. However, they require long binning windows to approximate the state distribution and experience a stark drop in the R^2 scores with smaller binning windows. Nevertheless, the trendlines of the rEFH and ANN models indicate that the rEFH can achieve a better latency versus fidelity tradeoff than the ANN. This stems from the ANN requiring a considerable history of binning windows to extract temporal information, which result in high latency. For the ANN, reducing the number of binning windows led to a significant decrease in fidelity. The shallow LSTM attains a much lower latency versus fidelity tradeoff than the traditional decoders and ANNs, indicated by the trendlines, achieving peak R^2 scores above 60% while having a latency between 4.05 ms and 16.05 ms. Notably, the fidelity drops significantly when using a longer binning window of 16 ms, indicating that the LSTM relies on the temporal dimension of the neural data to extract information about neural

Table 2. Experimental results of 6 neural decoders for decoding finger velocities. Six recording sessions of primates executing reaching tasks were selected and split into the first 75% of the reaches for 10-fold cross-validation and the final 25% for testing. Notably, the latency of the UKF is consistently high, given the delay caused by the computationally intensive matrix inverse.

		Fidelity				Latency			Power consumption		Size
Decoder type		R^2	±sd	r	±sd	Bin (ms)	MAC	Total (ms)	Eff. Ops	Memory Access	Memory Footprint
Traditional neural decoder	UKF	0.4510[a]	0.15	—	—	16	28 799	269.6	23 M	116 k	736 Mb
		0.4485[a]	0.16	—	—	32					
		0.4290[a]	0.17	—	—	64					
		0.4122[a]	0.32	—	—	128					
	rEFH static	0.2103[a]	0.17	—	—	16	3131	17	7 M	12 k	224 Mb
		0.3879[a]	0.20	—	—	32		33			
		0.5042[a]	0.17	—	—	64		65			
		0.5961[a]	0.14	—	—	128		129			
	rEFH dynamic	0.4248[a]	0.18	—	—	16	3167	17.1	7 M	13 k	224 Mb
		0.4996[a]	0.17	—	—	32		33.1			
		0.6038[a]	0.14	—	—	64		65.1			
		0.6319[a]	0.14	—	—	128		129.1			
ANN neural decoder	ANN (hist = 7)	0.6369	0.06	0.8023	0.03	196	162	196.1	34 k	136 k	1 Mb
	ANN (hist = 4)	0.5768	0.05	0.7615	0.03	112		112.1	20 k	80 k	647 kb
	ANN (hist = 14)	**0.6515**	0.09	**0.8159**	0.04	392		392.1	66 k	265k	2 Mb
	ANN (2x neuron)	0.6403	0.06	0.8051	0.03	196	322	196.1	71 k	285k	2 Mb
	ANN (0.5x neuron)	0.63	0.05	0.7989	0.03	196	82	196.1	17 k	67 k	532 kb
	LSTM (bin = 4 ms)	0.6014	0.04	0.7781	0.03	4	146	**4.05**	14 k	69 k	611 kb
	LSTM (bin = 8 ms)	0.6391	0.05	0.8027	0.04	8	146	8.05			
	LSTM (bin = 16 ms)	0.5026	0.15	0.7128	0.10	16	146	16.05			
	LSTM64	0.6036	0.05	0.7784	0.03	4	578	4.19	53 k	224 k	2 Mb
	LSTM128	0.6109	0.05	0.7834	0.04	4	1154	4.38	162 k	659k	5 Mb
SNN neural decoder	SNN1	0.5948	0.06	0.7723	0.04	4	167	**4.05**	**167**	**1503**	**158 kb**
	SNN2	0.6292	0.06	0.7968	0.04	4	202	4.07	202	1815	240 kb
	SNN3	0.6071	0.08	0.7861	0.05	4	305	4.08	229	2068	322 kb

[a] The results of the UKF and the rEFH are reported from Marmerstein *et al* [17], which only comprise R^2 scores.

dynamics. The SNN achieved the best tradeoff with competitive R^2 scores and lower latency for all three models examined. The latency scales only marginally with an increasing number of layers.

The overall trend of the six decoders shows that recurrent neural networks (RNNs), such as the SNN and the LSTM, can achieve competitive fidelity to non-recurrent ones while extracting neural dynamics from intra-cortical spike data and having significantly lower latency, with SNNs maintaining lower latency for higher fidelity than LSTMs.

5.2. Power consumption vs. fidelity

MACs, reported as the number of inner products of matrix-vector multiplications, are indifferent to the length of the vectors in the inner product. This makes them inadequate for assessing power consumption. Instead, the total number of required operations during inference provides a better estimate of the actual energy cost. Figure 7 shows the operational cost versus fidelity tradeoff of the neural decoders.

All traditional decoders require matrix operations with large inner products, leading to, by far, the most operations to effectively decode neural signals. The average number of operations for traditional decoders across the experiments was between 700 000 and 2300 000. The three decoders have the same high number of total operations across the various binning windows because unlike the other models, the length of the binning window does not affect the operational cost. Traditional decoders had the most extensive range of R^2 scores, ranging from 23% to 66%. NN-based decoders represent a shift towards more computationally

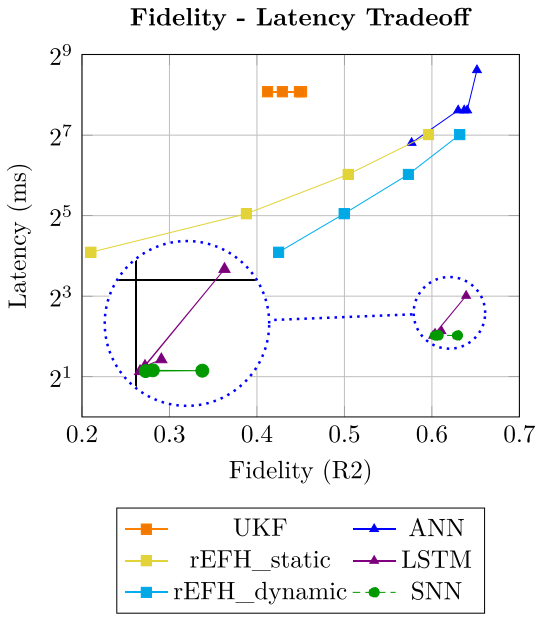

Figure 6. The accuracy versus latency trade-off of the six evaluated decoders. The R^2 fidelity is plotted on the horizontal axis, and the vertical axis is the millisecond latency. In the plot, traditional decoders are represented as squares, artificial neural network-based decoders as triangles, and spiking neural networks (SNNs) as circles. The decoders with the best trade-off are in the bottom right corner. Recurrent decoders such as LSTM and SNN achieve substantially lower latency while maintaining competitive accuracy.

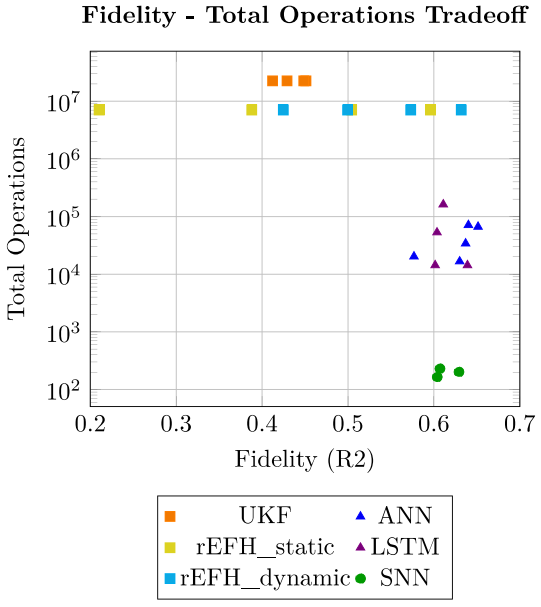

Figure 7. Visualized trade-off between R^2 score and total operations. The specific closed-loop setting requires neural decoders in the bottom right corner. ANN-based decoders, such as the ANN and the LSTM depicted as triangles, are an improvement compared to traditional decoders, represented as squares. Yet, the SNN, denoted by circles, achieves the lowest operational cost while maintaining competitive R^2 scores. Comparing memory access instead of total operations shows the same trend and is redundant.

efficient decoding algorithms. Our experiments demonstrated significantly reduced power consumption compared to traditional decoders. The LSTM with 16 hidden neurons required 4700 operations, whereas the ANN with seven binned windows required approximately 34 000 operations. Compared to the traditional decoders, this significant improvement comes with consistently high fidelity, with R^2 values of ANN-based decoders ranging from 50% to 65%. The SNN decoders exhibited the highest energy efficacy among the decoder types considered in this study. On average, SNN decoders required 200 operations while achieving competitive fidelity levels, with R^2 values ranging from 60% to 63%.

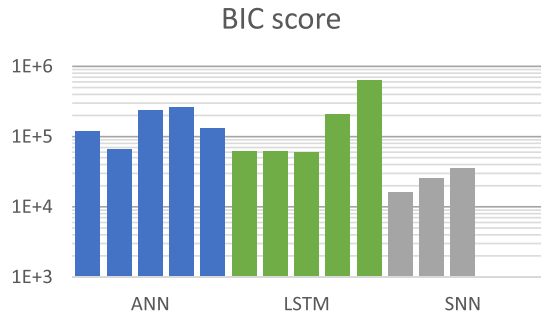

Figure 8. The evaluated NN-based decoders' Bayesian information criterion (BIC) shows that the SNN achieve far lower BIC scores than other decoders, suggesting that SNNs are more suitable for the constrained closed-loop setting. A low BIC indicates that a decoder can learn better, given its complexity. The lowest BIC score is achieved by SNN1. The various models appear in the same order as presented in table 2.

Comparing memory access instead of total effective operations reveals the same tradeoff and is not reiterated. A comparison of the power consumption and fidelity metrics reveals an intriguing trade-off among the three decoder types. While traditional decoders require substantial operations, they offer a range of R^2 scores. In contrast, ANN-based and SNN-based decoders provide the advantage of reduced energy costs, with SNN decoders exhibiting the lowest computational load while maintaining competitive fidelity levels.

5.3. Bayesian information criterion (BIC)

The BIC serves as a valuable instrument for the comparing various neural networks, effectively addressing the concern of increased model complexity and its potential impact on performance enhancement. The BIC introduces a penalization term for the number of model parameters, thereby discouraging the adoption of overly complex models with excessive weights and biases. This penalty term effectively balances fidelity and model complexity, enabling us to discern whether a model's performance improvements stem from an increased number of learnable parameters or architectural design.

Figure 8 presents the BIC values for various configurations of the three neural network-based decoders. Notably, the single-layer SNN, characterized by its minimal complexity, attains the lowest BIC score. In contrast, SNN2 and SNN3 had significantly higher BIC scores despite exhibiting performance improvements. This discrepancy suggests that the performance improvements are disproportional to the increased number of learnable parameters in these models. All SNNs achieved much lower BIC scores than the traditional and ANN-based neural decoder.

6. Discussions

In conclusion, optimizing neural decoders for closed-loop iBCI systems capable of CLN presents a delicate balance, requiring careful consideration of the trade-offs between fidelity, latency, power consumption, and memory size. Our findings emphasize that although more complex and deeper neural architectures with more trainable parameters, hold the potential for improved decoding accuracy, optimizing only for fidelity by increasing the complexity of a network can result in reduced usability for closed-loop iBCIs. The decoding accuracy reaffirms the findings of Glaser *et al* [11] that conventional neural network-based decoders can achieve the highest R^2 scores. However, we observe that this comes at the cost of increased latency and power consumption. Even when only considering fidelity, evaluating the BIC across the three NN-based decoders showed that SNNs consistently outperformed the ANN and the LSTM, achieving significantly lower BIC scores. This indicates that the performance improvement of the ANN is due to disproportionally more learnable parameters. Remarkably, the single-layer SNN emerged as the top performer out of the models we benchmarked in this paper, signifying its suitability for effectively learning data variance, particularly when considering the number of learnable parameters. This highlights that the shallow SNN is preferable for robust and energy-efficient neural decoding, given its complexity among the three neural network-based decoder types we evaluated.

The ability of a neural decoder to effectively harness sparsity represents a crucial design consideration in closed-loop iBCI systems. Conventional neural data are characterized by their inherent sparsity and temporal encoding [21, 35, 73, 74], with rate-based encoding accounting for a mere fraction of the neural activity in regions such as the visual cortex [75]. The inherent sparsity of neural spikes provides SNNs with a distinct

advantage, enabling them to capitalize on the spatiotemporal structure of the input data, which is less pronounced in non-neuromorphic ANNs.

SNNs have previously demonstrated their potential for reduced power consumption and lower latency in various applications. In our study, we reaffirm and extend these prior findings, indicating that SNNs can extract neural dynamics from extracellular spiking data, while maintaining competitive fidelity and showing superior performance in terms of power consumption and latency.

The latency metric introduced in our analysis operates under the assumption that the computation of an inner product is equivalent to a single addition in terms of clock cycles. Although this abstraction aligns with standard practices for MAC operations, it is essential to acknowledge that this assumption may only partially represent the potential hardware optimizations for vector accumulations. While the process of counting additions might initially appear as a potential disadvantage when comparing SNNs to ANNs in the context of latency, we observe that despite this methodological abstraction, SNNs consistently achieved substantially lower latency when compared to traditional decoders and the ANN. The longer latency of traditional decoders and ANNs is primarily attributed to their reliance on long binning windows for extracting temporal information and their high operational cost. Notably, the sole exception to this trend is the LSTM, which demonstrates a latency level comparable to that of SNNs. This observation reaffirms the findings of Zenke *et al* [73], who demonstrated the efficiency of RNNs, such as LSTMs and SNNs, in exploiting the temporal structure of neural data, emphasizing their capacity to achieve competitive fidelity with low latency in a closed-loop neural decoding system. These insights underscore that even without accounting for the potential hardware optimizations, SNNs can exhibit a marked advantage in terms of latency over traditional decoder models and non-recurrent ANNs, which require more extensive computational resources owing to their temporal information extraction procedures.

The advantages of employing neuromorphic SNNs for closed-loop iBCIs become more apparent when the energy cost is evaluated using the total number of operations. In this context, SNNs significantly outperform traditional and ANN-based decoders. Our results show that traditional and ANN-based decoders require several orders of magnitude more operations to attain performance levels comparable to SNNs. The remarkable reduction in the number of operations required by SNNs can be attributed to their constrained design, which minimizes the number of learnable parameters while still delivering competitive performance. However, the primary driving forces behind the substantially lower energy cost associated with SNNs lies in their ability to exploit sparsity [46] and their more energy-efficient operations. This capacity allows for approximately 5% of the operations to be executed, emphasizing the exceptional efficiency achieved by SNNs while maintaining high fidelity. Following previously reported estimates of required energy per operation, we observe an approximate energy cost of around 2 μW per inference for the SNNs [50, 76], which is 50 times lower than for the LSTM, 100 times lower for the ANN, and 10 000 times lower than for the UKF.

The field of neurotechnology and BCIs are rapidly expanding. New advancements and technologies enable the development of more effective, user-friendly, and versatile BCIs. This paper discusses two main challenges in designing processors for implantable closed-loop neural decoders: low energy consumption to minimize heat diffusion, and low latency to enable real-time CLN. We defined metrics for neural decoders and benchmarked common decoding methods to predict a primate's finger kinematics. This study explores the suitability of low latency and low computing neural decoders and highlights the potential advantages of neuromorphic SNNs for CLN. Our results show that SNNs can balance decoding accuracy and operational efficiency, offering immense potential for reshaping the landscape of neural decoders and opening new frontiers in closed-loop intracortical human-brain interaction.

Using neuromorphic SNNs for CLN is an area of research with great promise, as indicated by successfully predicting an NHP finger kinematic in this study. However, the list of evaluated decoders is non-exhaustive, and only a single neural decoder, the SNN, was optimized for latency and power consumption. This allows for comparing commonly used decoders, yet it favors inherently efficient and fast neuromorphic decoders. Additionally, only one exemplary dataset was evaluated as representative of closed-loop iBCI tasks requiring low latency and power consumption. Therefore, this study highlights the potential of SNN for iBCI for CLN, however, further studies are required to explore the suitability of these networks for other types of neural decoding tasks and to optimize their performance to meet the requirements of CLN systems. Developing fully implantable iBCIs with local processing capabilities is crucial for reducing energy consumption and latency and improving the real-time applicability of CLN systems.

Overall, the outlook for neural engineering and BCIs is bright. New developments will improve neural recording and decoding technologies, ultimately enhancing our understanding of the brain and its complex neural processes.

7. Conclusion

Our study introduces methods to extrapolate algorithmic-to-hardware metrics that allow evaluating the low latency and high energy efficacy requirements of iBCI suitable for CLN. We present six commonly used neural decoders and compare them in predicting an NHP fine motor kinematics from binned neural activity. Our results highlight the potential advantages of neuromorphic SNNs in the context of iBCIs capable of CLN. In our benchmark, we observe that SNNs outperform other commonly used decoders, with evident performance differences when compared against benchmarked traditional decoders. Notably, the exceptionally low latency of SNNs and LSTMs, surpassing that of traditional decoders and non-RNNs, arises from their innate ability to extract temporal information from spiking neural data. The power efficiency can be attributed to the adeptness of SNNs in exploiting sparsity and their deliberately constrained architectural design. Our results show that SNNs can achieve competitive decoding performance in less than 5 ms, using less than 1% of computational resources, and more than 50 times less energy than other neural decoding methods in this benchmark. This makes them highly suitable candidates for closed-loop iBCI challenges and positions them as a game-changing technology for reshaping the landscape of neural decoders. Significant advancements in CLN can be achieved by adopting SNNs as the preferred neural decoder. Their capacity for efficient and accurate neural signal processing holds the potential to revolutionize BCI applications, enhancing our ability to interact with and understand the intricacies of the human brain.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: https://zenodo.org/records/583331.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement No. 101001448)

ORCID iDs

Paul Hueber https://orcid.org/0009-0002-7800-8590
Guangzhi Tang https://orcid.org/0000-0002-0204-9225
Hua-Peng Liaw https://orcid.org/0009-0003-4721-3556
Nergis Tomen https://orcid.org/0000-0003-3916-1859
Yao-Hong Liu https://orcid.org/0000-0002-3256-6741

References

[1] Li M, Ruan H, Qi Y, Guo T, Wang P and Pan G 2019 Odor recognition with a spiking neural network for bioelectronic nose *Sensors* **19** 993
[2] Moses D A *et al* 2021 Neuroprosthesis for decoding speech in a paralyzed person with anarthria *New Engl. J. Med.* **385** 217–27
[3] Tang J, LeBel A, Jain S and Huth A G 2023 Semantic reconstruction of continuous language from non-invasive brain recordings *Nat. Neurosci.* **26** 858–66
[4] Zhou A, Johnson B C and Muller R 2018 Toward true closed-loop neuromodulation: artifact-free recording during stimulation *Curr. Opin. Neurobiol.* **50** 119–27
[5] Zanos S 2019 Closed-loop neuromodulation in physiological and translational research *Cold Spring Harb. Perspect. Med.* **9** a034314
[6] Dethier J, Nuyujukian P, Ryu S I, Shenoy K V and Boahen K 2013 Design and validation of a real-time spiking-neural-network decoder for brain–machine interfaces *J. Neural Eng.* **10** 036008
[7] Ciliberti D, Michon F and Kloosterman F 2018 Real-time classification of experience-related ensemble spiking patterns for closed-loop applications *eLife* **7** e36275
[8] Petrof I, Viaene A N and Sherman S M 2015 Properties of the primary somatosensory cortex projection to the primary motor cortex in the mouse *J. Neurophysiol.* **113** 2400–7
[9] Pei F *et al* 2021 Neural latents benchmark'21: evaluating latent variable models of neural population activity (arXiv:2109.04463)
[10] Mattson P *et al* 2020 MLPerf: an industry standard benchmark suite for machine learning performance *IEEE Micro* **40** 8–16
[11] Glaser J I, Benjamin A S, Chowdhury R H, Perich M G, Miller L E and Kording K P 2020 Machine learning for neural decoding *eneuro* **7** ENEURO.0506-19.2020
[12] Yik J *et al* 2023 NeuroBench: advancing neuromorphic computing through collaborative, fair and representative benchmarking (arXiv:2304.04640)
[13] Delgado J M R, Hamlin H and Chapman W P 1952 Technique of intracranial electrode implacement for recording and stimulation and its possible therapeutic value in psychotic patients *Stereotact Funct. Neurosurg.* **12** 315–9
[14] Evarts E V 1966 Pyramidal tract activity associated with a conditioned hand movement in the monkey *J. Neurophysiol.* **29** 1011–27
[15] Steinmetz N A *et al* 2021 Neuropixels 2.0: a miniaturized high-density probe for stable, long-term brain recordings *Science* **372** eabf4588

[16] Schwarz D A et al 2014 Chronic, wireless recordings of large-scale brain activity in freely moving rhesus monkeys Nat. Methods 11 670–6
[17] Marmerstein J T, McCallum G A and Durand D M 2022 Decoding vagus-nerve activity with carbon nanotube sensors in freely moving rodents Biosensors 12 114
[18] Lebedev M A et al 2011 Future developments in brain-machine interface research Clinics 66 25–32
[19] Lebedev M A and Nicolelis M A L 2011 Toward a whole-body neuroprosthetic Progress in Brain Research vol 194 (Elsevier) pp 47–60
[20] Makin J G, O'Doherty J E, Cardoso M M B and Sabes P N 2018 Superior arm-movement decoding from cortex with a new, unsupervised-learning algorithm J. Neural Eng. 15 026010
[21] Moran D W and Schwartz A B 1999 Motor cortical representation of speed and direction during reaching J. Neurophysiol. 82 2676–92
[22] Kumarasinghe K, Kasabov N and Taylor D 2021 Brain-inspired spiking neural networks for decoding and understanding muscle activity and kinematics from electroencephalography signals during hand movements Sci. Rep. 11 2486
[23] Hotson G, Smith R J, Rouse A G, Schieber M H, Thakor N V and Wester B A 2016 High precision neural decoding of complex movement trajectories using recursive Bayesian estimation with dynamic movement primitives IEEE Robot. Autom. Lett. 1 676–83
[24] Liu Y, Nour M M, Schuck N W, Behrens T E J and Dolan R J 2022 Decoding cognition from spontaneous neural activity Nat. Rev. Neurosci. 23 204–14
[25] Yang Q, Walker E, Cotton R J, Tolias A S and Pitkow X 2021 Revealing nonlinear neural decoding by analyzing choices Nat. Commun. 12 6557
[26] Dong Y, Wang S, Huang Q, Berg R W, Li G and He J 2023 Neural decoding for intracortical brain–computer interfaces Cyborg Bionic Syst. 4 0044
[27] Mridha M F, Das S C, Kabir M M, Lima A A, Islam M R and Watanobe Y 2021 Brain-computer interface: advancement and challenges Sensors 21 5746
[28] Klaes C 2018 Invasive brain-computer interfaces and neural recordings from humans Handbook of Behavioral Neuroscience vol 28 (Elsevier) pp 527–39
[29] Scangos K W et al 2021 Closed-loop neuromodulation in an individual with treatment-resistant depression Nat. Med. 27 1696–700
[30] Valencia D and Alimohammad A 2022 Towards in vivo neural decoding Biomed. Eng. Lett. 12 185–95
[31] Luo Y, Teng K-H, Li Y, Mao W, Lian Y and Heng C-H 2019 A 74- μ W 11-Mb/s wireless vital signs monitoring SoC for three-lead ECG, respiration rate, and body temperature IEEE Trans. Biomed. Circuits Syst. 13 907–17
[32] Kim H, Kim S, Van Helleputte N, Artes A, Konijnenburg M, Huisken J, Van Hoof C and Yazicioglu R F 2014 A configurable and low-power mixed signal SoC for portable ECG monitoring applications IEEE Trans. Biomed. Circuits Syst. 8 257–67
[33] He Y et al 2022 An implantable neuromorphic sensing system featuring near-sensor computation and send-on-delta transmission for wireless neural sensing of peripheral nerves IEEE J. Solid-State Circuits 57 3058–70
[34] Simeral J D et al 2021 Home use of a percutaneous wireless intracortical brain-computer interface by individuals with tetraplegia IEEE Trans. Biomed. Eng. 68 2313–25
[35] Even-Chen N, Muratore D G, Stavisky S D, Hochberg L R, Henderson J M, Murmann B and Shenoy K V 2020 Power-saving design opportunities for wireless intracortical brain–computer interfaces Nat. Biomed. Eng. 4 984–96
[36] Sriram K et al 2023 SCALO: an accelerator-rich distributed system for scalable brain-computer interfacing Proc. 50th Annual Int. Symp. on Computer Architecture (ACM) pp 1–20
[37] Stevenson I H and Kording K P 2011 How advances in neural recording affect data analysis Nat. Neurosci. 14 139–42
[38] Wolf P D 2008 Thermal considerations for the design of an implanted cortical brain–machine interface (BMI) Indwelling Neural Implants: Strategies for Contending with the in Vivo Environment ed W M Reichert (CRC Press/Taylor & Francis)
[39] Velliste M, Perel S, Spalding M C, Whitford A S and Schwartz A B 2008 Cortical control of a prosthetic arm for self-feeding Nature 453 1098–101
[40] VanRullen R and Reddy L 2019 Reconstructing faces from fMRI patterns using deep generative neural networks Commun. Biol. 2 193
[41] Dash D, Ferrari P and Wang J 2020 Decoding imagined and spoken phrases from non-invasive neural (MEG) signals Front Neurosci. 14 290
[42] Pandarinath C et al 2018 Inferring single-trial neural population dynamics using sequential auto-encoders Nat. Methods 15 805–15
[43] Todorova S, Sadtler P, Batista A, Chase S and Ventura V 2014 To sort or not to sort: the impact of spike-sorting on neural decoding performance J. Neural Eng. 11 056005
[44] Nason S R et al 2020 A low-power band of neuronal spiking activity dominated by local single units improves the performance of brain–machine interfaces Nat. Biomed. Eng. 4 973–83
[45] Taeckens E, Dong R and Shah S 2022 A biologically plausible spiking neural network for decoding kinematics in the hippocampus and premotor cortex (https://doi.org/10.1101/2022.11.09.515838)
[46] Liao J et al 2022 An energy-efficient spiking neural network for finger velocity decoding for implantable brain-machine interface 2022 IEEE 4th Int. Conf. on Artificial Intelligence Circuits and Systems (AICAS) (IEEE) pp 134–7
[47] Li S, Li J and Li Z 2016 An improved unscented Kalman filter based decoder for cortical brain-machine interfaces Front Neurosci. 10 232582
[48] Sze V, Chen Y H, Yang T J and Emer J S 2020 Efficient Processing of Deep Neural Networks (Springer International Publishing) (https://doi.org/10.1007/978-3-031-01766-7)
[49] Liu Z, Xiao X, Li C, Ma S and Rangyu D 2022 Optimizing convolutional neural networks on multi-core vector accelerator Parallel Comput. 112 102945
[50] Horowitz M 2014 1.1 Computing's energy problem (and what we can do about it) 2014 IEEE Int. Solid-State Circuits Conf. Digest of Technical Papers (ISSCC) (IEEE) pp 10–14
[51] Liang Y P et al 2021 Brief industry paper: an energy-reduction on-chip memory management for intermittent systems 2021 IEEE 27th Real-Time and Embedded Technology and Applications Symp. (RTAS) (IEEE) pp 429–32
[52] Shaeri M, Afzal A and Shoaran M 2022 Challenges and opportunities of edge AI for next-generation implantable BMIs 2022 IEEE 4th Int. Conf. on Artificial Intelligence Circuits and Systems (AICAS) (IEEE) pp 190–3
[53] Karageorgos I et al 2020 Hardware-software co-design for brain-computer interfaces 2020 ACM/IEEE 47th Annual Int. Symp. on Computer Architecture (ISCA) (IEEE) pp 391–404
[54] Shin U, Ding C, Zhu B, Vyza Y, Trouillet A, Revol E C M, Lacour S P and Shoaran M 2022 NeuralTree: a 256-channel 0.227-μJ/class versatile neural activity classification and closed-loop neuromodulation SoC IEEE J. Solid-State Circuits 57 3243–57

[55] An H, Nason-Tomaszewski S R, Lim J, Kwon K, Willsey M S, Patil P G, Kim H-S, Sylvester D, Chestek C A and Blaauw D 2022 A power-efficient brain-machine interface system with a sub-mw feature extraction and decoding ASIC demonstrated in nonhuman primates *IEEE Trans. Biomed. Circuits Syst.* **16** 395–408
[56] Boi F, Moraitis T, De Feo V, Diotalevi F, Bartolozzi C, Indiveri G and Vato A 2016 A bidirectional brain-machine interface featuring a neuromorphic hardware decoder *Front Neurosci.* **10** 215903
[57] Dyer E L, Gheshlaghi Azar M, Perich M G, Fernandes H L, Naufel S, Miller L E and Körding K P 2017 A cryptography-based approach for movement decoding *Nat. Biomed. Eng.* **1** 967–76
[58] Wang W, Chan S S, Heldman D A and Moran D W 2007 Motor cortical representation of position and velocity during reaching *J. Neurophysiol.* **97** 4258–70
[59] Wan E A and Van Der Merwe R 2000 The unscented Kalman filter for nonlinear estimation *Proc. IEEE 2000 Adaptive Systems for Signal Processing, Communications and Control Symp. (Cat. No.00EX373)* (IEEE) pp 153–8
[60] Pan V Y, Soleymani F and Zhao L 2018 An efficient computation of generalized inverse of a matrix *Appl. Math. Comput.* **316** 89–101
[61] Rumelhart D E, McClelland J L and Feldman J 1987 *Parallel Distributed Processing Explorations in the Microstructure of Cognition: Foundations* (A Bradford Book)
[62] Eshraghian J K *et al* 2021 Training spiking neural networks using lessons from deep learning (arXiv:2109.12894)
[63] Tavanaei A, Ghodrati M, Kheradpisheh S R, Masquelier T and Maida A 2019 Deep learning in spiking neural networks *Neural Netw.* **111** 47–63
[64] Göltz J *et al* 2021 Fast and energy-efficient neuromorphic deep learning with first-spike times *Nat. Mach. Intell.* **3** 823–35
[65] Diehl P U *et al* 2015 Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing *2015 Int. Joint Conf. on Neural Networks (IJCNN)* (IEEE) pp 1–8
[66] Neil D, Pfeiffer M and Liu S-C 2016 Learning to be efficient: algorithms for training low-latency, low-compute deep spiking neural networks *Proc. 31st Annual ACM Symp. on Applied Computing* (ACM) pp 293–8
[67] Rathi N and Roy K 2023 DIET-SNN: a low-latency spiking neural network with direct input encoding and leakage and threshold optimization *IEEE Trans. Neural Netw. Learn. Syst.* **34** 3174–82
[68] Willsey M S, Nason-Tomaszewski S R, Ensel S R, Temmar H, Mender M J, Costello J T, Patil P G and Chestek C A 2022 Real-time brain-machine interface in non-human primates achieves high-velocity prosthetic finger movements using a shallow feedforward neural network decoder *Nat. Commun.* **13** 6899
[69] Shaikh S, So R, Sibindi T, Libedinsky C and Basu A 2019 Towards intelligent intra-cortical BMI (i ^{2} BMI): low-power neuromorphic decoders that outperform Kalman filters (https://doi.org/10.1101/772988)
[70] Li X, Chen S, Hu X and Yang J 2019 Understanding the disharmony between dropout and batch normalization by variance shift *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* (IEEE) pp 2677–85
[71] Pascanu R, Mikolov T and Bengio Y 2013 On the difficulty of training recurrent neural networks (arXiv:1211.5063)
[72] Bengio Y, Simard P and Frasconi P 1994 Learning long-term dependencies with gradient descent is difficult *IEEE Trans. Neural Netw.* **5** 157–66
[73] Zenke F *et al* 2021 Visualizing a joint future of neuroscience and neuromorphic engineering *Neuron* **109** 571–5
[74] Maass W 1997 Networks of spiking neurons: the third generation of neural network models *Neural Netw.* **10** 1659–71
[75] Olshausen B A and Field D J 2006 What is the other 85% of V1 doing? *23 Problems in Systems Neuroscience* ed J L Van Hemmen and T J Sejnowski (Oxford University Press) pp 182–212
[76] Tang G *et al* 2023 Open the box of digital neuromorphic processor: Towards effective algorithm-hardware co-design (arXiv:2303.15224)