SCALO: An Accelerator-Rich Distributed System for Scalable Brain-Computer Interfacing

Karthik Sriram*
karthik.sriram@yale.edu
Yale University

Raghavendra Pradyumna Pothukuchi*
raghav.pothukuchi@yale.edu
Yale University

Michał Gerasimiuk
michal.gerasimiuk@yale.edu
Yale University

Muhammed Ugur
muhammed.ugur@yale.edu
Yale University

Oliver Ye
oliver.ye@yale.edu
Yale University

Rajit Manohar
rajit.manohar@yale.edu
Yale University

Anurag Khandelwal
anurag.khandelwal@yale.edu
Yale University

Abhishek Bhattacharjee
abhishek@cs.yale.edu
Yale University

ABSTRACT

SCALO is the first distributed brain-computer interface (BCI) consisting of multiple wireless-networked implants placed on different brain regions. SCALO unlocks new treatment options for debilitating neurological disorders and new research into brain-wide network behavior. Achieving the fast and low-power communication necessary for real-time processing has historically restricted BCIs to single brain sites. SCALO also adheres to tight power constraints, but enables fast distributed processing. Central to SCALO's efficiency is its realization as a full stack distributed system of brain implants with accelerator-rich compute. SCALO balances modular system layering with aggressive cross-layer hardware-software co-design to integrate compute, networking, and storage. The result is a lesson in designing energy-efficient networked distributed systems with hardware accelerators from the ground up.

CCS CONCEPTS

• **Hardware → Neural systems**; • **Computer systems organization → Heterogeneous (hybrid) systems**; *Real-time system architecture*; • **Computing methodologies → Distributed computing methodologies**.

KEYWORDS

Brain-Computer Interfaces, BCI, Hardware Accelerators, Low Power

*Karthik Sriram is the lead Ph.D. student on SCALO. This research is part of Karthik's dissertation thesis. Raghavendra Pradyumna Pothukuchi is an Associate Research Scientist and the lead advisor on this research. To highlight the fact that Karthik's and Raghavendra's contributions supersede the contributions of the other authors, we are acknowledging them as joint first authors.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
ISCA '23, June 17–21, 2023, Orlando, FL, USA.
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0095-8/23/06.
https://doi.org/10.1145/3579371.3589107

ACM Reference Format:
Karthik Sriram, Raghavendra Pradyumna Pothukuchi, Michał Gerasimiuk, Muhammed Ugur, Oliver Ye, Rajit Manohar, Anurag Khandelwal, and Abhishek Bhattacharjee. 2023. SCALO: An Accelerator-Rich Distributed System for Scalable Brain-Computer Interfacing. In *Proceedings of the 50th Annual International Symposium on Computer Architecture (ISCA '23), June 17–21, 2023, Orlando, FL, USA.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3579371.3589107

1 INTRODUCTION

Brain-computer interfaces (BCIs) connect biological neurons in the brain with computers and machines. BCIs are advancing our understanding of the brain [6, 46, 68], helping treat neurological/neuropsychiatric disorders, and helping restore lost sensorimotor function [22, 48, 60, 68, 76, 120, 155]. BCIs are also enabling novel human-machine interactions [137] with new applications in industrial robotics [173] and personal entertainment [86].

BCIs sense and/or stimulate the brain's neural activity using either wearable surface electrodes, or through surgically implanted surface and depth electrodes [68]. BCIs have historically simply relayed the neural activity picked up by electrode sensors to computers that process or "decode" that neural activity [22, 68]. But, emerging neural applications increasingly benefit from BCIs that also include processing capabilities. Such BCIs enable continuous and autonomous operation without tethering [6, 22, 23, 68, 124, 155].

In this work, we focus on the design of processors for surgically implanted BCIs that are at the cutting edge of neural engineering. Although they pose surgical risks, implanted BCIs collect far higher fidelity neural signals than wearable BCIs [9, 101]. Consequently, implantable BCIs are used in state-of-the-art research applications [22, 68, 155] and have been clinically approved to treat epilepsy and Parkinson's disease [78, 133, 144, 151], show promise (via clinical trials) in restoring movement to paralyzed individuals, offer a path to partially restoring vision to visually-impaired individuals, and more [43, 109, 143, 151].

Implantable BCI processors are challenging to design. They are limited to only a few milliwatts of power as overheating the brain by just >1 °C risks damaging cellular tissue [115, 161]. At the same

time, implantable BCIs are expected to process exponentially growing volumes of neuronal data [131] within milliseconds [163, 175]. Most modern BCIs [7, 24, 57, 78, 96, 133] achieve low power by specializing to a single task [7, 24] and by sacrificing neural processing data rates [57, 78, 96, 133]. Neither option is ideal. BCIs should instead be flexible, so that algorithms on board can be personalized to individuals and so that many new and existing algorithms can be supported [45, 53, 104, 159, 172]. And, BCIs should process higher data rates to infer more about the brain. To achieve these goals, we recently proposed HALO, an accelerator-rich processor that achieves low power at neural data rates orders of magnitude higher than prior work (46Mbps), but also achieves flexibility via programmable inter-accelerator dataflow [53, 54, 130].

While HALO successfully balances power, data rate, and flexibility, it interfaces with only a single brain site, whereas future BCIs will consist of distributed implants that *interface with multiple brain sites*. Applications that process neural data from multiple brain sites over multiple timescales are becoming common as neuroscience research is increasingly showing that the brain's functions (and disorders) are based on temporally-varying physical and functional connectivity among brain regions [6, 10, 135]. Assessing brain connectivity requires placing communicating implants in different brain regions, with storage that enables multi-timescale analysis. Unfortunately, no existing BCIs integrate adequate storage for such long-scale analysis. Even worse, communication is problematic. Because wired networks impose surgical risk and potential infection [152], wireless networking is desirable. Unfortunately, however, wireless networking offers lower data rates (10× lower than compute) under milliwatts of power.

We address these challenges by proposing and building *SCALO*, the first BCI architecture for multi-site brain interfacing in real time. SCALO is a distributed system of wirelessly networked implants. Each implant has a HALO processor augmented with storage and compute to support distributed BCI applications. SCALO includes an integer linear programming (ILP)-based scheduler that optimally maps applications to the accelerators and creates network/storage schedules to feed our hardware accelerators. SCALO has a programming interface that is easily plugged into widely-used signal processing frameworks like TrillDSP [91], XStream [37], and MATLAB [75]. SCALO continues to support HALO's single-implant applications [53], but also enables, for the first time, three new classes of distributed applications [88, 175].

The first class consists of *internal closed-loop* applications that modulate brain activity [175] without communicating with systems external to the BCI. These applications monitor multiple brain sites, and when necessary, respond autonomously with electrical stimulation. Examples include detection and treatment of epileptic seizure spread, essential tremor, and Parkinson's disease [10, 45].

The second class consists of *external closed-loop* applications where BCIs communicate with systems external to the brain and BCI [88, 175]. Examples include neural prostheses for speech and brain-controlled screen control devices [51, 100, 160, 163, 174].

The third class consists of *interactive human-in-the-loop* applications where clinicians query the BCI for data or dynamically adjust processing/stimulation parameters [66, 126]. This is useful to validate BCI detection of seizures [66], personalize stimulation algorithms to individuals [159], or debug BCI operation.

SCALO achieves ultra power-efficient operation by tightly co-designing compute with storage, networking, scheduling, and application layers. We use knowledge of neural decoding methods to reduce communication between implants comprising the distributed BCI by: (1) building locality-sensitive hash measures to filter candidates for expensive signal similarity analysis across implants; (2) reducing data dimensionality by hierarchically splitting computations in classifiers and neural networks; and, unusually, (3) by centralizing rather than distributing key computations when appropriate (e.g., like matrix inversion in our applications).

SCALO consists of hardware accelerators or processing elements (PEs) to support (1)-(3) above with low latency and power. We build the PEs so that they can be reconfigured to realize many applications, and compose them in a GALS (Globally Asynchronous Locally Synchronous) architecture [53]. By realizing each PE in its independent clock domain, we allow it to be tuned for the minimal power to sustain a given application-level processing rate. We use per-implant non-volatile memory (NVM) to store prior signals and hash data. Our storage layout is optimized for PE access patterns.

SCALO also consists of per-implant radios that support an ultra-wideband (UWB) wireless network. We build our PEs to directly access the network and storage, avoiding the bottlenecks that traditional accelerator-based systems (including ultra-low-power coarse-grained reconfigurable arrays or CGRAs [38, 39, 92, 134, 140, 153]) suffer in relying on CPUs to orchestrate data movement.

SCALO's components are predictable in latency and power, facilitating optimal compute/network scheduling with an ILP. For PEs whose output data generation rates are based on input patterns (e.g., data compression), our ILP uses worst-case bounds.

We evaluate SCALO with a physical synthesis flow in a 28 nm CMOS process coupled with network and storage models. Our evaluations are supported by prior partial chip tape-outs of HALO in a 12 nm CMOS process. SCALO achieves an aggregate neural interfacing data rate of 506 Mbps using 11 implants to assess and arrest seizure propagation within 10 ms of seizure onset; 188 Mbps using 4 implants to relay intended movements to external prostheses within 50 ms and restore sensorimotor function; and sorts 12,250 spikes per second per site with a latency of 2.5 ms. All applications expend less than 15 mW per implant. When used for interactive querying, SCALO supports 9 queries per second over 7 MB of data over 11 implants. Overall, our contributions include:

(1) A full-stack accelerator-rich distributed BCI, with unusually tight integration of compute with network and storage.
(2) The design of an optimal ILP-scheduler for mapping applications across distributed accelerators and network, enabled by a deterministic compute, network and storage design.
(3) An interface facilitating easy integration into existing data and signal processing platforms.

These technical contributions, in turn, translate into advances in neural decoding and computer systems design:

(1) ***Neural Decoding:*** The first distributed wireless BCI processing architecture for decoding, analysis, and electrical stimulation of brain-wide networks. SCALO offers the first on-device support for seizure propagation and movement intent analysis on multiple brain sites. SCALO includes configurable on-device locality sensitive hashing for fast signal similarity analysis.

(2) ***Computer Systems:*** An experiment in the design of an end-to-end distributed system of accelerators from the application layer to physical synthesis. Our evaluation shows 10–385× higher processing rates over prior work at 15 mW per implant.

2 BACKGROUND

2.1 Components of a BCI

BCI applications consist of signal measurement, feature extraction, classification/decision-making, and when applicable, neural feedback/stimulation [22, 68, 120]. BCIs consist of hardware components mirroring each of these four stages.

Signal measurement is performed by electrodes that read the electrical activity of neurons and analog-to-digital converters (ADCs) that digitize these signals. Arrays of 96–256 [14] electrodes or depth probes of 1–4 electrodes [13] are widely used. BCI ADCs typically sample at 5–50 KHz per electrode with 8–16 bit resolution [34, 160].

Feature extraction and classification/decision-making are performed on the digitized data. These portions of the neural pipeline were historically undertaken by external servers, but on-BCI computation is becoming increasingly important [6, 23, 25, 68, 124, 155]. When neural feedback is needed, the electrodes are repurposed (after digital-to-analog conversion) to electrically stimulate the brain. Electrical stimulation can, for example, mitigate seizure symptoms.

Traditional BCI communication with external server/prostheses relied on wires routed through a port embedded in the skull [17, 152, 157]. But, wiring restricts the individual's movement, hinders convenience, and is susceptible to infections and cerebrospinal fluid leaks [152]. Wireless radios avoid these issues and are consequently being used more widely [6, 61, 124, 152, 166].

Some BCIs use batteries that are implanted and single-use [133] or externally removable [17]. Recent BCIs are using implanted rechargeable batteries with inductive power transfer [5, 44, 47].

Taken together, all these components are packaged in hermetically-fused silica or titanium capsules. While safe power limits depend on implantation location and depth [122, 123], we use 15 mW per implant as a conservative limit [40, 44, 53, 58, 122, 128, 147].

2.2 BCI Applications & Kernels

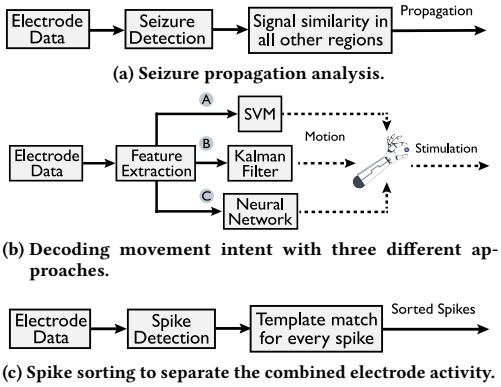
The space of BCI applications is rapidly growing [6, 68, 155]. Some require neural data from only a single brain region (e.g., spike sorting [16]) while many others (e.g., epileptic seizure propagation and movement intent decoding) require neural data from multiple brain regions [6, 10, 135]. We target three classes of distributed BCI applications that operate in autonomous closed loops [88, 175]. From each class, we study a representative application. Additionally, we also study spike sorting, a kernel that is often used to pre-preprocess neural data before subsequent application pipelines [16].

Internal closed-loop applications: Nearly 25 million individuals worldwide suffer from drug-resistant epilepsy and experience seizures that last tens of minutes to hours [29, 142, 164] per day. BCI-led closed-loop therapy can help these individuals immensely [48, 57]. SCALO supports epileptic seizure propagation calculations on device, which many recent studies show as being desirable [10, 15, 62, 104, 127]. Seizure propagation applications correlate neural signals from brain regions where seizures originate to current and historical signals from other brain sites [10, 52]. Correlations help identify the network dysfunction that underlies seizure spread, which in turn unlocks targeted treatment options [10, 65].

Figure 1a illustrates seizure propagation analysis [10, 52]. First, seizures are detected "locally" in each brain site. This is done with band-pass filtering and/or the fast Fourier transform (FFT), which generate features from contiguous time windows of neural data, and then using classifiers like support vector machines (SVMs) [119].

When a seizure is detected at a brain site, its neural data is correlated with recent and past neural signals from other brain sites. Many measures are used to determine correlation, including dynamic time warping (DTW), Euclidean distance, cross-correlation, and Earth Mover's Distance (EMD) [10, 70, 83]. Once correlated brain regions are identified and seizure spread is forecast, brain regions anticipating seizure spread are electrically stimulated to mitigate the spread.

Treatment effectiveness depends on accurate but also *timely* seizure forecasting [63, 99]. In consultation with the clinicians and researchers that we collaborate with at the Yale School of Medicine, we set a challenging 10 ms target from local seizure detection to seizure forecasting and electrical stimulation.



(a) Seizure propagation analysis.

(b) Decoding movement intent with three different approaches.

(c) Spike sorting to separate the combined electrode activity.

Figure 1: Overview of BCI applications supported by SCALO.

External closed-loop applications: These applications help individuals control assistive devices external to BCIs like artificial limbs [4, 20, 95, 125, 154, 168], cursors on computer screens [36, 93, 94, 158], or prostheses that translate brain signals corresponding to intended speech into text on computer screens [51, 100, 160, 163]. We select three neural processing algorithms representative of this category of applications and illustrate them in Figure 1b.

Pipeline Ⓐ classifies neural activity into one of a preset number of limb movements like finger pointing, arm stretch, and more [95, 125]. The features are extracted using FFT and filters, and used by a classifier to identify movement. Linear SVMs are most commonly used for classification [35, 60, 71, 87, 117]. More complex deep neural networks (DNNs) have been shown to outperform SVMs and are promising [125]. For now, SCALO supports linear SVMs and shallow networks as they require less training data than DNNs, have more intuitive parameter tuning, and are more interpretable [35, 60, 71]. We will study SCALO support for DNNs in future work.

Unlike pipeline Ⓐ (which identifies complex movements as a whole), pipelines Ⓑ and Ⓒ decode the position and velocity of arm/finger movements or cursor movements on screen [160, 163].

Pipelines Ⓑ and Ⓒ calculate spike band power in neural signals by taking the mean value of all neural signals in a time window (typically 50 ms). Pipelines may use a variant of Kalman Filter (Ⓑ) [163] or a shallow neural network (Ⓒ) [160] to decode movement intent.

Decoded intended movement is relayed to computer screens, artificial limbs, or even paralyzed limbs implanted with electrodes [4]. When the individual has also lost sensory function, the "feeling" of movement is emulated by relaying the impact of the movement back to the individual's BCI. The BCI then electrically stimulates relevant brain sites to emulate sensory function [4, 11, 23, 30, 68, 136, 146]. The entire movement decoding loop must complete within 50 ms [160] to effectively restore sensorimotor control.

Human-in-the-loop applications: Researchers, or clinicians wish to interactively query BCI devices. They may retrieve important neural data, configure device parameters for personalization, or verify correct operation. Low query latency is not just desirable, but often necessary. For example, a clinician may need to retrieve neural data and manually confirm that the BCI correctly detected a seizure. Or, a clinician may plan to test the effectiveness of a new electrical stimulation protocol for treatment. Faster device querying measurably improves BCI utility in such cases [66, 126]. It is important, however, that interactive querying does not disrupt the other BCI applications that are continuously running.

Spike sorting kernel: Each electrode usually measures the combined electrical activity of a cluster of spatially-adjacent neurons. This combined activity is also influenced by sensor time lag, signal attenuation, and sensor drift [16]. The goal of spike sorting is to separate combined neural activity into per-neuron waveforms. Unlike the applications discussed thus far, spike sorting is entirely local to each brain site. But, it is a widely used first step for important BCI applications that rely on neuron-level analysis [16, 79, 90, 107, 110]. In fact, spike sorting would also benefit other applications like movement intent decoding if it could be made faster (today, the prohibitive cost of spoke sorting prompts usage of approximated sorting [26, 139, 141]). SCALO offers power-efficient spike-sorting within a few milliseconds to fully unlock its potential.

Figure 1c shows a typical spike sorting pipeline. Spike waveforms are detected from electrode signals. They are then matched with templates corresponding to each neuron. Such templates may be obtained offline from prior recordings or generated online with clustering [112]. Spike waveforms are matched with templates using some of the same compute-intensive correlation measures from seizure propagation pipelines; e.g., DTW and EMD [19, 42, 129].

2.3 BCI Design Challenges

BCIs cannot exceed 15 mW and have tight response times (10 ms for seizure propagation, 50 ms for movement decoding, a few 100 ms for interactive querying, and a few milliseconds for spike sorting). Distributed processing is challenging because inter-implant communication radios have low data rates, and do not use multiple frequencies (to save power), requiring serial network access [108].

Overspecializing hardware to achieve low power is undesirable. Neural signaling differs across brain regions and across subjects [45, 159]. Signaling even evolves over time, and as a consequence of the brain's response to the implant [45, 132, 159]. No single processing algorithm and parameters is optimal for the application pipelines in Section 2.2. Instead, these pipelines must be customized to the implant site, the individual, and must be regularly re-calibrated.

Distributed BCIs heighten this tension in power and flexibility. The few distributed multi-site BCIs that have been built to date [3, 69, 135] use multiple sensor implants that offload processing to external computers [3, 69], but do not support on-BCI processing. This restricts their scope and timeliness [174].

SCALO is the first distributed multi-site BCI to offer on-BCI processing. One may initially expect thermal coupling between the implants in SCALO to restrict per-implant power budgets below the 15 mW target of single-site BCIs like HALO. As we detail in Section 5, however, the brain's cerebrospinal fluid and blood flow dissipate heat effectively on the cortex, making thermal coupling negligible even with relatively short inter-implant spacing.

2.4 Locality-Sensitive Hashing

While inter-implant thermal coupling is less of a concern, inter-implant communication latency becomes the barrier to the design of a wireless-networked multi-site BCI. In response, we lean on locality sensitive hashing (LSH) [50], a technique used for fast signal matching [59]. LSH offers a way to filter inter-implant communication to only those neural signals most likely to be correlated (as determined by similarity measures like DTW, EMD, etc.).

We base SCALO's design on prior LSH work for DTW [72] and EMD [41]. LSH approaches for DTW [72] first create sketches of neural signals by calculating the dot product of sliding windows in the signal with a random vector. The sketch of a window is 1 if the dot product is positive and 0 otherwise. Then, the occurrences of all n-grams formed by n consecutive sketches are counted. The n-grams and their counts are used by a randomized weighted min-hash step to produce the hash. The LSH for EMD [41] calculates the dot product of the entire signal with a random vector, and then computes a linear function of the dot product's square root.

3 THE SCALO ARCHITECTURE

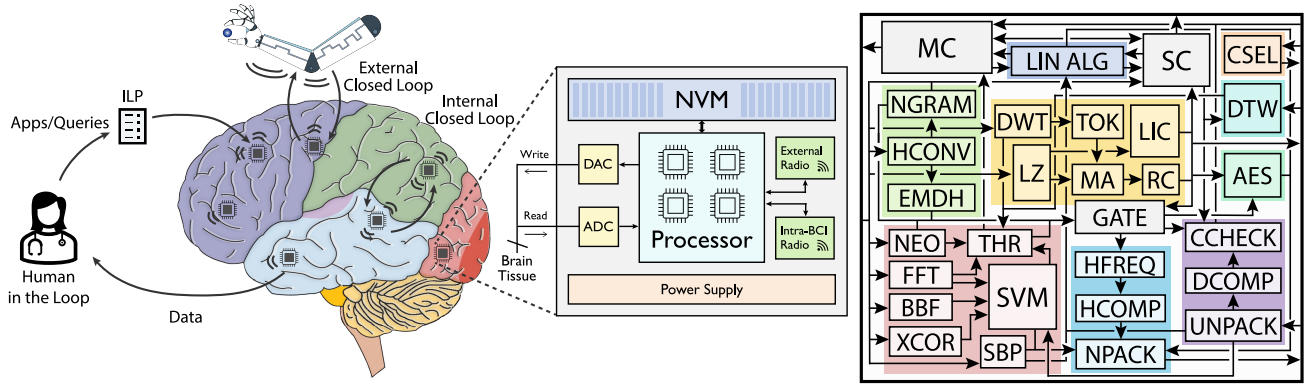
Figure 2 shows SCALO and its implants (or nodes). Each SCALO node contains 16-bit ADCs/DACs, an accelerator/PE-rich reconfigurable processor, an NVM layer, a radio for inter-node (intra-BCI) communication and another radio for external communication, as well as a power supply. SCALO can run various applications and interactive queries expressed widely-used high-level languages. An ILP scheduler maps their operations onto the nodes optimally.

3.1 On-BCI Distributed Neural Pipelines

Our first step is to convert the pipelines in Figure 1 into counterparts amenable for distributed processing. One enhancement is to enable the pipelines to use storage to assess correlations over multiple timescales. The more critical enhancement is to modify the pipeline to mitigate the inter-node communication bottleneck.

First, we split signal comparison into a fast hash check, and subsequent exact comparison. The hash check identifies neural data that is (in high probability) uncorrelated among brain regions, and hence unnecessary for inter-node exchange. Hashes are 100× smaller than signals, and can be quickly and accurately generated. They significantly filter compute and inter-node communication.

Second, we decompose classifiers like SVMs and neural networks (NNs) to reduce the dimension of data being communicated. Instead



(a) SCALO overview.

(b) The processor fabric in each of SCALO's nodes.

Figure 2: The SCALO BCI is a distributed network of nodes implanted in multiple brain sites. The nodes communicate wirelessly with each other and the environment. Each SCALO node has sensors, radios, analog/digital conversion, processing fabric, and storage; the processing fabric contains hardware accelerators and configurable switches to create different pipelines.

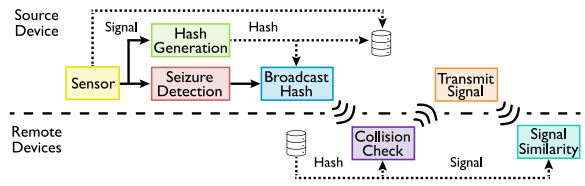
of the conventional approach of applying a classifier to all neural data from all brain sites, each of SCALO's nodes calculates a partial classifier output on its own data. All outputs are aggregated on a node to calculate the final result. Local classifier outputs are 100× smaller than the raw inputs; communicating the former rather than the latter reduces network usage significantly. Decomposing linear SVMs is trivial and does not affect accuracy. NNs are similarly decomposed by distributing the rows of the weight matrices.

Third, we centralize the matrix inversion operation used in the Kalman filter. The Kalman filter generates large matrices as intermediate products from lower-dimensional electrode features, and inverts one such matrix [160]. Distributing (and communicating) large matrices over our wireless (and serialized) network violates our response time goals (Section 2.3). Therefore, we directly send the electrode features from all sites to a single implant which computes the filter output, including the intermediate inversion step.

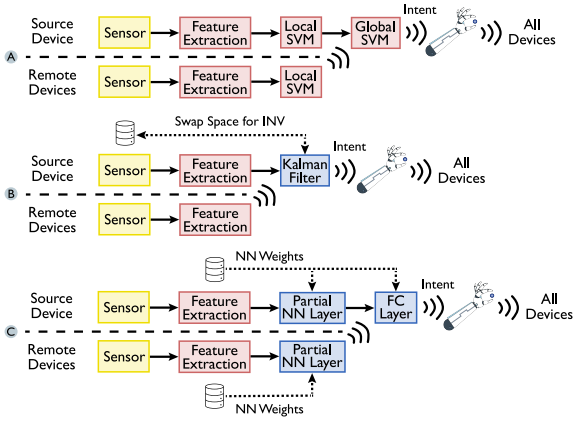
Figure 3a shows our new distributed seizure propagation analysis. Each electrode's samples are collected in a sliding window (e.g., 120 samples) and then used to generate a hash. Hashes are stored in the NVM. When a SCALO node detects a seizure on one or more signal windows, it broadcasts the corresponding hashes to other nodes. Receiver nodes check if these hashes match with any of their recently stored local hashes and respond on a match. The original node broadcasts the full signal window corresponding to the matching hash. Receiver nodes confirm seizure propagation by exactly comparing their local signals with the received ones. Finally, electrical stimulation can be applied at all locations with a seizure spread. Importantly, local per-node seizure detection (omitted in Figure Figure 3a) continues unabated during this correlation step.

Figure 3b shows our distributed movement decoding application. Algorithms Ⓐ and Ⓒ benefit from hierarchically decomposed SVMs and NNs. Each node computes a partial local output. A single node aggregates outputs and generates a final decision. In Algorithm Ⓑ, each node extracts features locally and transmits them to a node running the Kalman filter to decode movement intent.

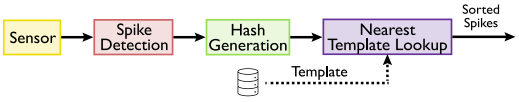
Finally, Figure 3c shows our online spike sorting pipeline. Spike sorting benefits from hash-based signal processing and storage. Spikes from the incoming signals are detected, and encoded with hashes. These hashes are compared with the hashes of templates that are locally stored in each node to classify the spike waveforms. Since spike sorting is a precursor to advanced processing [16], our fast online version can benefit many applications.



(a) Seizure propagation analysis.



(b) Decoding movement intent and stimulating as a response.



(c) Spike sorting.

Figure 3: High-level overview of the BCI applications supported for online distributed processing in SCALO.

3.2 Flexible & Energy-Efficient Accelerators

SCALO's nodes are based on an augmented version of our prior work, HALO [53]. SCALO's PEs can be reused across applications, and have deterministic latency/power. Wide-reuse PEs minimize design and verification effort, and on-chip area. Deterministic latency/power enables simple and optimal application scheduling.

Figure 2b shows the processor in each SCALO node. There are many PEs (functions described in the appendix) connected with programmable switches. The switches can be configured to realize various processing pipelines. Being a GALS design, it is easy to even configure PEs across multiple nodes in a pipeline. In addition to the PEs from our prior work for single-site applications [53], we incorporate new functionality to support distributed applications.

LSH support: We build hash support for four commonly-used signal similarity measures – Euclidean distance, cross-correlation (XCOR), DTW distance, and EMD [83].

Prior work has proposed an LSH specifically for DTW [72], but we discover that by varying the LSH's parameters, it can also serve as a hash for Euclidean distance and cross-correlation. Our discovery enables the design of a single LSH PE that can generate hashes for all three measures. To accommodate the LSH for EMD [41], we identify a shared dot product with the LSH for DTW (Section 2.4). In sum, we design three PEs to support all LSHs: dot product computation (HCONV), n-gram count and weighted min-hash (NGRAM), and square root (EMDH).

A crucial aspect of our LSH PEs is that the weighted min-hash calculation from prior work [72] uses a variable-latency randomization step. To guarantee deterministic latency and power while preserving the LSH property, we use an alternative method [55].

When hashes are received by a node for matching, they are sent to the CCHECK PE that stores them in SRAM registers and sorts them in place. The PE reads local hashes up to a configurable past time (e.g., 100 ms) from the on-chip storage, and checks for matches with the received hashes using binary search.

Signal comparison: We use PEs for selecting the signals to be broadcast (CSEL) and for comparison (DTW, XCOR). The DTW PE uses a pipelined implementation of the standard DTW algorithm [64] with a Sakoe-Chiba band parameter for faster computation [113]. The same PE measures Euclidean distance by setting the band parameter to 1. We reuse the XCOR PE from HALO [53].

EMD is more computationally expensive than all other measures, but we use a fast version [102] for which the on-chip general-purpose microcontroller (described later) was sufficient. We are currently investigating PE design for the full EMD calculation.

Linear Algebra Computations: While HALO originally integrated an SVM PE, distributed applications require more complex linear algebra (e.g., matrix multiplication and inversion) for which we build linear algebra PEs (LIN ALG). LIN ALG has PEs for matrix multiplication and addition with a constant matrix (MAD), matrix addition (ADD), subtraction (SUB), and inversion (INV). MAD can be configured to perform multiplication (MUL) only. All PEs use 16 KB registers with single-cycle access to store input matrices and constants. Larger entries can be read from the NVM.

Because SCALO does not support loops, applications with several MAD operations can be accelerated by either replicating MAD PEs or saving values to memory. For <10 MAD operations, the latency benefits of PE replication outweigh its hardware cost, and we use 10 MAD PEs in the LIN ALG cluster. Four MAD PEs are tiled into 4-way blocks to support large matrix operations found in the Kalman filter. We do not tile all MAD PE units since the remaining operations in the Kalman filter use smaller matrices.

We implement rectified linear activation (ReLU) and normalization operations used in NNs by adding configurable parameters to the MAD and ADD units. When the ReLU parameter is set, the units suppress negative outputs by replacing them with 0. When normalization is set, the units read the mean and standard deviation as parameters and normalize the output. We implement matrix inversion in hardware using the Gauss-Jordan elimination method [106].

Networking Support: The intra-SCALO network carries hashes and signals/signal features. As network data rate is low, compression increases transmitted item count. However, because compression reduces redundancy, it also increases susceptibility to network errors. We strike a balance based on the likelihood of errors.

Signals remain lengthy after compression (≈120–240 B) and can suffer errors for a given network bit error rate (BER). Measures like DTW are naturally resilient to errors for uncompressed signals, but lose accuracy using erroneous compressed signals. Signal features like those used to decode movement intent are lengthy and sensitive to errors when compressed. We therefore avoid compressing them.

Hash comparison also fails quickly with erroneous hashes, but such errors are 100× less likely because hashes are short even before compression (1–2 B). We therefore compress hashes. When hashes suffer an error, comparison can still proceed with subsequent hashes because brain signals at a site are temporally correlated. We show that it takes an unusually high BER to delay the application (e.g., seizure propagation) by 1 ms (Section 6.7).

HALO's PEs (i.e., LZ/LZMA) [53] were originally built to transmit large volumes of data to external servers and are not suitable for low-latency hash compression. We develop new PEs to compress intra-SCALO communication. The HFREQ PE collects each node's hash values and sorts them by frequency of occurrence. The HCOMP PE applies multiple compression algorithms serially. It first encodes the hashes with dictionary coding, then uses run-length encoding of the dictionary indexes [105], and finally uses Elias-γ coding [32] on the run-length counts. By customizing the compression strategy to the data, HCOMP achieves a compression ratio that is only 10% lower than that of LZ4/LZMA, but uses 7× less power.

Compressed data is sent to the NPACK PE, which adds checksums before transmission. There are UNPACK and DCOMP PEs to decode and decompress packets respectively, on the receiving side.

Storage Control: Access to the on-chip NVM is managed by the SC PE. This PE has SRAM to buffer writes before they are sent to the NVM as 4 KB pages and during erase operations. The SRAM is also used to reorganize the data layout (Section 3.3) to speedup future reads from the NVM. Finally, SC uses registers to store metadata (e.g., the last written page) to speedup recent data retrieval.

Microcontroller: SCALO has a RISC-V microcontroller, MC, for several operations. It configures PEs into pipelines, and runs neural stimulation commands. The MC is also used for computations not supported by any PEs such as new algorithms, or infrequently run

system operations such as clock synchronization (Section 3.6). The MC runs at a low frequency of 20 MHz and integrates 8 KB SRAM.

Optimal Power Tuning: Each of SCALO's PEs operates in its own clock domain, similar to our prior work on HALO [53]. However, HALO supported only one frequency per PE. This is not optimal for SCALO's applications, which sometimes operate on only on a subset of electrode data. For example, seizure propagation requires exact comparison for only a few signals to remain under target response times. Running PEs at only one target frequency even when input data rates may be lower, wastes power.

We add support for multiple frequencies per PE, and pick the lowest necessary to sustain a target data rate, minimizing power. SCALO's PEs support a frequency f_{max}^{PE}, high enough for the maximum data rate, and divide it to f_{max}^{PE}/k, where k is user-programmable. Division is achieved using a simple state machine and counter that passes through every k clock pulses. The counter consumes only μWs [12]. We use multiple frequency rails to ensure the PE has the same latency despite variable number of inputs.

3.3 Per-Implant NVM Storage

Each node integrates 128 GB NVM to store (in separate partitions) signals, hashes, and application data (e.g., weight matrices, spike templates). The MC uses a fourth NVM partition. Partition sizes are configurable. When full, the oldest partition data is overwritten.

We co-design the NVM data layout with PE access patterns to meet ms-scale response times. SCALO's ADCs and LSH PEs generate values sequentially by electrode. Stored as is, extracting a contiguous signal window of an electrode (used by most operations) requires reading from several discontinuous NVM locations. We reorganize neural data to store contiguous signals in chunks (with a configurable chunk size). This enables data retrieval with fast contiguous NVM reads. Our approach takes 5× longer for writes (1.75 ms), but is 10× faster for reads (0.035 ms). Data is written once but read multiple times, and writes are not on the critical path of execution, while reads are. These two factors make our approach more efficient. We reuse SC PE write buffers for this reorganization.

3.4 Networking

SCALO incorporates three networks. From our HALO work [53], we retain the inter-PE circuit switched network, and the wireless network to communicate with external devices up to 10 m. We add a new wireless network for intra-SCALO communication, using a custom protocol with TDMA [80]. Our ILP generates a fixed network schedule across all the nodes (Section 3.5).

The intra-SCALO network packets have an 84-bit header, and a variable data size up to a maximum packet size of 256 bytes. The header and data have 32-bit CRC32 checksums [103]. On an error, the receiver drops the packet if it contains hashes, but uses it if it has signals. This is because signal comparison measures like DTW are naturally resilient to a few errors. We evaluate the impact of allowing erroneous signal packets in Section 6.6.

3.5 Optimal System Scheduling

We use a software ILP-based scheduler to map tasks to PEs and generate storage and network schedules. The deterministic latency and power characteristics of our system components makes optimal software scheduling feasible. The scheduler takes as input the dataflow graph of applications and queries, constraints like the response time, and priorities of application tasks/stages (e.g., seizure detection versus signal comparison). A higher priority for a task ensures that more electrodes signals are processed in it relative to the others when all signals cannot be processed in all tasks.

Each task is modeled as a flow, and the ILP maximizes the number of electrodes processed in each flow. It ensures that overall response time and power constraints are met. It is acceptable for two flows to share the same PE. In this case, the signals from each flow are interleaved and run at a single frequency, completing within the same time as if they were run independently. The hardware tags the signals from each flow to route them to the correct destinations.

3.6 System Maintenance

Clock synchronization: SCALO's distributed processing requires the clocks in each BCI node to be synchronized with a precision of a few μs. The clocks we use are based on pausable clock generators and clock control units [85, 170] that suffer only picoseconds of clock uncertainty, a scale much smaller than our μs target. SCALO also operates at the temperature of the human body and does not experience clock drift due to temperature variance. Nevertheless, SCALO synchronizes clocks once a day using SNTP [82].

In SNTP, one SCALO node is designated as the server. All other nodes send messages to it to synchronize clocks. The clients send their previously synchronized clock times and current times, while the server sends its corresponding times. Clocks are adjusted based on the difference between these values. This process repeats until all the clocks are synchronized within the desired precision. During clock synchronization, the intra-SCALO network is unavailable for other operations, but tasks like seizure detection that do not need the network continue unimpeded.

Wireless Charging: Powering BCIs is an open problem, especially for distributed implants. We assume that the SCALO nodes are wirelessly powered, similar to prior demonstrations for distributed [67] and centralized sensor implants [2, 44, 124, 165, 171] (even though wired power delivery through a hub is also possible [97]). When charging wirelessly, we pause all pipelines to avoid overheating. While charging frequency and duration varies by algorithm and battery technology [124, 171], recent work has shown that it is possible to have 24-hour operation with 2 hours of charging [124].

3.7 Programming & Compilation

Figure 4 shows how SCALO is programmed. Clinicians or neuroscientists create programs in popular high-level languages like MATLAB [75] or TrillDSP [91] to describe signal processing pipelines or interactive queries. We support a subset of these languages to enable static scheduling (e.g., only fixed loop iterations).

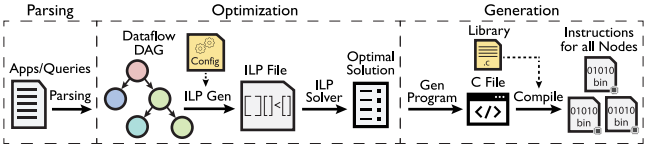


Figure 4: Programming and Interacting with SCALO.

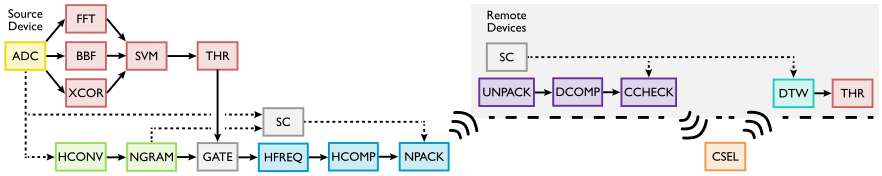


Figure 5: Seizure detection and propagation on SCALO. The colors of the PEs are matched with the high-level tasks from Figure 3a.

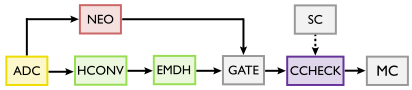


Figure 7: Spike sorting on SCALO.

Listing 1 shows movement intent decoding with a Kalman filter, and Listing 2 shows a complex interactive query, both written in TrillDSP. The query detects seizures from signals in the last 5 s at all nodes, and sends the data 100 ms before/after detected seizures.

```
var movements = stream.window(wsize=50ms).sbp()
                      .kf(kf_params).call_runtime()
```

Listing 1: Movement Intent using a Kalman filter in TrillDSP.

```
var seizure_data = stream.Map( // group by location
    s => s.select(s => s.data), s.locID)
    .window(wsize=4ms).select(w => w.time >= -5000).
        select(w => w.seizure_detect(), w[-100ms:100ms])
```

Listing 2: Interactively querying seizure data.

Programs are parsed into dataflow directed acyclic graphs (DAGs). A configuration file maintains details of the components (the power consumption of PEs, radio data rates, etc.), overall constraints, and priorities (Section 3.5). The DAG and configuration file are used to formulate an ILP, which is solved with standard software (e.g., [74]).

The optimal schedule from the ILP solver contains the mapping of tasks to PEs and the network schedule. It is translated to assembly instructions that can be run on the per-node MCs. Translation occurs in two steps. From the ILP output, we first generate a C program using a library of predefined functions to configure the parameters of the PEs and their connections. Next, the program and the library are compiled to obtain RISC-V binaries. We also develop a lightweight runtime on the MC that listens to the external radio for data and code, and reconfigures PEs and pipelines.

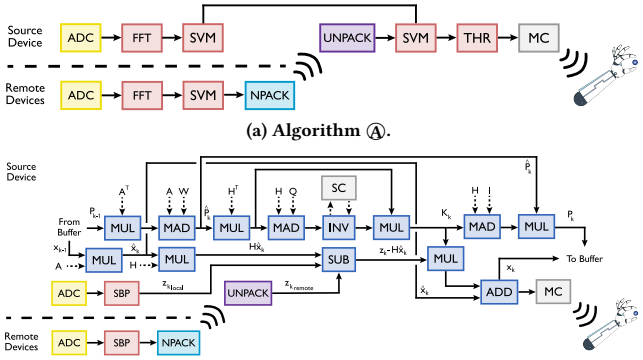
4 DEPLOYING SCALO

Figure 5 shows the seizure propagation pipeline on SCALO. The colors of the PEs are matched with the high level tasks from Figure 3. In this pipeline, seizure detection uses FFT, Butterworth bandpass filters (BBF) and XCOR for feature extraction, followed by an SVM for classification [119]. Signals are compared with DTW [70].

Figure 6 shows the movement intent pipelines on SCALO. Algorithm Ⓐ's implementation is derived from multiple sources [95, 117, 125]. Algorithms Ⓑ and Ⓒ are implemented based on prior designs [163] and [160], respectively. In our implementation of Algorithm Ⓑ, we do not change the Kalman filter parameters online as done in some variants [163] although SCALO supports it.

In Figure 6b, since the Kalman filter needs its output from the previous time step, we save this value to a buffer at the end of the pipeline. Additionally, the inversion operation (INV) needs to use the NVM because the matrix is too big to fit in the PE memory.

Last, Figure 7 shows online spike sorting using EMD hashes and templates, derived from prior work [112] and [42].



(a) Algorithm Ⓐ.

(b) Algorithm Ⓑ.

(c) Algorithm Ⓒ.

Figure 6: Movement intent on SCALO.

5 EXPERIMENTAL SETUP

Processing fabric: SCALO's PEs are designed and synthesized with Cadence tools at a commercial 28 nm fully-depleted silicon-on-insulator (FD-SOI) CMOS process. We use standard cell libraries from STMicroelectronic and foundry-supplied memory macros that are interpolated to 40 °C, which is close to human body temperature. We design each PE for its highest frequency, and scale the power when using it at lower frequency. We run multi-corner, physically-aware synthesis, and use latency and power measurements from the worst variation corner. Table 1 shows these values. We confirm these values with partial tape-outs at 12 nm [130]. In the table, blank entries indicate data-dependent latencies. The SC can take 0.03 or 0.04 ms depending on the NVM being available or busy.

We assume that each node uses a standard 96-electrode array [14] to sense neural activity, and has a configurable 16-bit ADC [118] running at 30 KHz per electrode. The ADC dissipates 2.88 mW for 1 sample from all 96 electrodes. Each node also has a DAC for electrical stimulation [77], which can consume ≈0.6 mW of power.

Radio parameters: We use a radio that transmits/receives up to 10 m with external devices at 46 Mbps, 250 MHz, and consumes 9.2 mW [53]. For intra-SCALO communication, we consider a state-of-the-art radio designed for safe implantation [108]. We modify the radio, originally designed for asymmetric transmit/receive, for symmetric communication. The radio can transmit up to 20 cm (>

Table 1: Latency and Power of the PEs.

Processing Elements	Max Freq (MHz)	Power (μW) Leakage (SRAM)	Dyn/Elec	Latency (ms)	Area (KGE)
ADD	3	0.08 (0.00)	0.983	2	68
AES	5	53 (0.00)	0.61	-	55
BBF	6	66.00 (19.88)	0.35	4.00	23
BMUL	3	145 (0.00)	1.544	2	77
CCHECK	16.393	7.20 (0.88)	0.14	0.50	3
CSEL	0.1	4.00 (0.00)	6.00	0.04	2
DCOMP	16.393	7.20 (0.00)	0.14	0.50	3
DTW	50	167.93 (48.50)	26.94	0.003	72
DWT	3	4 (0.00)	0.02	4	2
EMDH	0.03	10.47 (0.00)	0.00	0.04	9
FFT	15.7	141.97 (85.58)	9.02	4.00	22
GATE	5	67.00 (34.37)	0.63	0.00	17
HCOMP	2.88	77.00 (0.00)	0.65	4.00	4
HCONV	3	89.89 (0.00)	0.80	1.50	8
HFREQ	2.88	61.98 (0.00)	0.52	4.00	6
INV	41	0.267 (0.00)	11.875	30	167
LIC	22.5	63 (6.00)	3.26	-	55
LZ	129	150 (95.00)	30.43	-	55
MA	92	194 (67.00)	32.76	-	55
NEO	3	12.00 (0.00)	0.03	4.00	5
NGRAM	0.2	15.69 (9.07)	0.08	1.50	10
NPACK	3	3.53 (0.00)	5.49	0.008	2
RC	90	29 (0.00)	7.95	-	55
SBP	3	12.00 (0.00)	0.03	0.03	6
SC	3.2	95.30 (64.49)	1.64	0.03-4	12
SUB	3	0.08 (0.00)	0.988	2	69
SVM	3	99.00 (53.58)	0.53	1.67	8
THR	16	2.00 (0.00)	0.11	0.06	1
TOK	6	5.57 (0.00)	0.14	0.001	3
UNPACK	3	3.53 (0.00)	5.49	0.008	2
XCOR	85	377.00 (306.88)	44.11	4.00	81

90^{th} percentile head breadth [169]). To estimate power and data rates, we use path-loss models [84] with a path-loss parameter of 3.5 for transmission through the brain, skull, and skin, like prior studies [114, 138]. Our radio can transmit/receive 7 Mbps at 4.12 GHz and consumes 1.721 mW. We evaluate other radios in Section 7.

Non-volatile memory: We use NVMs with 4 KB page sizes and 1 MB block sizes. An NVM operation can read 8 bytes, write a page, or erase a block. We use NVSim to model NVM and set the SLC NAND erase time (1.5 ms), program time (350 us), and voltage (2.7 V) from industrial technical reports [81]. We choose a low power transistor type, and use a temperature of 40 °C. NVSim estimates a leakage power of 0.26 mW, dynamic energies of 918.809 nJ and 1374 nJ per page for reads and writes, respectively. We use these parameters to size our SC buffers to 24 KB.

Thermal and power limits: No brain region can be overheated beyond 1 °C [161]. The corresponding power cap depends on packaging, implantation depth, and, for multiple implants, the spacing among implants. Like prior work [40, 53, 58, 122, 123, 147], we assume SCALO's implants are expected to be deployed as cuboidal strips or cylindrical capsules near the cortical surface, with the electrodes extending 1.5–2 mm into the brain gray matter. At this depth, no implant can dissipate more than 15 mW [53, 58, 122, 123].

Earlier finite-element analyses of heat dissipation through brain tissues have shown that the temperature increase from an implant falls exponentially with distance, due to blood and cerebrospinal fluid flow [33, 58, 122, 123, 148, 150]. At 10 mm from an implant's edge, the temperature rise is ≈5% of the peak, and at 20 mm, the rise is only 2% with negligible thermal coupling between implants.

We use 20 mm as the default spacing in SCALO. Assuming uniform and optimal distribution of implants on a hemispherical brain surface of 86 mm radius [89], up to 60 SCALO implants can be run at 15 mW each, with negligible thermal coupling. Nonetheless, since node placement may vary by deployment, we report SCALO's performance when the nodes can consume only 12, 9, and 6 mW power, i.e., 60%, 40%, and 20% lower limits.

Electrophysiological data: We use publicly available electrophysiological data for our evaluation [49, 73, 160]. For seizure detection and propagation, we use data from the Mayo Clinic [49] of a patient (label "I001_P013") with 76 electrodes implanted in the parietal and occipital lobes. This data was recorded for 4 days at 5 KHz, and is annotated with seizure instances. We upscaled the sampling frequency to 30 KHz, and split the dataset to emulate multiple implants.

We use overlapping 4 ms windows (120 samples) from the electrodes to detect seizures [116]. For propagation, we compare a seizure-positive signal with the signals in the last 100 ms at all other nodes. When hashing, we use an 8-bit hash for a 4 ms signal.

We use three datasets to evaluate spike sorting. We use the Spikeforest dataset [73], with recordings collected from the CA1 region of a rat hippocampus using tetrode electrodes at 30 KHz sampling frequency. The dataset contains spikes from 10 neurons, with 65,000 spikes from 4 channels that were manually sorted. We also use the Kilosort [98] dataset, which has 35,000 spikes from 30 neurons collected with a neuropixel probe with 384 channels [98]. Finally, we use the MEArec dataset [18], which contains 4,544 spikes from 20 neurons, generated using a neuron cell simulation model.

Alternative system architectures: Table 2 shows the systems that we compare SCALO against. *SCALO No-Hash* uses the SCALO architecture but without hashes. The power saved by removing the hash PEs is allocated to the remaining tasks optimally. *Central No-Hash* uses a single processor without hashes like most existing BCIs [3, 57, 121]. The processor is connected to the multiple sensors using wires. *Central* is another single-processor design, but uses hashes like SCALO. Finally, we have *HALO+NVM*, which uses a single HALO processor from prior work [53], augmented with an NVM to support our applications. Since this design does not have our new PEs, it uses the RISC-V processor for tasks like hashing.

We do not consider (1) wired distributed designs because it is impractical to have all-to-all wires on the brain surface, (2) wireless centralized designs as they have lesser compute available than the wired ones, and (3) designs without storage since all our applications need it. We map the applications onto all systems using the ILP, ensuring that each implant consumes < 15 mW.

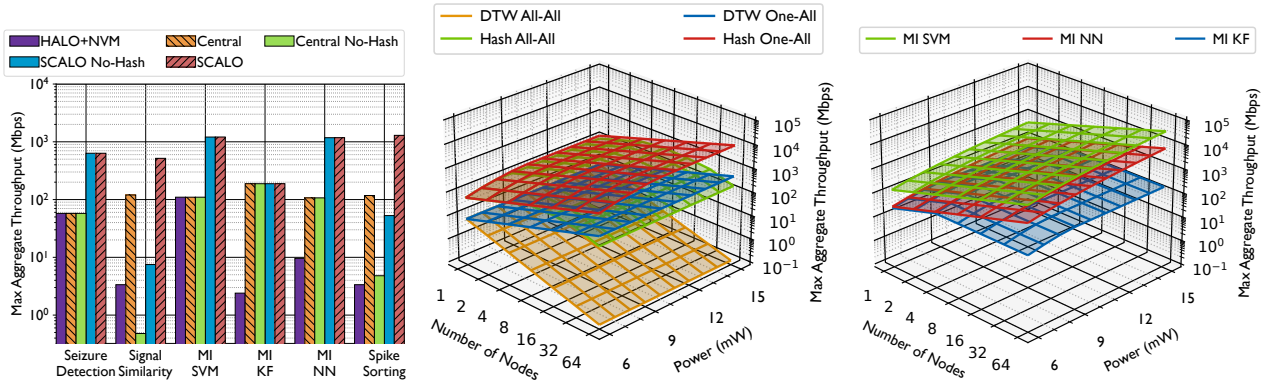
Table 2: Alternative BCI architectures.

Design	Architecture	Comparison	Communication
SCALO (Proposed)	Distributed	Hash, Signal	Wireless
SCALO No-Hash	Distributed	Signal	Wireless
Central No-Hash	Centralized	Signal	Wired
Central	Centralized	Hash, Signal	Wired
HALO+NVM	Centralized	Hash, Signal	Wired

6 EVALUATION

6.1 Comparing BCI Architectures

We compare BCI architectures using their "maximum aggregate throughput" per application. This value is the throughput achieved (over all nodes) for an application when it is the only one running on SCALO. Aggregate throughput is calculated by increasing the number of electrode signals (and ADCs) that the node can process



(a) Maximum aggregate throughput of SCALO and alternative BCI architectures for 11 nodes.

(b) Maximum aggregate throughput of signal similarity methods.

(c) Maximum aggregate throughput of movement intent applications.

Figure 8: Experimental quantification of SCALO's benefits.

until the available power is fully utilized, or response time is violated. We consider a total of 11 implanted sites, which has the highest seizure propagation throughput for SCALO and SCALO No-Hash (Section 6.3). We later vary the number of nodes.

Figure 8a shows performance results. We separate seizure detection and signal similarity in the seizure propagation application, since the former is local while the latter is distributed. Among the centralized designs, *HALO+NVM* does not have SCALO's new PEs but has the same performance as *Central* and *Central No-Hash* for seizure detection and SVM-based movement intent (MI SVM). This is because the PEs in *HALO+NVM* are sufficient for these tasks. On the other hand, *HALO+NVM* is 10–100× worse than *Central* for the remaining tasks because they are run on a slow microcontroller. For the spike sorting application, despite using hashing, *HALO+NVM* has a 40% lower throughput than *Central No-Hash* because checking for hash collisions on the microcontroller is slower than running an exact comparison on a PE in *Central No-Hash*. This performance gap highlights the need for hardware acceleration.

Central No-Hash has 250× and 24.5× lower throughput than *Central* for signal similarity and spike sorting respectively. These tasks benefit from hashes while *Central No-Hash* does not support hashing. The impact of not hashing is much more pronounced for signal similarity because this task involves inter-implant communication. Without hashes, the number of signals that can be communicated and compared under the power limit is low.

Central performs best among uniprocessor designs. However, the processor is the bottleneck for multi-site interfacing, and *Central* has 10× lower throughput than SCALO for all applications. One exception is the movement intent with Kalman filter (MI KF) application. In this case, SCALO also centralizes the computations (Section 3.1), resulting in a similar throughput for the two designs.

With *SCALO No-Hash*, overall processing capability scales with number of implants, as seen by throughput for seizure detection and MI SVM. However, *SCALO No-Hash* does not use hashing and performs worse than *Central* for signal similarity and spike sorting.

Finally, SCALO has the highest throughput for all applications. SCALO's LSH features enables scaling with more implants. Compared to *HALO+NVM*, which is the state-of-the-art prior work, SCALO's processing rates are 10× higher for seizure detection and MI KF, and are up to 385× higher for the remaining applications.

6.2 Performance Scalability

We evaluate the performance (maximum aggregate throughput) of SCALO for our applications with various node counts and per-node power limits. Among the applications, the seizure detection task and spike sorting are fully local to each node. Among these, seizure detection has more complex operations than spike sorting. The throughput of seizure detection at 15 mW is 79 Mbps and falls quadratically to 46 Mbps at 6 mW. Spike sorting has a throughput of 118 Mbps at 15 mW, which decreases linearly to 38.4 Mbps at 6 mW.

For the remaining applications, which are distributed, Figures 8b and 8c show their performance scaling with varying node counts and power limits. Figure 8b shows the performance of hash and exact (DTW) signal similarity methods separately, under two communication patterns each. The first is all-to-all, which is the worst case communication pattern that occurs when brain-wide correlations must be identified, e.g., when there is a seizure at all nodes. The other is one-to-all communication, which occurs when only a single node detects a seizure and must broadcast its data.

DTW All-All has the least throughput because only 16 electrode signals can be transmitted in this mode. The reason is that the intra-SCALO radio can only transmit ≈7 Mbps, while new electrode samples are obtained at 46 Mbps from the ADC. Increasing the number of nodes decreases the throughput further because each node must serially access the TDMA network. Being communication-limited, *DTW All-All* is unaffected by lowering power limits even up to 6 mW. The DTW PE only needs 4 mW to process data at the available radio transmission rate, and its throughput scales linearly with power only below 4 mW.

DTW One-All scales better as its communication cost is fixed. However, a one-to-all comparison is insufficient for general BCI applications. *DTW One-All* is also communication-limited like *DTW All-All* and remains unaffected by lower power up to 4 mW.

The throughput of *Hash All-All* is 10× higher than that of *DTW One-All* for node counts ≤6. Relative to *DTW All-All*, the performance advantage is even higher. *Hash All-All* throughput linearly

increases up to to 547 Mbps (for 6 nodes with 190 electrode signals), after which it begins to decrease. When the number of nodes is small, few TDMA slots are required to exchange all hashes, allowing throughput to linearly increase with node count. When node count increases beyond a limit (i.e. 6), it takes longer to communicate all hashes and overall throughput reduces. *Hash One-All* has a 10× higher throughput than even *Hash All-All*, and exhibits linear scaling since the communication cost is fixed.

Hash processing is not communication-limited, as the transmitted is small (1 B per electrode versus 256 B for DTW). Throughput falls linearly when the power limit is lowered. Keeping number of nodes equal, for *Hash All-All*, peak throughput reduces from 547 Mbps at 15 mW to 135.35 Mbps at 6 mW, while for *Hash One-All*, it reduces from 6,851 Mbps at 15 mW to 1,444 Mbps at 6 mW.

Figure 8c shows the performance of the movement intent applications. These use an all-to-one communication pattern: *MI KF* sends the features from all nodes, while *MI SVM* and *MI NN* send partial classifier outputs to one node. *MI SVM* transmits only 4 B per node even if the number of electrodes per node goes beyond 96, because it only needs to send the partial classifier output and not electrode data. Additionally, for a given power limit, *MI SVM* can process 3% more electrodes than hash generation because the SVM PE consumes 3% lower power than the hash PEs. Therefore, *MI SVM* has the highest throughput than even *Hash One-All*, which also scales linearly with node size.

MI NN, like *MI SVM*, has a fixed data transmission size per node, but the size of this data is higher (1024 B). Therefore, it has a lower throughput than *MI SVM* but has the same scaling trend. Both are power limited and see a linear decrease in throughput with power.

In contrast to the other MI applications, *MI KF* transmits much higher data at 4 B *per electrode*, since it transmits only features for centralized processing. Furthermore, the inversion step in *MI KF* at the receiver has a high usage of the NVM. Therefore, *MI KF*'s throughput scales linearly only up to 4 nodes, where the NVM bandwidth saturates and the application cannot process any more electrodes in the given response time. Therefore, with higher number of nodes, the number of electrodes that can be processed per node decreases, and overall throughput remains the same.

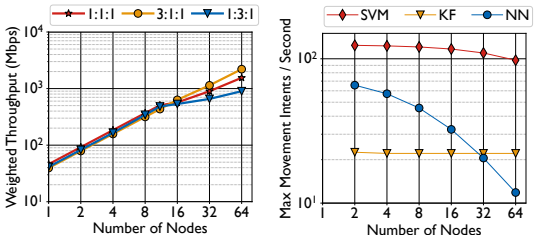
MI KF is limited only by NVM bandwidth above 8.5 mW power, and does not see any throughput reduction until the power limit reaches this value. Below this, throughput falls off quadratically.

6.3 Application Performance

We measure application-level performance via throughput for seizure propagation, number of intents per second for the movement applications, and the spikes sorted, for various node counts.

Seizure propagation has multiple inter-related tasks since seizure detection can run concurrently with hash or DTW comparison, and there is a choice between sending more hashes or signals in the given response time. Hence, it is necessary to specify priorities for these tasks to determine the application performance. Recall that the ILP maximizes the priority-weighted sum of the signals processed in the tasks. Although the ultimate choice of weights is determined by a clinician, we evaluate three sets of weights.

Figure 9a shows the maximum weighted aggregate throughput for seizure propagation with different weight choices (in the format; seizure detection:hash comparison:DTW comparison). With equal priority for all tasks, throughput increases linearly up to 506 Mbps, achieved at 11 nodes. The highest throughput per node is achieved at this node count. Beyond this value, overall throughput increases sublinearly due to communication costs. Other weight choices have different throughput and optimal node counts.



(a) Weighted throughput of seizure propagation tasks. **(b) Movement intents per second.**

Figure 9: Application level metrics on SCALO.

Conventional movement intent (MI) applications use a fixed time interval (e.g., 50 ms) to detect one intent. This limits the number of intents detected (i.e., 20 per second). SCALO decodes movements much faster than this interval.

Figure 9b shows the maximum number of intents detected per second on SCALO. This metric only accounts for intent detection, without the variable response latency of the prosthetic. SCALO significantly outperforms conventional MI SVM and MI NN, which offer only 20 intents per second and for a few electrodes [160] (not shown in the figure). For MI KF, which is the most complex MI application, SCALO also supports 20 intents per second but can process up to a total of 384 electrodes, which is up to 4 nodes for a 96-electrode node. For higher node count, SCALO can still retain its performance but the electrodes processed per node decrease.

Finally, SCALO sorts up to 12,250 spikes per second per node by using hashes to match spikes with preset templates on the NVM. For reference, leading off-device exact matching algorithms sort up to ≈15,000 spikes per second but use multicore CPUs or GPUs [28, 98]. The sorting accuracy of SCALO is within 5% of that achieved by exact template matching, which is 82%, 91%, and 73%, respectively for the SpikeForest [73], MEArec [18], and Kilosort [98] datasets.

6.4 Interactive Queries

We consider three common queries for data ranging from the past 110 ms (≈7 MB over all nodes) to the past 1 s (≈60 MB). They are: **Q1**, which returns all signals detected as a seizure; **Q2**, which returns all signals that match a given template using a hash; and **Q3**, which returns all data in the time range. For Q1 and Q2, we set the fraction of data that tests positive for their condition at 5%, 50%, and 100%.

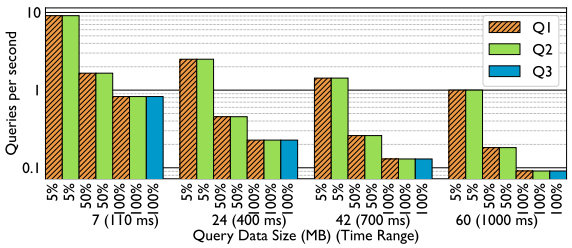


Figure 10: Interactive query throughput with 11 nodes.

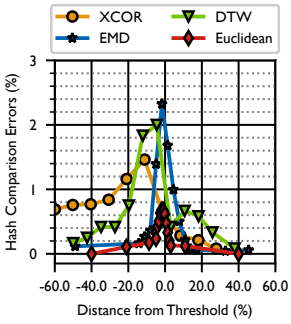


Figure 11: Hash errors.

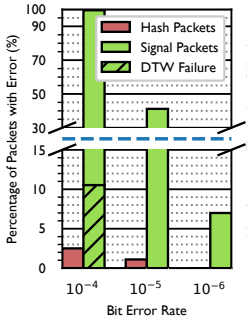


Figure 12: Network errors.

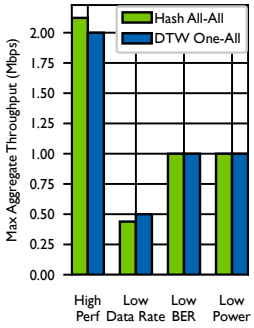


Figure 13: Impact of radio.

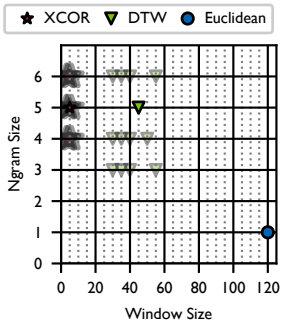


Figure 14: Hash flexibility.

Figure 10 shows SCALO's throughput with 11 nodes for our queries. SCALO supports up to 9 queries per second (QPS) for Q1 and Q2 over the last 110 ms data for 5% matched data, which is the common range of data queried over. If Q2 is run with DTW instead of hashes, the QPS is 8, which is only slightly lower, but the power consumption increases to 15 mW instead of the 3.57 mW consumed with hash-based matching. DTW-based matching is unsuitable when interactively querying in response to a seizure. Q3 takes 1.21 s, yielding a throughput of ≈0.8. The power-hungry external radio becomes the bottleneck for interactive querying. Query latency increases linearly with more search data because of radio latency. Still, SCALO can processes 1 QPS for Q1 and Q2 for the past 1 s data (≈60 MB) with 5% matched data, making it usable in real time.

6.5 Hash Encoding Accuracy

We measure the accuracy of hash-based signal comparison relative to exact comparison for various measures. For each measure, we set a similarity threshold. If the measure for a given pair of signals is above the threshold, they are considered similar. We then configure our hash generation functions for this threshold, and check for the same outcome using hashes, i.e., only similar signals should generate the same hash. Figure 11 shows the percentage of errors between hash and signal comparison as a function of the signals' distance from the threshold. The total errors, represented as area under the curve, are few at <8.5%. Most hash errors occur close to the threshold, where even exact comparison is of low confidence in identifying a match, and errors taper off quickly with distance from the threshold. Note that we bias the hashes towards false positives (left of the threshold), since they can be resolved by an exact comparison.

6.6 Impact of Network Errors

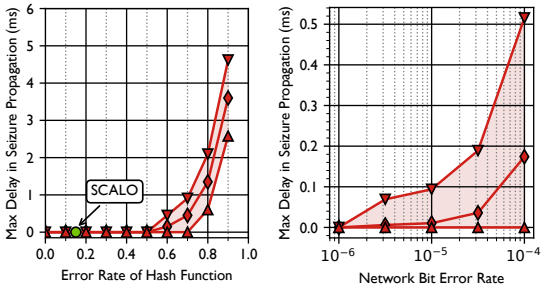
The intra-SCALO network protocol drops packets carrying hashes when there is a checksum error, but allows signal packets to flow into PEs since signal similarity measures are naturally resilient to errors. We simulate bit-error ratios (BERs) with uniformly-random bit flips in packet headers/data. Figure 12 shows the fraction of hash/signal packets with an error at different BERs, and the fraction of erroneous signal packets that flipped the similarity measure (DTW). The BER is $< 10^{-5}$ for the radio we use.

Figure 12 shows that signals and hashes suffer errors as BER increases, but signals are more susceptible since they are longer. However, even though many signal packets suffer errors, they have no impact on the final signal similarity outcome since the measures are naturally resilient. For our design (BER $< 10^{-5}$), <1% of hash packets have errors and there is no DTW failure.

6.7 Error Impact on Applications

Hash errors, caused either due to incorrect encoding or network faults, can affect application performance. However, signals in the brain are spatially and temporally correlated, providing some resiliency to such errors. We use the time-sensitive seizure propagation application to study the impact of hash errors. In this application, a false negative or a hash packet error can cause seizure propagation confirmation to be delayed.

Figure 15 shows the maximum delay in seizure propagation for each type of error when there is a correlated seizure in two brain regions, showing the 100% intervals after 1000 repetitions. Figure 15a shows that hash encoding errors (which in this case are false negatives because there is an ongoing correlated seizure) do not cause any noticeable impact until the error rate is around 50%. The reason for this resiliency is that when a seizure occurs, it is captured by multiple electrodes. It is highly unlikely that all such signals are incorrectly encoded to completely miss the correlation at this time step. For reference, we observe only 12.5% of false negatives in SCALO. Furthermore, a seizure lasts for a few seconds, meaning that another round of correlation checking can occur at the next time step even if it is missed in the current time step.



(a) Encoding errors. **(b) Network errors.**

Figure 15: Maximum delay in detecting seizure propagation due to hash errors, averaged over all seizures. Shaded regions show the full range of observations.

Figure 15b shows the application-level delay with network BER. Recall that all hashes from a node can be sent in one packet. Therefore, a network error results in the loss of the hashes from all

electrodes at a node, and correlation can resume only at the next time step. Consequently, network errors are more harmful than encoding errors. However, these errors are also much less likely to occur (note that the Y axis of Figure 15b is different from Figure 15a), and the worst delay even at a BER of 10^{-4} is 0.5 ms. For reference, the radio we use has a BER of 10^{-5}.

7 DESIGN SPACE EXPLORATION

Hash Parameter selection: Figure 14 shows the best parameters for LSH (window size and n-gram size—Section 2.4) to approximate different signal measures. We also show parameters (with lighter colors in the figure) that are within 90% of the true positive rate achieved by the corresponding best configuration. This flexibility enables reusing the same hash (and PEs) for different measures.

Radio parameters: There are many radio designs for safe implantation with various trade-offs between the data rate, power, and BER [8, 21, 56, 111]. We evaluate the performance of hash (All-All) and DTW (One-All) with four such radios listed in Table 3. For all radios, we maintain a transmission distance of 20 cm and scale the remaining parameters appropriately for this distance [84, 114, 138, 156]. Low Power is our default radio.

Table 3: Alternative radio designs. Our choice is Low Power

Name	BER	Data rate (Mbps)	Power (mW)
Low Power	10^{-5}	7	1.71
High Perf	10^{-6}	14	6.85
Low BER	10^{-6}	7	3.4
Low Data Rate	10^{-5}	3.5	0.855

Figure 13 shows the throughput of the applications with the different radios, normalized with that of our default choice (*Low Power*) . The *High Perf* radio doubles the throughput of both applications because they are communication sensitive. However, the radio power becomes 4×, occupying nearly half the available 15 mW budget. The *Low BER* radio has the same performance as our default, but has 2× the power. This trade-off is not advantageous since our BER is already low (10^{-5}). Lastly, the *Low Data Rate* radio results in a 50% lower performance for the applications, which is unacceptable for our response time targets.

8 RELATED WORK

Single-site BCIs: Commercial and research BCIs have focused largely on single-site monitoring and stimulation [53, 57, 78, 96, 133], and have no support for distributed systems, making them unsuitable for the applications that we target. Most implantable BCIs offer little to no storage and stream data out continuously instead. NeuroChip [121] is an exception but is wired to an external case with a 128 GB SD card that is physically extracted for offline data analysis.

Distributed implants: A growing interest in distributed analyses of the brain [6, 10, 135] has motivated the design of multi-site BCIs [3, 27, 69]. These BCIs, however, lack on-board processing and stream data to a separate computer, or a chest or scalp mounted processing hub. Unfortunately, such centralization restricts the response time and throughput of the BCI, limiting its utility for distributed applications.

Implantation architecture: SCALO presents just one example of a distributed BCI. Alternative designs could include hubs that are chest-implanted [65, 97, 145], or scalp mounted [124, 167]. Hubs can serve as wired sources of power for the implants [97], while the hub itself could be powered by removable or wirelessly charged batteries [65, 97, 167] (it is less risky to wirelessly charge a chest-implanted or externally mounted device). The hub may also act as the sole processor in the system, using the distributed implants only as sensors [65]. Yet another approach is to use wearable hubs [63]. The SCALO architecture can be adapted to suit these various scenarios, although one or more functionalities may not be applicable.

Accelerators for BCI applications: Recent work has designed specialized hardware accelerators for spike sorting using template matching [1], and DNN accelerators for classification using unary networks [162]. These designs are promising, but consume higher power than our target for implantation. We will study integrating them into SCALO in the future.

Mastermind [31][1] is a wearable SoC (System-on-Chip) to support real-time Hierarchical Wasserstein Alignment, an algorithm for movement intent detection. Like Mastermind, SCALO supports movement intent pipelines, albeit with a different set of architectural choices.

9 CONCLUSION

SCALO enables BCI interfacing with multiple brain regions and provides, for the first time, on-device computation for important BCI applications. SCALO offers two orders of magnitude higher task throughput, and provides real-time support for interactive querying with up to 9 QPS over 7 MB data or 1 QPS over 60 MB data. SCALO's design principles—i.e., its modular PE architecture, fast-but-approximate hash-based approach to signal similarity, support for low-power and efficiently-indexed non-volatile storage, and a centralized planner that produces near-optimal mapping of task schedules to devices—can be instrumental to success in other power-constrained environments like IoT (internet of things) as well.

10 ACKNOWLEDGEMENTS

We thank Ján Veselý for advocating for the addition of storage on HALO, which in turn unlocked the ability to build SCALO. We thank Hitten Zaveri, Dennis Spencer, Imran Quraishi, and Nick Turk-Browne for educating us on a variety of neuroscientific and clinical use cases likely to benefit from distributed BCIs. We thank Nick Lindsay, Xiayuan Wen, Ioannis Karageorgos, Ben Cifu, Lenny Khazan, and Gabe Petrov for their feedback on various aspects of the augmented HALO design, which eventually paved the path to SCALO. We thank Sweta Yamini Pothukuchi for calculating the maximal node placement on brain, and for her continued support of our work on this research, despite severe hardship. We thank Janet Kayfetz for her helpful advice on academic writing.

This work was supported in part by the Swebelius Foundation, a gift from NetApp, the NSF's awards 2118851 and 2040682, as well as a Computing Innovation Fellowship from the CRA for Raghavendra Pradyumna Pothukuchi (under NSF grant 2127309). Finally, we wish to acknowledge Dragomir Radev, our much-missed colleague and

[1]Regrettably, we inadvertently omitted a discussion of Mastermind as related work in our original publication. We correct this omission in this version of our manuscript.

friend. We are grateful for his unwavering encouragement of our BCI research over the years, and dedicate this paper to his memory.

Image Credits

Brain image in Figure 2 taken from [149] under license CC BY NC SA, modified by cropping, and removing lines and text.

Prosthetic image in Figure 2 ©[Andrii Symonenko] / Adobe Stock.

REFERENCES

[1] Ameer M. S. Abdelhadi, Eugene Sha, Ciaran Bannon, Hendrik Steenland, and Andreas Moshovos. 2021. Noema: Hardware-Efficient Template Matching for Neural Population Pattern Detection. In *MICRO-54: 54th Annual IEEE/ACM International Symposium on Microarchitecture* (Virtual Event, Greece) *(MICRO '21)*. Association for Computing Machinery, New York, NY, USA, 522–534. https://doi.org/10.1145/3466752.3480121

[2] Naubahar S. Agha, Jacob Komar, Ming Yin, David A. Borton, and Arto Nurmikko. 2013. A fully wireless platform for correlating behavior and neural data from an implanted, neural recording device: Demonstration in a freely moving swine model. In *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, 989–992. https://doi.org/10.1109/NER.2013.6696102

[3] Nur Ahmadi, Matthew L Cavuto, Peilong Feng, Lieuwe B Leene, Michal Maslik, Federico Mazza, Oscar Savolainen, Katarzyna M Szostak, Christos-Savvas Bouganis, Jinendra Ekanayake, et al. 2019. Towards a Distributed, Chronically-Implantable Neural Interface. In *2019 9th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, 719–724. https://doi.org/10.1109/NER.2019.8716998

[4] A Bolu Ajiboye, Francis R Willett, Daniel R Young, William D Memberg, Brian A Murphy, Jonathan P Miller, Benjamin L Walter, Jennifer A Sweet, Harry A Hoyen, Michael W Keith, P Hunter Peckham, John D Simeral, John P Donoghue, Leigh R Hochberg, and Robert F Kirsch. 2017. Restoration of reaching and grasping movements through brain-controlled muscle stimulation in a person with tetraplegia: a proof-of-concept demonstration. *The Lancet* 389, 10081 (May 2017), 1821–1830. https://doi.org/10.1016/s0140-6736(17)30601-3

[5] Elon Musk and. 2019. An Integrated Brain-Machine Interface Platform With Thousands of Channels. *Journal of Medical Internet Research* 21, 10 (Oct. 2019), e16194. https://doi.org/10.2196/16194

[6] Richard A. Andersen, Tyson Aflalo, Luke Bashford, David Bjånes, and Spencer Kellis. 2022. Exploring Cognition with Brain–Machine Interfaces. *Annual Review of Psychology* 73, 1 (Jan. 2022), 131–158. https://doi.org/10.1146/annurev-psych-030221-030214

[7] Joseph N. Y. Aziz, Karim Abdelhalim, Ruslana Shulyzki, Roman Genov, Berj L. Bardakjian, Miron Derchansky, Demitre Serletis, and Peter L. Carlen. 2009. 256-Channel Neural Recording and Delta Compression Microsystem With 3D Electrodes. *IEEE Journal of Solid-State Circuits* 44, 3 (March 2009), 995–1005. https://doi.org/10.1109/jssc.2008.2010997

[8] Hadi Bahrami, S. Abdollah Mirbozorgi, An T. Nguyen, Benoit Gosselin, and Leslie A. Rusch. 2016. System-Level Design of a Full-Duplex Wireless Transceiver for Brain–Machine Interfaces. *IEEE Transactions on Microwave Theory and Techniques* 64, 10 (Oct. 2016), 3332–3341. https://doi.org/10.1109/tmtt.2016.2600301

[9] Tonio Ball, Markus Kern, Isabella Mutschler, Ad Aertsen, and Andreas Schulze-Bonhage. 2009. Signal quality of simultaneously recorded invasive and non-invasive EEG. *NeuroImage* 46, 3 (July 2009), 708–716. https://doi.org/10.1016/j.neuroimage.2009.02.028

[10] Fabrice Bartolomei, Stanislas Lagarde, Fabrice Wendling, Aileen McGonigal, Viktor Jirsa, Maxime Guye, and Christian Bénar. 2017. Defining epileptogenic networks: Contribution of SEEG and signal analysis. *Epilepsia* 58, 7 (2017), 1131–1147. https://doi.org/10.1111/epi.13791

[11] Sliman J. Bensmaia and Lee E. Miller. 2014. Restoring sensorimotor function through intracortical interfaces: progress and looming challenges. *Nature Reviews Neuroscience* 15, 5 (2014), 313–325. https://doi.org/10.1038/nrn3724

[12] Ned Bingham and Rajit Manohar. 2019. QDI Constant-Time Counters. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 27, 1 (Jan. 2019), 83–91. https://doi.org/10.1109/tvlsi.2018.2867289

[13] Alliance Biomedica. 2019. Spencer Probe Depth Electrodes Retrieved August 10, 2019 from http://alliancebiomedica.com/index.php?route=product/product&product_id=164. (Aug. 2019).

[14] Blackrock Microsystems. 2019. The Benchmark for Multichannel, High-density Neural Recording. https://www.blackrockmicro.com/electrode-types/utah-array/. Retrieved August 10, 2019.

[15] Hal Blumenfeld. 2014. What Is a Seizure Network? Long-Range Network Consequences of Focal Seizures. In *Issues in Clinical Epileptology: A View from the Bench*. Springer Netherlands, 63–70. https://doi.org/10.1007/978-94-017-8914-1_5

[16] Réka Barbara Bod, János Rokai, Domokos Meszéna, Richárd Fiáth, István Ulbert, and Gergely Márton. 2022. From End to End: Gaining, Sorting, and Employing High-Density Neural Single Unit Recordings. *Frontiers in Neuroinformatics* 16 (June 2022). https://doi.org/10.3389/fninf.2022.851024

[17] David A Borton, Ming Yin, Juan Aceros, and Arto Nurmikko. 2013. An implantable wireless neural interface for recording cortical circuit dynamics in moving primates. *Journal of Neural Engineering* 10, 2 (Feb. 2013), 026010. https://doi.org/10.1088/1741-2560/10/2/026010

[18] Alessio Paolo Buccino and Gaute Tomas Einevoll. 2020. MEArec: A Fast and Customizable Testbench Simulator for Ground-truth Extracellular Spiking Activity. *Neuroinformatics* 19, 1 (July 2020), 185–204. https://doi.org/10.1007/s12021-020-09467-7

[19] Yingqiu Cao, Nikolai Rakhilin, Philip H Gordon, Xiling Shen, and Edwin C Kan. 2016. A real-time spike classification method based on dynamic time warping for extracellular enteric neural recording with large waveform variability. *Journal of Neuroscience Methods* 261 (2016), 97–109. https://doi.org/10.1016/j.jneumeth.2015.12.006

[20] Jose M Carmena, Mikhail A Lebedev, Roy E Crist, Joseph E Odoherty, David M Santucci, Dragan F Dimitrov, Parag G Patil, Craig S Henriquez, and Miguel A. L Nicolelis. 2003. Learning to Control a Brain–Machine Interface for Reaching and Grasping by Primates. *PLOS Biology* 1, 2 (2003). https://doi.org/10.1371/journal.pbio.0000042

[21] Moo Sung Chae, Zhi Yang, Mehmet R. Yuce, Linh Hoang, and Wentai Liu. 2009. A 128-Channel 6 mW Wireless Neural Recording IC With Spike Feature Extraction and UWB Transmitter. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 17, 4 (Aug. 2009), 312–321. https://doi.org/10.1109/tnsre.2009.2021607

[22] Santosh Chandrasekaran, Matthew Fifer, Stephan Bickel, Luke Osborn, Jose Herrero, Breanne Christie, Junqian Xu, Rory K. J. Murphy, Sandeep Singh, Matthew F. Glasser, Jennifer L. Collinger, Robert Gaunt, Ashesh D. Mehta, Andrew Schwartz, and Chad E. Bouton. 2021. Historical perspectives, challenges, and future directions of implantable brain-computer interfaces for sensorimotor applications. *Bioelectronic Medicine* 7, 1 (Sept. 2021). https://doi.org/10.1186/s42234-021-00076-6

[23] Ujwal Chaudhary, Niels Birbaumer, and Ander Ramos-Murguialday. 2016. Brain–computer interfaces for communication and rehabilitation. *Nature Reviews Neurology* 12, 9 (01 Sep 2016), 513–525. https://doi.org/10.1038/nrneurol.2016.113

[24] Tsan-Jieh Chen, Chi Jeng, Shun-Ting Chang, Herming Chiueh, Sheng-Fu Liang, Yu-Cheng Hsu, and Tzu-Chieh Chien. 2011. A Hardware Implementation of Real-Time Epileptic Seizure Detector on FPGA. In *2011 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE. https://doi.org/10.1109/biocas.2011.6107718

[25] Jaeouk Cho, Geunchang Seong, Yonghee Chang, and Chul Kim. 2021. Energy-Efficient Integrated Circuit Solutions Toward Miniaturized Closed-Loop Neural Interface Systems. *Frontiers in Neuroscience* 15 (May 2021). https://doi.org/10.3389/fnins.2021.667447

[26] Breanne P Christie, Derek M Tat, Zachary T Irwin, Vikash Gilja, Paul Nuyujukian, Justin D Foster, Stephen I Ryu, Krishna V Shenoy, David E Thompson, and Cynthia A Chestek. 2014. Comparison of spike sorting and thresholding of voltage waveforms for intracortical brain–machine interface performance. *Journal of Neural Engineering* 12, 1 (Dec. 2014), 016009. https://doi.org/10.1088/1741-2560/12/1/016009

[27] Jason E Chung, Hannah R Joo, Jiang Lan Fan, Daniel F Liu, Alex H Barnett, Supin Chen, Charlotte Geaghan-Breiner, Mattias P Karlsson, Magnus Karlsson, Kye Y Lee, et al. 2019. High-Density, Long-Lasting, and Multi-region Electrophysiological Recordings Using Polymer Electrode Arrays. *Neuron* 101, 1 (2019), 21–31. https://doi.org/10.1016/j.neuron.2018.11.002

[28] Jason E Chung, Jeremy F Magland, Alex H Barnett, Vanessa M Tolosa, Angela C Tooker, Kye Y Lee, Kedar G Shah, Sarah H Felix, Loren M Frank, and Leslie F Greengard. 2017. A Fully Automated Approach to Spike Sorting. *Neuron* 95, 6 (2017), 1381–1394. https://doi.org/10.1016/j.neuron.2017.08.030

[29] Jennifer Couzin-Frankel. 2021. As Epilepsy Drugs Fail Nearly One-Third of Patients, Scientists Seek Root Causes of Seizures. https://doi.org/10.1126/science.aba5182

[30] Radu Darie, Marc Powell, and David Borton. 2017. Delivering the Sense of Touch to the Human Brain. *Neuron* 93, 4 (2017), 728–730. https://doi.org/10.1016/j.neuron.2017.02.008

[31] Guy Eichler, Luca Piccolboni, Davide Giri, and Luca P. Carloni. 2021. MasterMind: Many-Accelerator SoC Architecture for Real-Time Brain-Computer Interfaces. In *2021 IEEE 39th International Conference on Computer Design (ICCD)*. 101–108. https://doi.org/10.1109/ICCD53106.2021.00027

[32] P. Elias. 1975. Universal Codeword Sets and Representations of the Integers. *IEEE Transactions on Information Theory* 21, 2 (March 1975), 194–203. https://doi.org/10.1109/tit.1975.1055349

[33] Maged M. Elwassif, Qingjun Kong, Maribel Vazquez, and Marom Bikson. 2006. Bio-Heat Transfer Model of Deep Brain Stimulation Induced Temperature changes. In *2006 International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE. https://doi.org/10.1109/iembs.2006.259425

[34] Nir Even-Chen, Dante G. Muratore, Sergey D. Stavisky, Leigh R. Hochberg, Jaimie M. Henderson, Boris Murmann, and Krishna V. Shenoy. 2020. Power-Saving Design Opportunities for Wireless Intracortical Brain–Computer Interfaces. *Nature Biomedical Engineering* (2020). https://doi.org/10.1038/s41551-020-0595-9

[35] D. Garrett, D.A. Peterson, C.W. Anderson, and M.H. Thaut. 2003. Comparison of linear, nonlinear, and feature selection methods for EEG signal classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 11, 2 (June 2003), 141–144. https://doi.org/10.1109/tnsre.2003.814441

[36] Vikash Gilja, Chethan Pandarinath, Christine H Blabe, Paul Nuyujukian, John D Simeral, Anish A Sarma, Brittany L Sorice, János A Perge, Beata Jarosiewicz, Leigh R Hochberg, Krishna V Shenoy, and Jaimie M Henderson. 2015. Clinical translation of a high-performance neural prosthesis. *Nature Medicine* 21, 10 (Sept. 2015), 1142–1145. https://doi.org/10.1038/nm.3953

[37] Lewis Girod, Yuan Mei, Ryan Newton, Stanislav Rost, Arvind Thiagarajan, Hari Balakrishnan, and Samuel Madden. 2008. XStream: a Signal-Oriented Data Stream Management System. In *2008 IEEE 24th International Conference on Data Engineering*. 1180–1189. https://doi.org/10.1109/ICDE.2008.4497527

[38] Graham Gobieski, Ahmet Oguz Atli, Kenneth Mai, Brandon Lucia, and Nathan Beckmann. 2021. Snafu: An Ultra-Low-Power, Energy-Minimal CGRA-Generation Framework and Architecture. In *2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA)*. IEEE. https://doi.org/10.1109/isca52012.2021.00084

[39] Graham Gobieski, Souradip Ghosh, Marijn Heule, Todd Mowry, Tony Nowatzki, Nathan Beckmann, and Brandon Lucia. 2022. RipTide: A Programmable, Energy-Minimal Dataflow Compiler and Architecture. In *2022 55th IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE. https://doi.org/10.1109/micro56248.2022.00046

[40] Jamie J. Van Gompel, S. Matthew Stead, Caterina Giannini, Fredric B. Meyer, W. Richard Marsh, Todd Fountain, Elson So, Aaron Cohen-Gadol, Kendall H. Lee, and Gregory A. Worrell. 2008. Phase I trial: safety and feasibility of intracranial electroencephalography using hybrid subdural electrodes containing macro- and microelectrode arrays. *Neurosurgical Focus* 25, 3 (Sept. 2008), E23. https://doi.org/10.3171/foc/2008/25/9/e23

[41] David Gorisse, Matthieu Cord, and Frederic Precioso. 2011. Locality-Sensitive Hashing for Chi2 Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 2 (2011), 402–409. https://doi.org/10.1109/TPAMI.2011.193

[42] Lukas Grossberger, Francesco P. Battaglia, and Martin Vinck. 2018. Unsupervised clustering of temporal patterns in high-dimensional neuronal ensembles using a novel dissimilarity measure. *PLOS Computational Biology* 14, 7 (July 2018), e1006283. https://doi.org/10.1371/journal.pcbi.1006283

[43] Jason J Han. 2021. Synchron receives FDA approval to begin early feasibility study of their endovascular, brain-computer interface device. *Artificial Organs* 45, 10 (2021), 1134–1135. https://doi.org/10.1111/aor.14049

[44] Reid R. Harrison, Ryan J. Kier, Bradley Greger, Florian Solzbacher, Cynthia A. Chestek, Vikash Gilja, Paul Nuyujukian, Stephen I. Ryu, and Krishna V. Shenoy. 2008. Wireless neural signal acquisition with single low-power integrated circuit. In *2008 IEEE International Symposium on Circuits and Systems*. IEEE. https://doi.org/10.1109/iscas.2008.4541776

[45] Adam O. Hebb, Jun Jason Zhang, Mohammad H. Mahoor, Christos Tsiokos, Charles Matlack, Howard Jay Chizeck, and Nader Pouratian. 2014. Creating the Feedback Loop: Closed Loop Neurostimulation. *Neurosurgery Clinics of North America* 25, 1 (2014), 187–204. https://doi.org/10.1016/j.nec.2013.08.006

[46] Christian Herff, Dean J Krusienski, and Pieter Kubben. 2020. The Potential of Stereotactic-EEG for Brain-Computer Interfaces: Current Progress and Future Directions. *Frontiers in Neuroscience* 14 (2020), 123. https://doi.org/10.3389/fnins.2020.00123

[47] Xiaoxiao Hou, Craig Galligan, Jeffrey Ashe, David A. Borton, and Marc Powell. [n. d.]. Toward multi-area distributed network of implanted neural interrogators. In *Biosensing and Nanomedicine X* (San Diego, United States, 2017-08-29), Hooman Mohseni, Massoud H. Agahi, and Manijeh Razeghi (Eds.). SPIE, 18. https://doi.org/10.1117/12.2276046

[48] L. Huang and G. van. 2013. Brain Computer Interface for Epilepsy Treatment. In *Brain-Computer Interface Systems - Recent Progress and Future Prospects*. InTech. https://doi.org/10.5772/55800

[49] iee.org. 2023. ieeg.org Retrievied April 20, 2023 from http://ieeg.org. (April 2023).

[50] Piotr Indyk and Rajeev Motwani. 1998. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing - STOC '98*. ACM Press. https://doi.org/10.1145/276698.276876

[51] Beata Jarosiewicz, Anish A. Sarma, Daniel Bacher, Nicolas Y. Masse, John D. Simeral, Brittany Sorice, Erin M. Oakley, Christine Blabe, Chethan Pandarinath, Vikash Gilja, Sydney S. Cash, Emad N. Eskandar, Gerhard Friehs, Jaimie M. Henderson, Krishna V. Shenoy, John P. Donoghue, and Leigh R. Hochberg. 2015. Virtual Typing by People with Tetraplegia Using a Self-Calibrating Intracortical Brain-Computer Interface. *Science Translational Medicine* 7, 313 (Nov. 2015). https://doi.org/10.1126/scitranslmed.aac7328

[52] Viktor K Jirsa, Timothée Proix, Dionysios Perdikis, Michael Marmaduke Woodman, Huifang Wang, Jorge Gonzalez-Martinez, Christophe Bernard, Christian Bénar, Maxime Guye, Patrick Chauvel, et al. 2017. The Virtual Epileptic Patient: Individualized whole-brain models of epilepsy spread. *Neuroimage* 145 (2017), 377–388. https://doi.org/10.1016/j.neuroimage.2016.04.049

[53] Ioannis Karageorgos, Karthik Sriram, Ján Veselỳ, Michael Wu, Marc Powell, David Borton, Rajit Manohar, and Abhishek Bhattacharjee. 2020. Hardware-software co-design for brain-computer interfaces. In *2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA)*. IEEE, 391–404. https://doi.org/10.1109/ISCA45697.2020.00041

[54] Ioannis Karageorgos, Karthik Sriram, Ján Veselý, Nick Lindsay, Xiayuan Wen, Michael Wu, Marc Powell, David Borton, Rajit Manohar, and Abhishek Bhattacharjee. 2021. Balancing Specialized Versus Flexible Computation in Brain–Computer Interfaces. *IEEE Micro* 41, 3 (2021), 87–94. https://doi.org/10.1109/MM.2021.3065455

[55] David Karger, Eric Lehman, Tom Leighton, Rina Panigrahy, Matthew Levine, and Daniel Lewin. 1997. Consistent Hashing and Random Trees: Distributed Caching Protocols for Relieving Hot Spots on the World Wide Web. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing - STOC '97*. ACM Press. https://doi.org/10.1145/258533.258660

[56] Hossein Kassiri, Arezu Bagheri, Nima Soltani, Karim Abdelhalim, Hamed Mazhab Jafari, M. Tariqus Salam, Jose Luis Perez Velazquez, and Roman Genov. 2014. Inductively-powered direct-coupled 64-channel chopper-stabilized epilepsy-responsive neurostimulator with digital offset cancellation and tri-band radio. In *ESSCIRC 2014 - 40th European Solid State Circuits Conference (ESSCIRC)*. IEEE. https://doi.org/10.1109/esscirc.2014.6942030

[57] Hossein Kassiri, Sana Tonekaboni, M. Tariqus Salam, Nima Soltani, Karim Abdelhalim, Jose Luis Perez Velazquez, and Roman Genov. [n. d.]. Closed-Loop Neurostimulators: A Survey and A Seizure-Predicting Design Example for Intractable Epilepsy Treatment. 11, 5 ([n. d.]), 1026–1040. https://doi.org/10.1109/TBCAS.2017.2694638

[58] Sohee Minsun Kim, Prashant Tathireddy, R. Normann, and Florian Solzbacher. 2007. Thermal Impact of an Active 3-D Microelectrode Array Implanted in the Brain. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 15 (Dec 2007), 493–501. https://doi.org/10.1109/TNSRE.2007.908429

[59] Yongwook Bryce Kim. 2017. *Physiological Time Series Retrieval and Prediction with Locality-Sensitive Hashing*. Ph. D. Dissertation. Massachusetts Institute of Technology. http://hdl.handle.net/1721.1/111903

[60] Wing kin Tam, Tong Wu, Qi Zhao, Edward Keefer, and Zhi Yang. 2019. Human motor decoding from neural signals: a review. *BMC Biomedical Engineering* 1, 1 (Sept. 2019). https://doi.org/10.1186/s42490-019-0022-z

[61] Asimina Kiourti, Cedric W. L. Lee, Junseok Chae, and John L. Volakis. 2016. A Wireless Fully Passive Neural Recording Device for Unobtrusive Neuropotential Monitoring. *IEEE Transactions on Biomedical Engineering* 63, 1 (Jan. 2016), 131–137. https://doi.org/10.1109/tbme.2015.2458583

[62] Mark A. Kramer and Sydney S. Cash. 2012. Epilepsy as a Disorder of Cortical Network Organization. *The Neuroscientist* 18, 4 (Jan. 2012), 360–372. https://doi.org/10.1177/1073858411422754

[63] Vaclav Kremen, Benjamin H. Brinkmann, Inyong Kim, Hari Guragain, Mona Nasseri, Abigail L. Magee, Tal Pal Attia, Petr Nejedly, Vladimir Sladky, Nathanial Nelson, Su-Youne Chang, Jeffrey A. Herron, Tom Adamski, Steven Baldassano, Jan Cimbalnik, Vince Vasoli, Elizabeth Fehrmann, Tom Chouinard, Edward E. Patterson, Brian Litt, Matt Stead, Jamie Van Gompel, Beverly K. Sturges, Hang Joon Jo, Chelsea M. Crowe, Timothy Denison, and Gregory A. Worrell. 2018. Integrating Brain Implants With Local and Distributed Computing Devices: A Next Generation Epilepsy Management System. *IEEE Journal of Translational Engineering in Health and Medicine* 6 (2018), 1–12. https://doi.org/10.1109/JTEHM.2018.2869398

[64] JB Kruskall and M Liberman. 1983. The symmetric time warping algorithm: From continuous to discrete. *Time Warps, String Edits and Macromolecules: The Theory and Practice of Sequence Comparison* (1983).

[65] Levin Kuhlmann, Klaus Lehnertz, Mark P. Richardson, Björn Schelter, and Hitten P. Zaveri. 2018. Seizure prediction — ready for a new era. *Nature Reviews Neurology* 14, 10 (Aug. 2018), 618–630. https://doi.org/10.1038/s41582-018-0055-2

[66] Mustafa Aykut Kural, Jin Jing, Franz Fürbass, Hannes Perko, Erisela Qerama, Birger Johnsen, Steffen Fuchs, M Brandon Westover, and Sándor Beniczky. 2022. Accurate identification of EEG recordings with interictal epileptiform discharges using a hybrid approach: Artificial intelligence supervised by human experts. *Epilepsia* 63, 5 (2022), 1064–1073. https://doi.org/10.1111/epi.17206

[67] Farah Laiwalla, Jihun Lee, Ah-Hyoung Lee, Ethan Mok, Vincent Leung, Steven Shellhammer, Yoon-Kyu Song, Lawrence Larson, and Arto Nurmikko. 2019. A Distributed Wireless Network of Implantable Sub-mm Cortical Microstimulators for Brain-Computer Interfaces. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 6876–6879. https://doi.org/10.1109/EMBC.2019.8857217

[68] Mikhail A. Lebedev and Miguel A. L. Nicolelis. 2017. Brain-Machine Interfaces: From Basic Science to Neuroprostheses and Neurorehabilitation. *Physiological*

Reviews 97, 2 (April 2017), 767–837. https://doi.org/10.1152/physrev.00027.2016

[69] Jihun Lee, Vincent Leung, Ah-Hyoung Lee, Jiannan Huang, Peter Asbeck, Patrick P Mercier, Stephen Shellhammer, Lawrence Larson, Farah Laiwalla, and Arto Nurmikko. 2021. Neural recording and stimulation using wireless networks of microimplants. *Nature Electronics* 4, 8 (2021), 604–614. https://doi.org/10.1038/s41928-021-00631-8

[70] A.C. Linke, L.E. Mash, C.H. Fong, M.K. Kinnear, J.S. Kohli, M. Wilkinson, R. Tung, R.J. Jao Keehn, R.A. Carper, I. Fishman, and R.-.A. Müller. [n. d.]. Dynamic time warping outperforms Pearson correlation in detecting atypical functional connectivity in autism spectrum disorders. 223 ([n. d.]), 117383. https://doi.org/10.1016/j.neuroimage.2020.117383

[71] F Lotte, L Bougrain, A Cichocki, M Clerc, M Congedo, A Rakotomamonjy, and F Yger. 2018. A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update. *Journal of Neural Engineering* 15, 3 (April 2018), 031005. https://doi.org/10.1088/1741-2552/aab2f2

[72] Chen Luo and Anshumali Shrivastava. 2017. SSH (Sketch, Shingle, & Hash) for Indexing Massive-Scale Time Series. In *Proceedings of the Time Series Workshop at NIPS 2016 (Proceedings of Machine Learning Research, Vol. 55)*, Oren Anava, Azadeh Khaleghi, Marco Cuturi, Vitaly Kuznetsov, and Alexander Rakhlin (Eds.). PMLR, Barcelona, Spain, 38–58. https://proceedings.mlr.press/v55/luo16.html

[73] Jeremy Magland, James J Jun, Elizabeth Lovero, Alexander J Morley, Cole Lincoln Hurwitz, Alessio Paolo Buccino, Samuel Garcia, and Alex H Barnett. 2020. SpikeForest, reproducible web-facing ground-truth validation of automated neural spike sorters. *Elife* 9 (2020), e55167. https://doi.org/10.7554/eLife.55167

[74] Andrew Makhorin. 2008. GLPK (GNU linear programming kit). (2008). http://www.gnu.org/s/glpk/glpk.html

[75] Starting Matlab. 2012. Matlab. *The MathWorks, Natick, MA* (2012).

[76] Dennis J. McFarland, Janis Daly, Chadwick Boulay, and Muhammad A. Parvaz. 2017. Therapeutic applications of BCI technologies. *Brain-Computer Interfaces* 4, 1-2 (April 2017), 37–52. https://doi.org/10.1080/2326263x.2017.1307625

[77] Medtronic. 2008. Medtronic Activa PC Multi-program neurostimulator implant manual. http://www.neuromodulation.ch/sites/default/files/pictures/activa_PC_DBS_implant_manuel.pdf. Retrieved August 10, 2019.

[78] Medtronic. 2018. Deep Brain Stimulation Systems - Activa PC. https://www.medtronic.com/us-en/healthcare-professionals/products/neurological/deep-brain-stimulation-systems/activa-pc.html. Retrieved August 10, 2019.

[79] Edward M. Merricks, Elliot H. Smith, Guy M. McKhann, Robert R. Goodman, Lisa M. Bateman, Ronald G. Emerson, Catherine A. Schevon, and Andrew J. Trevelyan. 2015. Single unit action potentials in humans and the effect of seizure activity. *Brain* 138, 10 (July 2015), 2891–2906. https://doi.org/10.1093/brain/awv208

[80] Guowang Miao, Jens Zander, Ki Won Sung, and Slimane Ben Slimane. 2016. *Fundamentals of mobile data networks*. Cambridge University Press.

[81] Inc. Micron Technology. [n. d.]. MT29F128G08AKCABH2-10. https://www.micron.com/products/nand-flash/slc-nand/part-catalog/mt29f128g08akcabh2-10. Retrieved November 17, 2022.

[82] D. Mills. 1995. *Simple Network Time Protocol (SNTP)*. Technical Report. https://doi.org/10.17487/rfc1769

[83] Rosaleena Mohanty, William A. Sethares, Veena A. Nair, and Vivek Prabhakaran. 2020. Rethinking Measures of Functional Connectivity via Feature Extraction. *Scientific Reports* 10, 1 (Jan. 2020). https://doi.org/10.1038/s41598-020-57915-w

[84] Andreas F Molisch, Kannan Balakrishnan, Chia-Chin Chong, Shahriar Emami, Andrew Fort, Johan Karedal, Juergen Kunisch, Hans Schantz, Ulrich Schuster, and Kai Siwiak. 2004. IEEE 802.15. 4a channel model-final report. *IEEE P802* 15, 04 (2004), 0662.

[85] Alberto Moreno and Jordi Cortadella. 2017. Synthesis of All-Digital Delay Lines. In *2017 23rd IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC)*. 75–82. https://doi.org/10.1109/ASYNC.2017.10

[86] Christian Mühl, Brendan Allison, Anton Nijholt, and Guillaume Chanel. 2014. A survey of affective brain computer interfaces: principles, state-of-the-art, and challenges. *Brain-Computer Interfaces* 1, 2 (April 2014), 66–84. https://doi.org/10.1080/2326263x.2014.912881

[87] K.-R. Muller, C.W. Anderson, and G.E. Birch. 2003. Linear and nonlinear methods for brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 11, 2 (2003), 165–169. https://doi.org/10.1109/TNSRE.2003.814484

[88] Maxwell D. Murphy, David J. Guggenmos, David T. Bundy, and Randolph J. Nudo. 2016. Current Challenges Facing the Translation of Brain Computer Interfaces from Preclinical Trials to Use in Human Patients. *Frontiers in Cellular Neuroscience* 9 (Jan. 2016). https://doi.org/10.3389/fncel.2015.00497

[89] D. A. Nelson and S. A. Nunneley. 1998. Brain temperature and limits on transcranial cooling in humans: quantitative modeling results. *European Journal of Applied Physiology* 78, 4 (Aug. 1998), 353–359. https://doi.org/10.1007/s004210050431

[90] Adam R Neumann, Robrecht Raedt, Hendrik W Steenland, Mathieu Sprengers, Katarzyna Bzymek, Zaneta Navratilova, Lilia Mesina, Jeanne Xie, Valerie Lapointe, Fabian Kloosterman, Kristl Vonck, Paul A J M Boon, Ivan Soltesz, Bruce L McNaughton, and Artur Luczak. 2017. Involvement of fast-spiking cells in ictal sequences during spontaneous seizures in rats with chronic temporal lobe epilepsy. *Brain* 140, 9 (Aug. 2017), 2355–2369. https://doi.org/10.1093/brain/awx179

[91] Milos Nikolic, Badrish Chandramouli, and Jonathan Goldstein. 2017. Enabling Signal Processing over Data Streams *(SIGMOD '17)*. Association for Computing Machinery, New York, NY, USA, 95–108. https://doi.org/10.1145/3035918.3035935

[92] Tony Nowatzki, Newsha Ardalani, Karthikeyan Sankaralingam, and Jian Weng. 2018. Hybrid optimization/heuristic instruction scheduling for programmable accelerator codesign. In *Proceedings of the 27th International Conference on Parallel Architectures and Compilation Techniques*. ACM. https://doi.org/10.1145/3243176.3243212

[93] P. Nuyujukian, J. C. Kao, S. I. Ryu, and K. V. Shenoy. 2017. A Nonhuman Primate Brain–Computer Typing Interface. *Proc. IEEE* 105, 1 (Jan 2017), 66–72. https://doi.org/10.1109/JPROC.2016.2586967

[94] Paul Nuyujukian, Jose Albites Sanabria, Jad Saab, Chethan Pandarinath, Beata Jarosiewicz, Christine H. Blabe, Brian Franco, Stephen T. Mernoff, Emad N. Eskandar, John D. Simeral, Leigh R. Hochberg, Krishna V. Shenoy, and Jaimie M. Henderson. 2018. Cortical Control of a Tablet Computer by People with Paralysis. *PLoS ONE* 13, 11 (2018). https://doi.org/10.1371/journal.pone.0204566

[95] Catherine L Ojakangas, Ammar Shaikhouni, Gerhard M Friehs, Abraham H Caplan, Mijail D Serruya, Maryam Saleh, Daniel S Morris, and John P Donoghue. 2006. Decoding Movement Intent From Human Premotor Cortex Neurons for Neural Prosthetic Applications. *Journal of Clinical Neurophysiology: Official Publication of the American Electroencephalographic Society* 23, 6 (2006), 577. https://doi.org/10.1097/01.wnp.0000233323.87127.14

[96] Gerard O'Leary, David M. Groppe, Taufik A. Valiante, Naveen Verma, and Roman Genov. 2018. NURIP: Neural Interface Processor for Brain-State Classification and Programmable-Waveform Neurostimulation. *IEEE Journal of Solid-State Circuits* 53, 11 (Nov. 2018), 3150–3162. https://doi.org/10.1109/jssc.2018.2869579

[97] Thomas J Oxley, Peter E Yoo, Gil S Rind, Stephen M Ronayne, C M Sarah Lee, Christin Bird, Victoria Hampshire, Rahul P Sharma, Andrew Morokoff, Daryl L Williams, Christopher MacIsaac, Mark E Howard, Lou Irving, Ivan Vrljic, Cameron Williams, Sam E John, Frank Weissenborn, Madeleine Dazenko, Anna H Balabanski, David Friedenberg, Anthony N Burkitt, Yan T Wong, Katharine J Drummond, Patricia Desmond, Douglas Weber, Timothy Denison, Leigh R Hochberg, Susan Mathers, Terence J O'Brien, Clive N May, J Mocco, David B Grayden, Bruce C V Campbell, Peter Mitchell, and Nicholas L Opie. 2020. Motor neuroprosthesis implanted with neurointerventional surgery improves capacity for activities of daily living tasks in severe paralysis: first in-human experience. *Journal of NeuroInterventional Surgery* 13, 2 (Oct. 2020), 102–108. https://doi.org/10.1136/neurintsurg-2020-016862

[98] Marius Pachitariu, Shashwat Sridhar, and Carsen Stringer. 2023. Solving the spike sorting problem with Kilosort. (Jan. 2023). https://doi.org/10.1101/2023.01.07.523036

[99] Miguel Pais-Vieira, Amol P. Yadav, Derek Moreira, David Guggenmos, Amílcar Santos, Mikhail Lebedev, and Miguel A. L. Nicolelis. 2016. A Closed Loop Brain-machine Interface for Epilepsy Control Using Dorsal Column Electrical Stimulation. *Scientific Reports* 6, 1 (Sept. 2016). https://doi.org/10.1038/srep32814

[100] Chethan Pandarinath, Paul Nuyujukian, Christine H Blabe, Brittany L Sorice, Jad Saab, Francis R Willett, Leigh R Hochberg, Krishna V Shenoy, and Jaimie M Henderson. 2017. High Performance Communication by People with Paralysis Using an Intracortical Brain-Computer Interface. *eLife* 6 (Feb. 2017). https://doi.org/10.7554/elife.18554

[101] Josef Parvizi and Sabine Kastner. 2018. Promises and limitations of human intracranial electroencephalography. *Nature Neuroscience* 21, 4 (March 2018), 474–483. https://doi.org/10.1038/s41593-018-0108-2

[102] Ofir Pele and Michael Werman. 2009. Fast and robust Earth Mover's Distances. In *2009 IEEE 12th international conference on computer vision*. IEEE, 460–467. https://doi.org/10.1109/ICCV.2009.5459199

[103] W. W. Peterson and D. T. Brown. 1961. Cyclic Codes for Error Detection. *Proceedings of the IRE* 49, 1 (1961), 228–235. https://doi.org/10.1109/JRPROC.1961.287814

[104] Timothée Proix, Fabrice Bartolomei, Maxime Guye, and Viktor K. Jirsa. 2017. Individual brain structure and modelling predict seizure propagation. *Brain* 140, 3 (Feb. 2017), 641–654. https://doi.org/10.1093/brain/awx004

[105] Ida Mengyi Pu. 2006. *Fundamental Data Compression*. Elsevier. https://doi.org/10.1016/b978-0-7506-6310-6.x5000-4

[106] Enrique S. Quintana, Gregorio Quintana, Xiaobai Sun, and Robert van de Geijn. 2001. A Note On Parallel Matrix Inversion. *SIAM Journal on Scientific Computing* 22, 5 (Jan. 2001), 1762–1771. https://doi.org/10.1137/s1064827598345679

[107] R. Quian Quiroga, L. Reddy, C. Koch, and I. Fried. 2007. Decoding Visual Inputs From Multiple Neurons in the Human Temporal Lobe. *Journal of Neurophysiology* 98, 4 (Oct. 2007), 1997–2007. https://doi.org/10.1152/jn.00125.2007

[108] Hamed Rahmani and Aydin Babakhani. 2021. A Wirelessly Powered Reconfigurable FDD Radio With On-Chip Antennas for Multi-Site Neural Interfaces. *IEEE Journal of Solid-State Circuits* 56, 10 (2021), 3177–3190. https:

//doi.org/10.1109/JSSC.2021.3076014

[109] Adrien B Rapeaux and Timothy G Constandinou. 2021. Implantable brain machine interfaces: first-in-human studies, technology challenges and trends. *Current Opinion in Biotechnology* 72 (Dec. 2021), 102–111. https://doi.org/10.1016/j.copbio.2021.10.001

[110] Hernan Gonzalo Rey, Carlos Pedreira, and Rodrigo Quian Quiroga. 2015. Past, present and future of spike sorting techniques. *Brain research bulletin* 119 (2015), 106–117. https://doi.org/10.1016/j.brainresbull.2015.04.007

[111] Sakib Reza and Ifana Mahbub. 2022. A Power Budget Analysis for an Implantable UWB Transceiver for Brain Neuromodulation Application. In *2022 IEEE USNC-URSI Radio Science Meeting (Joint with AP-S Symposium)*. IEEE. https://doi.org/10.23919/usnc-ursi52669.2022.9887429

[112] Ueli Rutishauser, Erin M Schuman, and Adam N Mamelak. 2006. Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo. *Journal of Neuroscience Methods* 154, 1-2 (2006), 204–224. https://doi.org/10.1016/j.jneumeth.2005.12.033

[113] H. Sakoe and S. Chiba. 1978. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 26, 1 (Feb. 1978), 43–49. https://doi.org/10.1109/tassp.1978.1163055

[114] Mariella Särestöniemi, Carlos Pomalaza-Raez, Kamran Sayrafian, Teemu Myllylä, and Jari Iinatti. 2022. A Preliminary Study of RF Propagation for High Data Rate Brain Telemetry. In *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Springer International Publishing, 126–138. https://doi.org/10.1007/978-3-030-95593-9_11

[115] Claudia Serrano-Amenos, Frank Hu, Po T. Wang, Spencer Kellis, Richard A. Andersen, Charles Y. Liu, Payam Heydari, An H. Do, and Zoran Nenadic. 2020. Thermal Analysis of a Skull Implant in Brain-Computer Interfaces. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE. https://doi.org/10.1109/embc44109.2020.9175483

[116] Preya Shah, Arian Ashourvan, Fadi Mikhail, Adam Pines, Lohith Kini, Kelly Oechsel, Sandhitsu R Das, Joel M Stein, Russell T Shinohara, Danielle S Bassett, et al. 2019. Characterizing the role of the structural connectome in seizure dynamics. *Brain* 142, 7 (2019), 1955–1972. https://doi.org/10.1093/brain/awz125

[117] Aqsa Shakeel, Muhammad Samran Navid, Muhammad Nabeel Anwar, Suleman Mazhar, Mads Jochumsen, and Imran Khan Niazi. 2015. A Review of Techniques for Detection of Movement Intention Using Movement-Related Cortical Potentials. *Computational and Mathematical Methods in Medicine* 2015 (2015). https://doi.org/10.1155/2015/346217

[118] Junhua Shen, Akira Shikata, Lalinda D. Fernando, Ned Guthrie, Baozhen Chen, Mark Maddox, Nikhil Mascarenhas, Ron Kapusta, and Michael C. W. Coln. 2018. A 16-bit 16-MS/s SAR ADC With On-Chip Calibration in 55-nm CMOS. *IEEE Journal of Solid-State Circuits* 53, 4 (2018), 1149–1160. https://doi.org/10.1109/JSSC.2017.2784761

[119] H. Shiao, V. Cherkassky, J. Lee, B. Veber, E. E. Patterson, B. H. Brinkmann, and G. A. Worrell. 2017. SVM-Based System for Prediction of Epileptic Seizures From iEEG Signal. *IEEE Transactions on Biomedical Engineering* 64, 5 (May 2017), 1011–1022. https://doi.org/10.1109/TBME.2016.2586475

[120] Jerry J. Shih, Dean J. Krusienski, and Jonathan R. Wolpaw. 2012. Brain-Computer Interfaces in Medicine. *Mayo Clinic Proceedings* 87, 3 (March 2012), 268–279. https://doi.org/10.1016/j.mayocp.2011.12.008

[121] Larry E Shupe, Frank P Miles, Geoff Jones, Richy Yun, Jonathan Mishler, Irene Rembado, R Logan Murphy, Steve I Perlmutter, and Eberhard E Fetz. 2021. Neurochip3: An Autonomous Multichannel Bidirectional Brain-Computer Interface for Closed-Loop Activity-Dependent Stimulation. *Frontiers in Neuroscience* 15 (2021). https://doi.org/10.3389/fnins.2021.718465

[122] Kanber Mithat Silay, Catherine Dehollain, and Michel Declercq. 2008. Numerical analysis of temperature elevation in the head due to power dissipation in a cortical implant. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE. https://doi.org/10.1109/iembs.2008.4649312

[123] Kanber Mithat Silay, Catherine Dehollain, and Michel Declercq. 2011. Numerical Thermal Analysis of a Wireless Cortical Implant with Two-Body Packaging. *BioNanoScience* 1, 3 (June 2011), 78–88. https://doi.org/10.1007/s12668-011-0009-2

[124] John D Simeral, Thomas Hosman, Jad Saab, Sharlene N Flesher, Marco Vilela, Brian Franco, Jessica N Kelemen, David M Brandman, John G Ciancibello, Paymon G Rezaii, et al. 2021. Home Use of a Percutaneous Wireless Intracortical Brain-Computer Interface by Individuals With Tetraplegia. *IEEE Transactions on Biomedical Engineering* 68, 7 (2021), 2313–2325. https://doi.org/10.1109/TBME.2021.3069119

[125] Nicholas D. Skomrock, Michael A. Schwemmer, Jordyn E. Ting, Hemang R. Trivedi, Gaurav Sharma, Marcia A. Bockbrader, and David A. Friedenberg. 2018. A Characterization of Brain-Computer Interface Performance Trade-Offs Using Support Vector Machines and Deep Neural Networks to Decode Movement Intent. *Frontiers in Neuroscience* 12 (Oct. 2018). https://doi.org/10.3389/fnins.2018.00763

[126] Vladimir Sladky, Petr Nejedly, Filip Mivalt, Benjamin H Brinkmann, Inyong Kim, Erik K St. Louis, Nicholas M Gregg, Brian N Lundstrom, Chelsea M Crowe, Tal Pal Attia, et al. 2022. Distributed brain co-processor for tracking spikes, seizures and behaviour during electrical brain stimulation. *Brain Communications* 4, 3 (2022), fcac115. https://doi.org/10.1093/braincomms/fcac115

[127] Elliot H. Smith and Catherine A. Schevon. 2016. Toward a Mechanistic Understanding of Epileptic Networks. *Current Neurology and Neuroscience Reports* 16, 11 (Sept. 2016). https://doi.org/10.1007/s11910-016-0701-2

[128] A. M. Sodagar, K. D. Wise, and K. Najafi. 2009. A Wireless Implantable Microsystem for Multichannel Neural Recording. *IEEE Transactions on Microwave Theory and Techniques* 57, 10 (Oct 2009), 2565–2573. https://doi.org/10.1109/TMTT.2009.2029957

[129] Boris Sotomayor-Gómez, Francesco P Battaglia, and Martin Vinck. 2021. SpikeShip: A method for fast, unsupervised discovery of high-dimensional neural spiking patterns. *bioRxiv* (2021), 2020–06. https://doi.org/10.1101/2020.06.03.131573

[130] Karthik Sriram, Xiayuan Wen, Ioannis Karageorgos, Ján Veselý, Nick Lindsay, Michael Wu, Lenny Khazan, Raghav Pradyumna Pothukuchi, Rajit Manohar, and Abhishek Bhattacharjee. 2023. HALO: A Hardware-Software Co-Designed Processor for Brain-Computer Interfaces. *IEEE Micro* (2023). https://doi.org/10.1109/MM.2023.3258907

[131] Ian Stevenson and Konrad Kording. 2011. How Advances in Neural Recording Affect Data Analysis. *Nature neuroscience* 14 (02 2011), 139–42. https://doi.org/10.1038/nn.2731

[132] Felice T Sun and Martha J Morrell. 2014. Closed-loop Neurostimulation: The Clinical Experience. *Neurotherapeutics* 11, 3 (2014), 553–563. https://doi.org/10.1007/s13311-014-0280-3

[133] Felice T Sun and Martha J Morrell. 2014. The RNS System: responsive cortical stimulation for the treatment of refractory partial epilepsy. *Expert Review of Medical Devices* 11, 6 (Aug. 2014), 563–572. https://doi.org/10.1586/17434440.2014.947274

[134] Steven Swanson, Ken Michelson, Andrew Schwerin, and Mark Oskin. 2003. WaveScalar. In *Proceedings. 36th Annual IEEE/ACM International Symposium on Microarchitecture, 2003. MICRO-36*. IEEE, 291–302. https://doi.org/10.1109/micro.2003.1253203

[135] Katarzyna M. Szostak, Peilong Feng, Federico Mazza, and Timothy G. Constandinou. 2021. Distributed Neural Interfaces: Challenges and Trends in Scaling Implantable Technology. In *Handbook of Neuroengineering*. Springer Singapore, 1–37. https://doi.org/10.1007/978-981-15-2848-4_11-1

[136] Sina Tafazoli, Camden J MacDowell, Zongda Che, Katherine C Letai, Cynthia R Steinhardt, and Timothy J Buschman. 2020. Learning to control the brain through adaptive closed-loop patterned stimulation. *Journal of Neural Engineering* 17, 5 (2020), 056007. https://doi.org/10.1088/1741-2552/abb860

[137] Desney S. Tan and Anton Nijholt (Eds.). 2010. *Brain-Computer Interfaces*. Springer London. https://doi.org/10.1007/978-1-84996-272-8

[138] Attaphongse Taparugssanagorn, Alberto Rabbachin, Matti Hämäläinen, Jani Saloranta, Jari Iinatti, et al. 2008. A Review of Channel Modelling for Wireless Body Area Network in Wireless Medical Communications. *The 11th International Symposium on Wireless Personal Multimedia Communications (WPMC* (2008).

[139] Sonia Todorova, Patrick Sadtler, Aaron Batista, Steven Chase, and Valérie Ventura. 2014. To sort or not to sort: the impact of spike-sorting on neural decoding performance. *Journal of neural engineering* 11, 5 (2014), 056005. https://doi.org/10.1088/1741-2560/11/5/056005

[140] Christopher Torng, Peitian Pan, Yanghui Ou, Cheng Tan, and Christopher Batten. 2021. Ultra-Elastic CGRAs for Irregular Loop Specialization. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE. https://doi.org/10.1109/hpca51647.2021.00042

[141] Eric M. Trautmann, Sergey D. Stavisky, Subhaneil Lahiri, Katherine C. Ames, Matthew T. Kaufman, Daniel J. O'Shea, Saurabh Vyas, Xulu Sun, Stephen I. Ryu, Surya Ganguli, and Krishna V. Shenoy. 2019. Accurate Estimation of Neural Population Dynamics without Spike Sorting. *Neuron* 103, 2 (July 2019), 292–308.e4. https://doi.org/10.1016/j.neuron.2019.05.003

[142] Farhad R. Udwadia, Patrick J. McDonald, Mary B. Connolly, Viorica Hrincu, and Judy Illes. 2020. Youth Weigh In: Views on Advanced Neurotechnology for Drug-Resistant Epilepsy. *Journal of Child Neurology* 36, 2 (Sept. 2020), 128–132. https://doi.org/10.1177/0883073820957810

[143] U.S. Food and Drug Administration. 2019. Implanted Brain-Computer Interface (BCI) Devices for Patients with Paralysis or Amputation - Non-clinical Testing and Clinical Considerations. https://www.fda.gov/regulatory-information/search-fda-guidance-documents/implanted-brain-computer-interface-bci-devices-patients-paralysis-or-amputation-non-clinical-testing. Retrieved August 10, 2019.

[144] U.S. Food and Drug Administration. 2021. FDA authorizes marketing of device to facilitate muscle rehabilitation in stroke patients. https://www.fda.gov/news-events/press-announcements/fda-authorizes-marketing-device-facilitate-muscle-rehabilitation-stroke-patients.

[145] Mariska J. Vansteensel, Elmar G.M. Pels, Martin G. Bleichner, Mariana P. Branco, Timothy Denison, Zachary V. Freudenburg, Peter Gosselaar, Sacha Leinders, Thomas H. Ottens, Max A. Van Den Boom, Peter C. Van Rijen, Erik J. Aarnoutse, and Nick F. Ramsey. 2016. Fully Implanted Brain–Computer Interface in a Locked-In Patient with ALS. *New England Journal of Medicine* 375, 21 (Nov.

2016), 2060–2066. https://doi.org/10.1056/nejmoa1608085
[146] Gabriel W Vattendahl Vidal, Mathew L Rynes, Zachary Kelliher, and Shikha Jain Goodwin. 2016. Review of Brain-Machine Interfaces Used in Neural Prosthetics with New Perspective on Somatosensory Feedback through Method of Signal Breakdown. *Scientifica* 2016 (2016). https://doi.org/10.1155/2016/8956432
[147] Jonathan Viventi, Dae-Hyeong Kim, Leif Vigeland, Eric S Frechette, Justin A Blanco, Yun-Soung Kim, Andrew E Avrin, Vineet R Tiruvadi, Suk-Won Hwang, Ann C Vanleer, Drausin F Wulsin, Kathryn Davis, Casey E Gelber, Larry Palmer, Jan Van der Spiegel, Jian Wu, Jianliang Xiao, Yonggang Huang, Diego Contreras, John A Rogers, and Brian Litt. 2011. Flexible, foldable, actively multiplexed, high-density electrode array for mapping brain activity in vivo. *Nature Neuroscience* 14, 12 (Nov. 2011), 1599–1605. https://doi.org/10.1038/nn.2973
[148] J. Vrba, R. Janca, M. Blaha, P. Jezdik, A. Belohlavkova, P. Krsek, and D. Vrba. 2019. Modeling of Brain Tissue Heating Caused by Direct Cortical Stimulation for Assessing the Risk of Thermal Damage. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 27, 3 (March 2019), 440–449. https://doi.org/10.1109/tnsre.2019.2898253
[149] Jennifer Walinga and Charles Stangor. [n. d.]. *Neuron and the Brain.* Affordable Course Transformation: The Pennsylvania State University.
[150] Huan Wang, Bonnie Wang, Kieran P. Normoyle, Kevin Jackson, Kevin Spitler, Matthew F. Sharrock, Claire M. Miller, Catherine Best, Daniel Llano, and Rose Du. 2014. Brain temperature and its fundamental properties: a review for clinical neuroscientists. *Frontiers in Neuroscience* 8 (Oct. 2014). https://doi.org/10.3389/fnins.2014.00307
[151] Jing Wang, Rongfeng Zhao, Peitong Li, Zhiqiang Fang, Qianqian Li, Yanling Han, Ruyan Zhou, and Yun Zhang. 2022. Clinical Progress and Optimization of Information Processing in Artificial Visual Prostheses. *Sensors* 22, 17 (Aug. 2022), 6544. https://doi.org/10.3390/s22176544
[152] Allen Waziri, Catherine A. Schevon, Joshua Cappell, Ronald G. Emerson, Guy M. McKhann, and Robert R. Goodman. 2009. Initial surgical experience with a dense cortical microarray in epileptic patients undergoing craniotomy for subdural electrode implantation. *Neurosurgery* 64, 3 (March 2009), 540–545. https://doi.org/10.1227/01.neu.0000337575.63861.10
[153] Jian Weng, Sihao Liu, Zhengrong Wang, Vidushi Dadu, and Tony Nowatzki. 2020. A Hybrid Systolic-Dataflow Architecture for Inductive Matrix Algorithms. In *2020 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE. https://doi.org/10.1109/hpca47549.2020.00063
[154] Johan Wessberg and Miguel A. L. Nicolelis. 2004. Optimizing a Linear Algorithm for Real-Time Robotic Control using Chronic Cortical Ensemble Recordings in Monkeys. *Journal of Cognitive Neuroscience* 16, 6 (2004), 1022–1035. https://doi.org/10.1162/0898929041502652
[155] Alik S. Widge, Darin D. Dougherty, and Chet T. Moritz. 2014. Affective brain-computer interfaces as enabling technology for responsive psychiatric stimulation. *Brain-Computer Interfaces* 1, 2 (April 2014), 126–136. https://doi.org/10.1080/2326263x.2014.912885
[156] K. Wiklundh. 2006. Relation between the amplitude probability distribution of an interfering signal and its impact on digital radio receivers. *IEEE Transactions on Electromagnetic Compatibility* 48, 3 (2006), 537–544. https://doi.org/10.1109/TEMC.2006.877782
[157] Francis Willett, Erin Kunz, Chaofei Fan, Donald Avansino, Guy Wilson, Eun Young Choi, Foram Kamdar, Leigh R. Hochberg, Shaul Druckmann, Krishna V. Shenoy, and Jaimie M. Henderson. 2023. A high-performance speech neuroprosthesis. (Jan. 2023). https://doi.org/10.1101/2023.01.21.524489
[158] Francis R. Willett, Donald T. Avansino, Leigh R. Hochberg, Jaimie M. Henderson, and Krishna V. Shenoy. 2021. High-performance brain-to-text communication via handwriting. *Nature* 593, 7858 (May 2021), 249–254. https://doi.org/10.1038/s41586-021-03506-2
[159] Francis R. Willett, Daniel R. Young, Brian A. Murphy, William D. Memberg, Christine H. Blabe, Chethan Pandarinath, Sergey D. Stavisky, Paymon Rezaii, Jad Saab, Benjamin L. Walter. Jennifer A. Sweet, Jonathan P. Miller, Jaimie M. Henderson, Krishna V. Shenoy, John D. Simeral, Beata Jarosiewicz, Leigh R. Hochberg, Robert F. Kirsch, and A. Bolu Ajiboye. 2019. Principled BCI Decoder Design and Parameter Selection Using a Feedback Control Model. *Scientific Reports* 9, 1 (2019), 8881. https://doi.org/10.1038/s41598-019-44166-7
[160] Matthew S. Willsey, Samuel R. Nason-Tomaszewski, Scott R. Ensel, Hisham Temmar, Matthew J. Mender, Joseph T. Costello, Parag G. Patil, and Cynthia A. Chestek. 2022. Real-time brain-machine interface in non-human primates achieves high-velocity prosthetic finger movements using a shallow feedforward neural network decoder. *Nature Communications* 13, 1 (Nov. 2022). https://doi.org/10.1038/s41467-022-34452-w
[161] Patrick D. Wolf. 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* (2008). https://www.ncbi.nlm.nih.gov/books/NBK3932/
[162] Di Wu, Jingjie Li, Zhewen Pan, Younghyun Kim, and Joshua San Miguel. 2022. uBrain: A Unary Brain Computer Interface. In *Proceedings of the 49th Annual International Symposium on Computer Architecture*. Association for Computing Machinery, 468–481. https://doi.org/10.1145/3470496.3527401
[163] W Wu, M. Black, Y. Gao, M. Serruya, A. Shaikhouni, J. Donoghue, and Elie Bienenstock. 2002. Neural Decoding of Cursor Motion Using a Kalman Filter. In *Advances in Neural Information Processing Systems*, S. Becker, S. Thrun, and K. Obermayer (Eds.), Vol. 15. MIT Press. https://proceedings.neurips.cc/paper_files/paper/2002/file/169779d3852b32ce8b1a1724dbf5217d-Paper.pdf
[164] Tao Xue, Shujun Chen, Yutong Bai, Chunlei Han, Anchao Yang, and Jianguo Zhang. 2022. Neuromodulation in drug-resistant epilepsy: A review of current knowledge. *Acta Neurologica Scandinavica* 146, 6 (Sept. 2022), 786–797. https://doi.org/10.1111/ane.13696
[165] Gürkan Yilmaz and Catherine Dehollain. [n. d.]. *Wireless Power Transfer and Data Communication for Neural Implants.* Springer International Publishing. https://doi.org/10.1007/978-3-319-49337-4
[166] Ming Yin, David A Borton, Jacob Komar, Naubahar Agha, Yao Lu, Hao Li, Jean Laurens, Yiran Lang, Qin Li, Christopher Bull, et al. 2014. Wireless Neurosensor for Full-Spectrum Electrophysiology Recordings during Free Behavior. *Neuron* 84, 6 (2014), 1170–1182. https://doi.org/10.1016/j.neuron.2014.11.010
[167] Ming Yin, Hao Li, Christopher Bull, David A. Borton, Juan Aceros, Lawrence Larson, and Arto V. Nurmikko. 2013. An externally head-mounted wireless neural recording device for laboratory animal research and possible human clinical use. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE. https://doi.org/10.1109/embc.2013.6610199
[168] D Young, F Willett, W D Memberg, B Murphy, P Rezaii, B Walter, J Sweet, J Miller, K V Shenoy, L R Hochberg, R F Kirsch, and A B Ajiboye. 2019. Closed-Loop Cortical Control of Virtual Reach and Posture using Cartesian and Joint Velocity Commands. *Journal of Neural Engineering* 16, 2 (Jan 2019), 026011. https://doi.org/10.1088/1741-2552/aaf606
[169] Joseph W Young. 1993. *Head and Face Anthropometry of Adult U.S. Citizens.* Technical Report. Federal Aviation Administration. https://rosap.ntl.bts.gov/view/dot/21363
[170] K.Y. Yun and R.P. Donohue. 1996. Pausible clocking: a first step toward heterogeneous systems. In *Proceedings International Conference on Computer Design. VLSI in Computers and Processors*. 118–123. https://doi.org/10.1109/ICCD.1996.563543
[171] Hamed Zaer, Ashlesha Deshmukh, Dariusz Orlowski, Wei Fan, Pierre-Hugues Prouvot, Andreas Nørgaard Glud, Morten Bjørn Jensen, Esben Schjødt Worm, Slávka Lukacova, Trine Werenberg Mikkelsen, Lise Moberg Fitting, John R. Adler, M. Bret Schneider, Martin Snejbjerg Jensen, Quanhai Fu, Vinson Go, James Morizio, Jens Christian Hedemann Sørensen, and Albrecht Stroh. 2021. An Intracortical Implantable Brain-Computer Interface for Telemetric Real-Time Recording and Manipulation of Neuronal Circuits for Closed-Loop Intervention. *Frontiers in Human Neuroscience* 15 (Feb. 2021). https://doi.org/10.3389/fnhum.2021.618626
[172] Rina Zelmann, Angelique C Paulk, Ishita Basu, Anish Sarma, Ali Yousefi, Britni Crocker, Emad Eskandar, Ziv Williams, G Rees Cosgrove, Daniel S Weisholtz, et al. 2020. CLoSES: A platform for closed-loop intracranial stimulation in humans. *NeuroImage* 223 (2020), 117314. https://doi.org/10.1016/j.neuroimage.2020.117314
[173] Biao Zhang, Jianjun Wang, and Thomas Fuhlbrigge. 2010. A review of the commercial brain-computer interface technology from perspective of industrial robotics. In *2010 IEEE International Conference on Automation and Logistics*. 379–384. https://doi.org/10.1109/ICAL.2010.5585311
[174] Bingzhao Zhu, Uisub Shin, and Mahsa Shoaran. 2021. Closed-Loop Neural Prostheses With On-Chip Intelligence: A Review and a Low-Latency Machine Learning Model for Brain State Detection. *IEEE Transactions on Biomedical Circuits and Systems* (2021). https://doi.org/10.1109/TBCAS.2021.3112756
[175] Christoph Zrenner, Paolo Belardinelli, Florian Müller-Dahlhaus, and Ulf Ziemann. 2016. Closed-Loop Neuroscience and Non-Invasive Brain Stimulation: A Tale of Two Loop. *Frontiers in cellular neuroscience* 10 (2016), 92. https://doi.org/10.3389/fncel.2016.00092

A ARTIFACT APPENDIX

Abstract

There are 4 artifacts for this paper: hash function library to reproduce Figure 11, a python program to reproduce Figure 12, a basic query processor to generate ILP programs to recreate Figures 8a, 8b, 8c, 9a, 9b. The artifact also contains a Dockerfile to install all requirements and run all tests to provide a push button solution.

Artifact check-list (meta-information)

- **Programs:** glpsol, python3, Docker, NVSim
- **Compilation:** Artifact includes scripts to compile NVSim from source

- **Data set:** Artifact includes data collected from patient with label I001_P013 downloaded from ieeg.org
- **Run-time environment:** Experiments are run on a Docker container running Ubuntu 22.04, Linux 5.19
- **Hardware:** A Linux System with Intel X86-64 CPU, 8 GB RAM.
- **Metrics:** 1) Application throughput calculated using ILP 2) Hash function error rates 3) Packet loss due to Bit Error Rate
- **Output:** Output generates plots in the paper, as described in sections further
- **Experiments:** Experiments measure hash error rates, and the packet loss due to bit errors,
- **How much disk space required (approximately)?:** 15 GB
- **How much time is needed to prepare workflow (approximately)?:** 10-15 minutes
- **How much time is needed to complete experiments (approximately)?:** 30-45 minutes
- **Publicly available?:** Yes, 10.5281/zenodo.7787128
- **Code licenses (if publicly available)?:** CC4
- **Archived?:** 10.5281/zenodo.7787128

Processing Elements

Table 4 lists all our PEs.

A.1 Description

A.1.1 How to access. You can access the artifact at https://zenodo.org/record/7787128

A.1.2 Hardware dependencies. A Linux machine with Docker installed, about 8 GB RAM, and 15 GB free disk space. The instructions to run the experiments are specific to Linux on X86, but the artifacts may work in other environments.

Table 4: Processing Element Names

Name	Function
ADD	Matrix Adder
AES	AES Encryption
BBF	Butterworth Bandpass Filter
BMUL	Block Multiplier
CCHECK	Collision Check
CSEL	Channel Selection
DCOMP	Decompression
DTW	Dynamic Time Warping
DWT	Discrete Wavelet Transform
EMDH	Earth-Mover's Distance Hash
FFT	Fast Fourier Transform
GATE	Gate Module to buffer data
HCOMP	Hash Compression
HCONV	Hash Convolution Operation
HFREQ	Hash Frequency
INV	Matrix Inverter
LIC	Linear Integer Coding
LZ	Lempel Ziv
MA	Markov Chain
NEO	Non-linear Energy Operator
NGRAM	Hash Ngram Generation
NPACK	Network Packing
RC	Range Coding
SBP	Spike Band Power
SC	Storage Controller
SUB	Matrix Subtractor
SVM	Support Vector Machine
THR	Threshold
TOK	Tokenizer
UNPACK	Network Unpacking
XCOR	Pearson's Cross Correlation

A.1.3 Software dependencies. **Docker** Our experiments can be run quickly using scripts on Docker, though we also specify how to run the experiments in python directly.

Software Python3, python libraries (matplotlib, numpy, scipy, statsmodel, python-dtw), glpsol (from glpk-ultils package)

A.1.4 Data sets. Data collected from patient with label I001_P013 downloaded from ieeg.org. This data set contains EEG signals collected from 76 electrodes implanted in the parietal and occipital lobes, recorded at 5 KHz. The dataset was upscaled to about 30 KHz an split to multiple files to simulate multiple BCI devices.

A.2 Installation

For a push button solution, install Docker using your Linux Distribution's software installer. You can then create a container that immediately runs all experiments. Run the following command after extracting the artifact in the folder containing the Dockerfile — `$: docker build -t hull-archive .`

This will start building an Ubuntu container, installing all dependencies, then automatically running the experiments using the scripts we provide.

Alternatively, you may run the experiments locally. The experiments depend on python3 and certain python3 libraries. These libraries can be installed by first installing python3 and pip3 using your distribution's installer. Following that, you can run —

`$: pip3 install -r work/requirements.txt`

inside the root directory of the artifact.

In addition to python, some experiments also use the GNU LP solver, glpsol. This can be installed using your distribution's installer (eg by installing glpk-utils on Ubuntu).

A.3 Experiment workflow

If you set up Docker, the experiment would run automatically. After the docker container is setup, you can copy the results on to the host machine by running the following commands in separate terminals.

```
$docker run –name artifact -it hull-archive
$docker cp artifact:/work /path/in/host/to/store/
```

You can access all results (pdf files) in the respective directories and view them by running `$: ls work/*/*.pdf`.

If you have set up a local install, you can run all experiments by navigating to the `work` folder in artifact, then running —

`$sh script.sh`

Which will start running experiments one by one. This is expected to take 20-30 minutes. Once done, you can access the results in the same way as mentioned above.

A.4 Evaluation and expected results

Hash Error Rates: This experiment evaluates the error rates of the hash functions we describe in the paper, It exists inside `work/hash` directory and is run using —

`$: python3 hash_err_rates.py`

It produces the `hash_hist.pdf` file in the same directory, and should look similar to Figure 10 in the paper. There may be slight differences due to randomization but such errors should be minimal (≈ 1-2% absolute error)

Network Bit Error Rates: This experiment evaluates the impact of bit errors on the end application accuracy, and recreates Figure 12 from the paper. It exists inside `work/ber` directory, and is run using —

```
$ python3 network_ber.py
```

Task throughput This experiment recreates Figures 8a, 8b, 8c in the paper. The figures can be generated directly by running `fig7a.sh`, `fig7b.sh`, and `fig7c.sh` inside the `work/ilp` directory. Each script setups up helper python scripts to use hardware information and application query to generate an ILP program. This ILP program is then solved for an optimal solution using `glpsol`, and then plotted using more helper python scripts. The hardware information of the device is stored in `HALO.json` for convenient access. Please refer to Table 4 to understand function of each PE. Examples of queries are stored in txt files in the same directory, e.g. `seizure_detection.txt` stores the query to run a seizure detection application. The directory also contains a `readme.md` file that explains the grammar of the queries. You may also read the shell script files for examples on running a query.

- `describe_device.py` is a helper python script to generate json files for hardware
- `create_ilp.py` takes in the hardware information json file, a query file, and a target number of devices to generate an ILP program to find an optimal schedule for it on the SCALO system.

Application Level Throughput: This experiment recreates Figures 9a and 9b in the paper. While these experiments may be run on the ILP as well, they take a long time to run due to the larger number of variables and devices in the ILP program. To obtain results faster, we have taken reduced linear equations that resulted from a prior solution to quickly plot solutions for larger problems. This can be generated by running

```
$: python3 work/lineqn/seizure_Plus_hash.py.
```

The equations, and their weights are stored and explained in `utils.py`.

NVSim: This experiment shows the configuration for the NVM used in SCALO. This is stored in `work/NVSim/HULL.cfg`. You can then run `$: ./nvsim HULL.cfg` to view the energy, and bandwidth numbers estimated by NVSim. Particularly, the tool estimates leakage power to be 0.26 mW, and dynamic energies of 918.809 nJ and 1374 nJ per page for reads and writes, respectively.

A.5 Experiment customization

Our scripts are set up to allow easy extension, customization, and experimentation. The hash error program is set up to be run on different input files with a small modification, along with code to allow fast exploration of all parameters of the hash. The ILP is set up to allow queries of different kinds with a readme explaining writing custom queries.

A.6 Methodology

Submission, reviewing and badging methodology:

- https://www.acm.org/publications/policies/artifact-review-badging
- http://cTuning.org/ae/submission-20201122.html
- http://cTuning.org/ae/reviewing-20201122.html