THEME ARTICLE: TOP PICKS FROM THE 2023 COMPUTER ARCHITECTURE CONFERENCES

Distributed Brain–Computer Interfacing With a Networked Multiaccelerator Architecture

Raghavendra Pradyumna Pothukuchi, Karthik Sriram, Michał Gerasimiuk, Muhammed Ugur, Rajit Manohar, Anurag Khandelwal, and Abhishek Bhattacharjee, *Yale University, New Haven, CT, 06511, USA*

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 brainwide 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.

Brain–computer interfaces (BCIs) connect biological neurons in the brain with computers and machines. They can advance our understanding of the brain, help treat neurological/neuropsychiatric disorders, restore lost sensorimotor function, enable novel human–machine interaction, and even enhance personal entertainment.[1]

BCIs sense and stimulate the brain's neural activity using either wearable surface electrodes or through surgically implanted surface and depth electrodes.[1] We have been designing processors for surgically implanted BCIs, which are the cutting edge of neuroengineering. Although they pose surgical risks, implanted BCIs collect far higher fidelity neural signals than wearable BCIs and, hence, are used in state-of-the-art research applications.[1] Many of these devices have received clinical approval, and even more are undergoing clinical trials to restore movement and vision as well as mitigate memory decline.

Until recently, BCIs have simply relayed the neural activity picked up by electrode sensors to computers that process or "decode" that neural activity. However, emerging neural applications increasingly benefit from BCIs that include processing capabilities. Such BCIs enable continuous and autonomous operation without tethering. Many researchers have responded to this need by developing BCIs capable of on-device processing (e.g., Shin et al.,[2] Karageorgos et al.,[3] and Eichler et al.[4]), either through innovative circuit design or with novel processing architectures.[a] We have been targeting the design of next-generation BCIs by using architectural innovation but also by leveraging co-design across the layers.

Implantable BCI processors are challenging to design. They cannot exceed a few milliwatts of power, as overheating the brain by even more than 1°C can damage cellular tissue. At the same time, they must deliver on many performance criteria. First, they must process exponentially growing volumes of neural data within milliseconds. This often also involves computationally intense algorithms for applications like epileptic seizure detection or MI decoding (Table 1). Second, the processors must be flexible to support different algorithms to personalize the computation and to understand or treat different diseases. Finally, they must support applications that process data from *multiple brain sites* over multiple timescales, as neuroscience research is increasingly showing that the brain's

0272-1732 © 2024 IEEE
Digital Object Identifier 10.1109/MM.2024.3411881
Date of publication 24 June 2024; date of current version 14 August 2024.

[a]Our recent review of these approaches is available at https://www.sigarch.org/the-brain-computer-interfacing-landscape-for-computer-architects/.

TABLE 1. Nomenclature used in this article.

Abbreviation	Expansion	Abbreviation	Expansion
ADC	analog-to-digital converter	KF	Kalman filter
ADD	matrix adder	LIC	linear integer coding
AES	AES Encryption	LSH	locality-sensitive hashing
BBF	Butterworth bandpass filter	LZ	Lempel Ziv
BCI	brain–computer interface	MA	Markov chain
BMUL	block multiplier	Mbps	megabits per second
CCHECK	collision check	MC	microcontroller
CSEL	channel selection	MI	movement intent
DAC	digital-to-analog converter	NEO	nonlinear energy operator
DAGs	directed acyclic graphs	NGRAM	hash Ngram generation
DCOMP	decompression	NNs	neural networks
DTW	dynamic time warping	NPACK	network packing
DWT	discrete wavelet transform	NVM	nonvolatile memory
EMDH	earth-mover's distance hash	PEs	processing elements
FC	fully connected	RC	range coding
FFT	fast Fourier transform	SBP	spike band power
GALS	globally asynchronous, locally synchronous	SC	storage controller
GATE	gate module to buffer data	SUB	matrix subtractor
HCOMP	hash compression	SVM	support vector machine
HCONV	hash convolution operation	THR	threshold
HFREQ	hash frequency	TOK	tokenizer
ILP	integer linear programming	UNPACK	network unpacking
INV	matrix inverter	XCOR	cross correlation

functions (and disorders) are based on temporally varying physical and functional connectivity among brain regions. Assessing brain connectivity requires placing communicating implants in different brain regions, with storage that enables multitimescale analysis.

BCIs today (e.g., Sun and Morrell[5]) achieve low power by specializing to a single function or by sacrificing data throughput. Neither option is ideal. BCIs should be flexible to support many applications and personalize their function. They must also support high data throughput to infer more about the brain.

Our previous work developed HALO, a multiaccelerator processor that supports orders of magnitude higher data rates (46 Mb/s) over prior work but is also flexible via programmable interaccelerator dataflow.[3] However, HALO has two key shortcomings. First, it interfaces with only a single brain site, whereas future applications will consist of distributed implants that interface with *multiple* brain sites. Second, HALO or any other BCI does not support the multitimescale signal analyses necessary to decode brain function.

The lack of a BCI capable of real-time distributed processing is hindering the brain sciences. Such systems are also difficult to build, especially as they must rely on wireless networking rather than wired connections that pose a risk of infection. The wireless radios safe for implantation offer only $0.1\times$ data rates compared to what is necessary to operate at the line rate of sensing. This calls for a new distributed computer architecture design strategy from the ground up.

Our solution is *SCALO*, the first architecture for multisite 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 ILP-based scheduler that optimally maps applications to the accelerators and creates network/storage schedules to feed compute. SCALO's programming interface is easily integrated with widely used signal processing frameworks like TrillDSP and MATLAB.

SCALO continues to support HALO's single-implant applications[3] but also enables, for the first time, three new classes of distributed applications: *internal closed-loop* applications that modulate brain activity without communicating with systems external to the BCI (e.g., treatment of epileptic seizure propagation), *external closed-loop* applications (e.g., neural prostheses), and *interactive human-in-the-loop* applications where clinicians query the BCI for data or dynamically adjust its parameters.

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. We show that SCALO is capable of processing up to tens of brain regions and hundreds of megabits per second of neural data within a few milliseconds, which is orders of magnitude higher performance than existing devices. Our technical contributions in architecting SCALO also translate to advances in neuroengineering, and computer system design.

BACKGROUND

BCI Applications and Kernels

The space of BCI applications is rapidly growing.[1] We target three classes of distributed BCI applications that operate in autonomous closed loops. From each class, we study a representative application. We also study spike sorting, a kernel often used to pre-preprocess neural data before subsequent application pipelines. Figure 1 shows these applications. Our original article[6] has more details about these pipelines.

Internal Closed-Loop Applications

Nearly 25 million individuals worldwide suffer from drug-resistant epilepsy. SCALO supports epileptic seizure propagation calculations on device. Figure 1(a) illustrates a typical pipeline. First, seizures are detected locally in each brain site. When a seizure is detected, the neural data at the site are correlated with recent and past signals from other brain sites to identify seizure propagation. Subsequently, correlated regions can be electrically stimulated to mitigate the seizure spread. The pipeline must complete within 10 ms to be effective.[6]

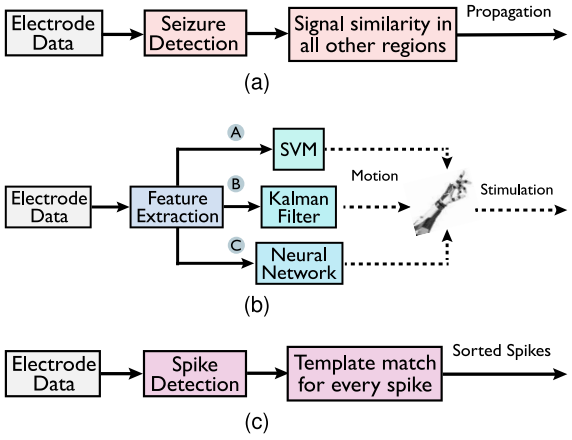

FIGURE 1. Overview of the BCI applications supported by SCALO. (a) Seizure propagation analysis. (b) Decoding movement intent with different approaches. (Source for the prosthetic image: © [gearstd]/Adobe Stock; used with permission.) (c) Spike sorting to separate the combined electrode activity.

External Closed-Loop Applications

These applications help individuals control assistive devices external to BCIs, like artificial limbs, speech decoding systems, or even paralyzed limbs implanted with electrodes. We select three common algorithms representative of this category, shown in Figure 1(b). All algorithms initially extract features relevant to their decoding strategy and apply different classification techniques to identify the intended movement. When the individual has also lost sensory function, the feedback from the movement is emulated by electrically stimulating relevant brain sites. This pipeline must complete within 50 ms.

Human-in-the-Loop Applications

Researchers or clinicians may need to interactively query BCI devices to retrieve important neural data, configure device parameters for personalization, or verify correct operation.

Spike Sorting Kernel

Spike sorting is entirely local to each brain site, but it is a widely used first step for many applications that rely on neuron-level analysis. Each electrode in a BCI usually measures the combined electrical activity of a cluster of spatially adjacent neurons, and spike sorting separates this combined activity into per-neuron waveforms. Figure 1(c) shows a typical pipeline. Spikes detected from electrodes are matched with templates corresponding to each neuron's activity to isolate

per-neuron waveforms. Template matching uses some of the same compute-intensive correlation measures from seizure propagation pipelines, e.g., DTW distance.

SCALO'S DESIGN STRATEGY

SCALO achieves ultrapower-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 NNs; 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 PEs to support points 1–3 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 architecture.[3] By realizing each PE in its independent clock domain, we allow it to be tuned for minimal power to sustain a given application-level processing rate. We use per-implant 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 ultrawideband wireless network. We build our PEs to directly access the network and storage, avoiding the bottlenecks that traditional accelerator-based systems (including ultralow-power coarse-grained reconfigurable arrays) suffer from 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.

THE SCALO ARCHITECTURE

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

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 internode communication bottleneck. Figure 3 shows the refactored applications, made amenable for distributed real-time processing.

First, we split the signal comparison, such as in seizure propagation analysis, into a fast hash check and subsequent exact comparison. We use locality-LSH,[7] and the hash check identifies neural data that are (in high probability) correlated among brain regions and, hence, are necessary for internode exchange. This

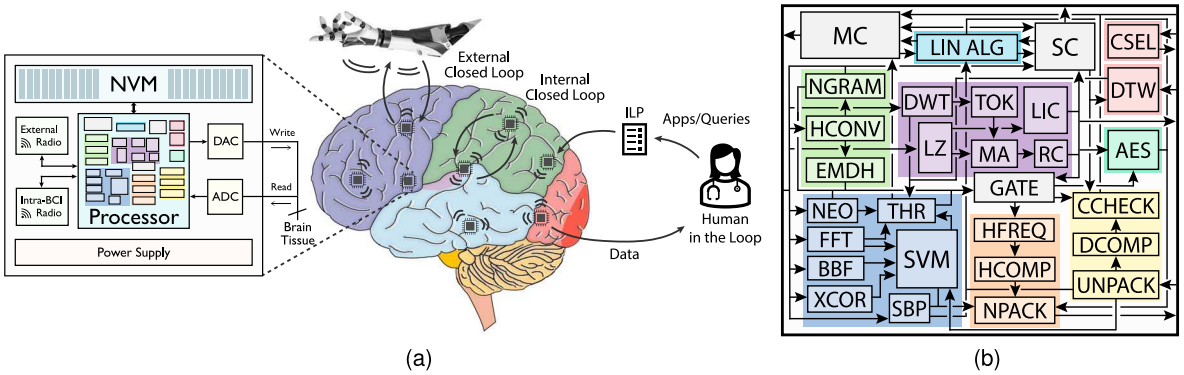

FIGURE 2. The SCALO BCI is a distributed network of nodes implanted in multiple brain sites. The nodes communicate wirelessly with each other and the environment. Each SCALO node has sensors, radios, analog/digital conversion, processing fabric, and storage; the processing fabric contains hardware accelerators and configurable switches to create different pipelines. (a) SCALO overview. (Source for the prosthetic image: © [gearstd]/Adobe Stock; used with permission. Source for the brain image: *Neuron and the Brain* [Walinga and Stangor], CC BY-NC-SA, modified by cropping and removing lines and text; used with permission.) (b) The processor fabric in each of SCALO's nodes.

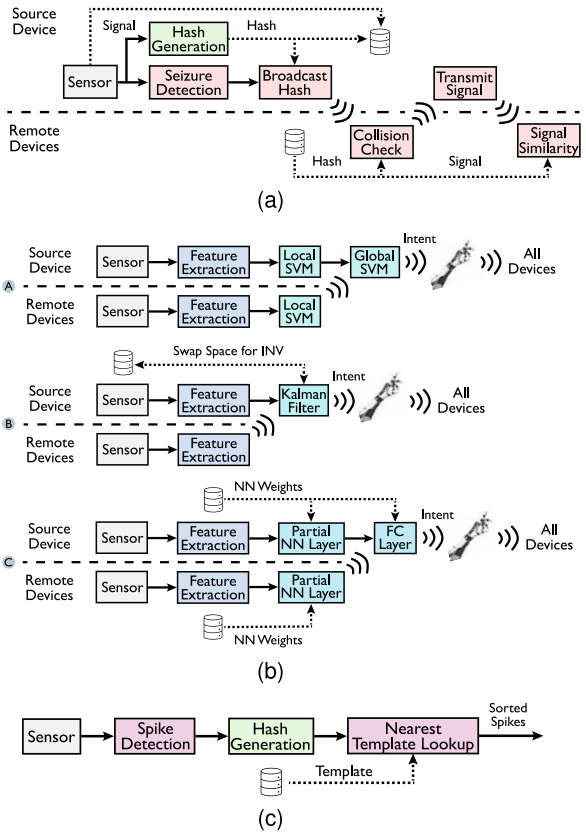

FIGURE 3. Overview of the BCI applications supported for online distributed processing in SCALO. The colors of the steps are matched with Figure 1. (a) Seizure propagation analysis. (b) Decoding MI. (Source for the prosthetic image: © [gearstd]/Adobe Stock; used with permission.) (c) Spike sorting.

avoids communicating or comparing all other data that are unlikely to be correlated across brain regions. Hashes are 100× smaller than signals and can be quickly and accurately generated.

Spike sorting also 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 instead of running an expensive signal similarity computation.

Second, we decompose classifiers like SVMs and NNs to reduce the data being communicated. Instead 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, and 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 KF. The KF generates large matrices as intermediate products from lower-dimensional electrode features and inverts one such matrix. Distributing (and communicating) large matrices over our wireless (and serialized) network violates our response time goals. Therefore, we directly send the electrode features from all sites to a single implant, which computes the filter output, including the intermediate inversion step.

Flexible and Energy-Efficient Accelerators

SCALO's nodes build on our prior work, HALO.[3] In addition to the PEs for single-site applications in HALO, we incorporate new functionality to support distributed applications. SCALO's PEs can be reused across applications and have deterministic latency/power. Wide-reuse PEs minimize the design and verification effort and the on-chip area. Deterministic latency/power enables simple and optimal application scheduling.

Figure 2(b) shows the processor in each SCALO node. There are many PEs connected with programmable switches and a low-power RISC-V MC for miscellaneous functionality. The switches can be configured to realize various processing pipelines.

Our full article[6] describes the design of the new accelerators, but we provide an example highlighting our approach. SCALO needs hardware LSH support for four commonly used signal similarity measures—Euclidean distance, XCOR, DTW distance, and earth mover's distance (EMD). Prior work has proposed an LSH specifically for DTW,[7] but we discover that, by varying this LSH's parameters, it can also serve as a hash for Euclidean distance and XCOR. Our discovery enables the design of a single function that can generate hashes for all three measures. The reconfigurability required to support these different measures does not pose an additional cost since the original DTW hash had to be made configurable anyway to support various deployments. To accommodate the LSH for EMD, we identify a shared dot product with the LSH for DTW. 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).

Finally, the weighted min-hash calculation in the LSH from prior work[7] uses a variable-latency randomization step. We use an alternative method to guarantee deterministic latency and power while preserving the LSH property.

Optimal Power Tuning

Each of SCALO's PEs operates in its own clock domain, similar to our prior work on HALO.[3] However, HALO supported only one frequency per PE. This is not optimal for SCALO's applications, which sometimes operate on only a subset of electrode data. For example, seizure propagation requires an 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 $f_{\max}^{\mathrm{PE}}$, high enough for the maximum data rate, and divide it to $f_{\max}^{\mathrm{PE}}/k$, where k is user programmable. We use multiple frequency rails to ensure the PE has the same latency despite a variable number of inputs.

Per-Implant NVM Storage

Each node integrates 128-GB NVM to store signals, hashes, and application data (e.g., weight matrices and spike templates). We co-design the NVM data layout with PE access patterns to meet millisecond-scale response times.

Networking

SCALO incorporates three networks. From our HALO work,[3] 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 time division multiple access.

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 make 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 electrode signals are processed in it relative to the others when all signals cannot be processed in all tasks.

Programming and Compilation

Clinicians or neuroscientists create programs in popular high-level languages like MATLAB or TrillDSP to describe signal processing pipelines or interactive queries. These are parsed into dataflow DAGs. The DAG and the system's configuration (the latencies and energy of the PEs) are used to formulate an ILP, which is solved to map tasks to PEs and schedule network access. This mapping is translated to assembly code that can be run on the per-node MCs.

EXPERIMENTAL SETUP

We realize SCALO with detailed physical synthesis at a 28-nm fully depleted silicon-on-insulator CMOS process, undergirded by data from our prior partial tape outs of HALO at 12 nm. We use standard cell libraries from STMicroelectronics and foundry-supplied memory macros that are interpolated to 40 °C.

We assume that each node uses a standard 96-electrode array to sense neural activity and has a configurable 16-bit ADC running at 30 kHz per electrode. The ADC dissipates 2.88 mW for one sample from all 96 electrodes. Each node also has a DAC for electrical stimulation, which can consume ≈0.6 mW of power. We use the radio for external communication from HALO[3] and use another radio from prior work[8] for interimplant communication. Finally, we estimate the storage latency and power using NVSim.

Electrophysiologic Data

We use publicly available electrophysiological datasets for our evaluation. For seizure detection and propagation, we use data from the Mayo Clinic of a patient (label "I001_P013") with 76 electrodes implanted in the parietal and occipital lobes. These data were recorded for four days at 5 kHz and are annotated with seizure instances. We upscaled the sampling frequency to 30 kHz and split the dataset to emulate multiple implants. For spike sorting, we use three datasets. Our data sources are described in Sriram et al.[6]

Alternative System Architectures

Table 2 shows the systems that we compare SCALO against. *SCALO No-Hash* uses the SCALO architecture but without hashes. *Central No-Hash* uses a single processor without hashes like most existing BCIs. The processor is connected to the multiple sensors using wires. *Central* is another single-processor design, but it uses hashes like SCALO. Finally, we have *HALO+NVM*, which uses a single HALO processor from prior work,[3] augmented with an NVM to support our applications.

TABLE 2. Alternative BCI architectures.

Design	Architecture	Comparison	Communication
SCALO (proposed)	Distributed	Hash and signal	Wireless
SCALO No-Hash	Distributed	Signal	Wireless
Central No-Hash	Centralized	Signal	Wired
Central	Centralized	Hash and signal	Wired
HALO + NVM	Centralized	Hash and signal	Wired

Since this design does not have our new PEs, it uses the RISC-V processor for tasks like hashing.

EVALUATION HIGHLIGHTS

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 until the available power is fully utilized or the response time is violated. We consider a total of 11 implanted sites, which results in the highest seizure propagation throughput for SCALO and SCALO No-Hash. Our original article[6] evaluates the designs with varying numbers of implants.

Figure 4 shows the 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 MI. 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 MC. 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 MC is slower than running an exact comparison on a PE in *Central No-Hash*. This performance gap highlights the need for hardware acceleration.

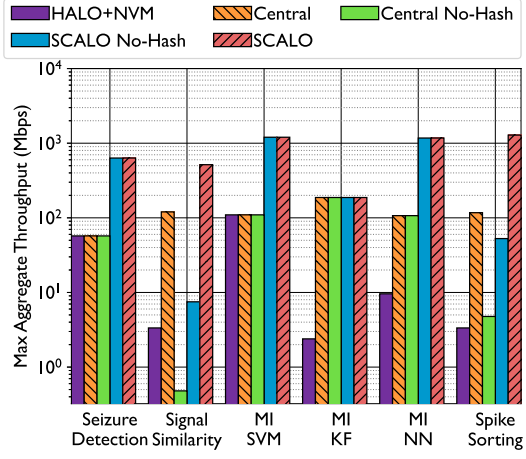

FIGURE 4. Maximum aggregate throughput of SCALO and alternative architectures for 11 nodes.

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.

Central performs best among uniprocessor designs. However, the processor is the bottleneck for multisite interfacing, and *Central* has 10× lower throughput than SCALO for all applications. One exception is the MI with the KF application, where SCALO also centralizes the computations, resulting in a similar throughput.

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 enable scaling to more implants. Compared to *HALO+NVM*, which is the state of the art, SCALO's processing rates are 10× higher for seizure detection and MI KF and are up to 385× higher for the remaining applications.

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 interrelated 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. Although a clinician determines the ultimate choice of weights, we evaluate three sets of weights.

Figure 5(a) 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 Mb/s,

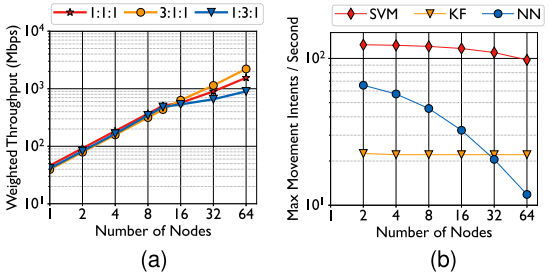

FIGURE 5. Application-level metrics on SCALO. (a) Weighted throughput of seizure propagation. (b) MIs per second.

achieved at 11 nodes. The highest throughput per node is achieved at this node count.

Figure 5(b) shows the maximum number of intents detected per second on SCALO. SCALO significantly outperforms conventional MI SVM and MI NN, which offer only 20 intents per second and for a few electrodes (not shown in the figure). For MI KF, the most complex MI application, SCALO also supports 20 intents per second but can process up to a total of 384 electrodes, up to four nodes for a 96-electrode node.

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. The sorting accuracy of SCALO is within 5% of that achieved by exact template matching.

PATH TO DEPLOYMENT

We are planning ex vivo tests in mice with neuroscientists at the Yale School of Medicine's Center for Brain & Mind Health and the Wu Tsai Institute for the Brain Sciences. Additionally, we are connecting with multiple other research groups beyond Yale to refine SCALO's design to meet their needs.

OTHER ASPECTS OF A MULTI-IMPLANT BCI

Making multi-implant systems viable requires addressing research and engineering beyond the processing that SCALO targets. These include efficient radios, power delivery systems like wirelessly chargeable batteries, novel packaging methods to deploy processing close to the sensors, and safe surgical procedures.

BCI radios that are safe for implantation are growing in efficiency (e.g., Rahmani and Babakhani[8]). However, they still offer only 0.1× the data rate needed to process neural signals at the line rate. This makes communication the most stringent constraint, even more than power, for most of our applications.

Power delivery is an open problem for both distributed and single-implant BCI design. Recent BCIs are using implanted rechargeable batteries with inductive power transfer.[9] Some BCIs include hubs that are chest implanted or scalp mounted, which can serve as wired sources of power for the implants. The hub itself could be powered by removable or wirelessly charged batteries. There has also been recent work on energy-harvesting brain implants based on glucose fuel cells, radio-frequency circuits, the flow of blood and cerebrospinal fluid, or other techniques.[10] While such systems offer sub-1 mW today and are yet to mature, they hold significant promise in realizing wireless distributed BCIs.

Another consideration is the design of suitable electrodes and their packaging with processing. We have assumed that the processor could be deployed close to the sensor. This could be done by bonding the electronics alongside or behind electrode arrays or by using flexible, biocompatible materials when the BCI is realized through electrode strips placed on the surface of the brain. Certain emerging BCI designs (e.g., Neural Dust[11]) use extreme-miniaturized, free-floating "motes" that can collect neural data, and it is an open problem to identify how compute can be added to these systems.

NEW COMPUTER ARCHITECTURE AND SYSTEMS CONSIDERATIONS

SCALO sets the stage for new computer architecture research to expand the performance of the compute and memory subsystems in BCIs and to augment necessary features like reliability and security.

Various research groups have demonstrated new BCI designs targeting applications that are different from ours, e.g., Shin et al.[2] and Eichler et al.[4]). SCALO provides the ideal platform to combine such modules and advance implantable BCIs thanks to its GALS design, which allows each accelerator to run independently. We are keen on exploring methods, e.g., leveraging automated program synthesis methods, to augment SCALO with such new compute capabilities.

Another important line of work is to expand the role of the memory subsystem in augmenting BCI performance. BCI deployments can use different types of sensors/stimulators with varying electrode counts and sensing rates. An ideal BCI processor would interoperate across all such deployments as long as the incoming data throughput remains below the system rating (e.g., 46 Mb/s per node, as with SCALO). However, this is difficult to achieve because the memory structures

inside the PEs are fixed, which limits the maximum number of electrodes that can be read, even if we were to reduce their sensing rate to keep the total ingestion throughput below the maximum rating. We have been exploring the possibility of swapping data between the PEs and the NVM to overcome this problem.[12] We foresee significant innovation to support such features in BCIs, typical in conventional processors, but under extreme resource constraints.

A relevant aspect as we expand BCI use cases is to rethink the role of the CPU. Currently, SCALO takes the CPU out of the loop for execution and even network and storage access. However, this creates its own challenges, like restricting the system to use only static scheduling and fixed processing patterns and, thus, limiting the overall adaptability. A key research direction is to explore what level of CPU involvement is optimal and if the CPU can still be replaced using more accelerators specializing in system services, such as those beginning to appear for conventional systems.

Finally, there is significant new research to strengthen other key features of an implanted BCI, such as reliability, security, and privacy. Being implanted, there are numerous sources of failure that are atypical in systems design, such as the formation of scar tissue on electrodes, micromovements of the implants in the brain, interference with external electronic systems, network errors in the brain tissue, and leakage of blood and cerebrospinal fluid, all of which need to be modeled and protected against. We can also foresee novel security challenges that can use external interference to disrupt the functionality of the devices or gain unintended access with potentially fatal outcomes, which require novel architecture measures to defend against. Additionally, there is also a new aspect of understanding the role of architecture in facilitating their ethical use and the impact of ethics and policy on the BCI architecture, since these devices blur the distinction between the self and the prosthetic.

Overall, we believe our work stimulates new research toward high-precision and high-bandwidth neural interfaces that also broadly advance computer architecture and system design.

CONCLUSION

SCALO enables BCI interfacing with multiple brain regions and provides, for the first time, on-device computation for important BCI applications. 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 nonvolatile 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 the Internet of Things, as well.

ACKNOWLEDGMENTS

This work was supported in part by the Swebelius Foundation, a gift from NetApp, National Science Foundation Award 2118851 and Award 2040682, and a Computing Innovation Fellowship from the Computing Research Association for Raghavendra Pradyumna Pothukuchi (under National Science Foundation Grant 2127309). Numerous individuals helped and supported us to make this work a reality, and we acknowledge them in our full article.[6] The text and figures are adapted from Sriram et al.,[6] CC BY-NC-SA. Raghavendra Pradyumna Pothukuchi and Karthik Sriram are joint first authors. Karthik Sriram and Michał Gerasimiuk completed this work while at Yale University.

REFERENCES

1. M. A. Lebedev and M. A. L. Nicolelis, "Brain-machine interfaces: From basic science to neuroprostheses and neurorehabilitation," *Physiol. Rev.*, vol. 97, no. 2, pp. 767–837, Apr. 2017, doi: 10.1152/physrev.00027.2016.
2. U. Shin et al., "NeuralTree: A 256-channel 0.227-µj/class versatile neural activity classification and closed-loop neuromodulation soc," *IEEE J. Solid-State Circuits*, vol. 57, no. 11, pp. 3243–3257, Nov. 2022, doi: 10.1109/JSSC.2022.3204508.
3. I. Karageorgos et al., "Hardware-software co-design for brain-computer interfaces," in *Proc. ACM/IEEE 47th Annu. Int. Symp. Comput. Archit. (ISCA)*, 2020, pp. 391–404.
4. G. Eichler, L. Piccolboni, D. Giri, and L. P. Carloni, "MasterMind: Many-accelerator SoC architecture for real-time brain-computer interfaces," in *Proc. Int. Conf. Comput. Des. (ICCD)*, 2021, pp. 101–108.
5. F. T. Sun and M. J. Morrell, "The RNS system: Responsive cortical stimulation for the treatment of refractory partial epilepsy," *Expert Rev. Med. Devices*, vol. 11, no. 6, pp. 563–572, Aug. 2014, doi: 10.1586/17434440.2014.947274.
6. K. Sriram et al., "SCALO: An accelerator-rich distributed system for scalable brain-computer interfacing," in *Proc. Int. Symp. Comput. Archit. (ISCA)*, 2023, pp. 563–572, doi: 10.1145/3579371.3589107.
7. C. Luo and A. Shrivastava, "SSH (sketch, shingle, & hash) for indexing massive-scale time series," in

Proc. NIPS Time Series Workshop, PMLR, 2017, pp. 38–58.
8. H. Rahmani and A. Babakhani, "A wirelessly powered reconfigurable FDD radio with on-chip antennas for multi-site neural interfaces," *IEEE J. Solid-State Circuits*, vol. 56, no. 10, pp. 3177–3190, Oct. 2021, doi: 10.1109/JSSC.2021.3076014.
9. E. Musk, "An integrated brain-machine interface platform with thousands of channels," *J. Med. Internet Res.*, vol. 21, no. 10, 2019, Art. no. e16194, doi: 10.2196/16194.
10. L.-G. Tran, H.-K. Cha, and W.-T. Park, "RF power harvesting: A review on designing methodologies and applications," *Micro Nano Syst. Lett.*, vol. 5, no. 1, Feb. 2017, Art. no. 14, doi: 10.1186/s40486-017-0051-0.
11. R. M. Neely, D. K. Piech, S. R. Santacruz, M. M. Maharbiz, and J. M. Carmena, "Recent advances in neural dust: Towards a neural interface platform," *Current Opin. Neurobiol.*, vol. 50, pp. 64–71, Jun. 2018, doi: 10.1016/j.conb.2017.12.010.
12. M. Ugur, R. P. Pothukuchi, and A. Bhattacharjee, "Swapping-centric neural recording systems," in *Proc. 15th Annu. Non-Volatile Memories Workshop*, 2024, pp. 1–2.

RAGHAVENDRA PRADYUMNA POTHUKUCHI is an associate research scientist and a Computing Research Association/National Science Foundation Computing Innovation Fellow at Yale University, New Haven, CT, 06511, USA. His research interests include computer architecture and systems, brain–computer interfaces, and quantum computing for cognitive modeling. Pothukuchi received his Ph.D. degree in computer science from the University of Illinois at Urbana-Champaign. He is a Member of IEEE and the Association for Computing Machinery. Contact him at raghav.pothukuchi@yale.edu.

KARTHIK SRIRAM is a software engineer at AMD, San Jose, CA, 95124, USA. His research interests include computer systems and architecture as well as hardware–software co-design, especially in the design of brain–computer interfaces. Sriram received his Ph.D. degree in computer science from Yale University. Contact him at mckarthik7.github.io.

MICHAŁ GERASIMIUK is a Ph.D. student in computer science at Stanford University, Stanford, CA, 94305, USA. His research interests include brain–computer interfaces, systems neuroscience, and the intersection of computing and biotechnology. Gerasimiuk received his B.S. degree in computer science from Yale University. He is a student member of the Association for Computing Machinery. Contact him at gerasimiuk@stanford.edu.

MUHAMMED UGUR is a Ph.D. student in computer science at Yale University, New Haven, CT, 06511, USA. His research interests include computer architecture and systems, specifically hardware–software co-design for emerging neural interfaces. Ugur received his M.S. degree in computer science and engineering from the University of Michigan. He is a Student Member of IEEE and the Association for Computing Machinery. Contact him at muhammed.ugur@yale.edu.

RAJIT MANOHAR is the John C. Malone Professor of Electrical Engineering and a professor of computer science at Yale University, New Haven, CT, 06511, USA. His research interests include the design and implementation of asynchronous circuits and systems. Manohar received his Ph.D. degree in computer science from Caltech. He is a Fellow of IEEE. Contact him at rajit.manohar@yale.edu.

ANURAG KHANDELWAL is an assistant professor of computer science at Yale University, New Haven, CT, 06511, USA. His research interests include problems in computer systems and networks. Khandelwal received his Ph.D. degree in computer science from the University of California, Berkeley. He is a member of the Association for Computing Machinery. Contact him at anurag.khandelwal@yale.edu.

ABHISHEK BHATTACHARJEE is a professor of computer science at Yale University, New Haven, CT, 06511, USA. His research interests include computer architecture and systems at all scales of computing, ranging from server systems for large-scale data centers to embedded systems for implantable brain–computer interfaces. Abhishek received his Ph.D. degree in electrical engineering from Princeton University. He is a Member of IEEE and the Association for Computing Machinery. Contact him at abhishek@cs.yale.edu.