Optimizing Neural Recording Front-Ends Toward Enhanced Spike Sorting Accuracy in High-Channel-Count Systems

Yunzhu Chen, *Graduate Student Member, IEEE*, Xiaolin Yang, *Member, IEEE*, Georges Gielen, *Fellow, IEEE*, and Carolina Mora Lopez, *Senior Member, IEEE*

***Abstract*— Spike sorting is a pivotal signal-processing technique used to extract information from raw extracellular recordings. Its performance is influenced by the characteristics of the neural recording front-end. This study explores how design choices in amplifiers, filters, and analog-to-digital converters (ADCs) affect the accuracy of well-established spike sorting algorithms. Our primary objective is to identify the minimal requirements that ensure high sorting accuracy while facilitating power- and area-efficient analog front-ends, which is especially needed for multi-channel recording-only applications. To achieve this, we use both synthetic and real datasets, serving as ground truth, processed through a generic MATLAB model of a neural recording front-end that simulates key electrical parameters impacting the signal integrity. These include the filter order and cutoff frequency, ADC resolution, ADC sampling frequency, and nonlinearity. Our findings indicate that optimal spike-sorting results are obtained with a 1st-order bandpass Butterworth filter ranging from 700 Hz to 7.5 kHz, coupled with an ADC that offers a 15-kHz sampling frequency at 8-bit resolution and no missing codes. These insights are crucial for designing high-channel-count neural interfaces where CMOS circuits must efficiently be optimized.**

***Index Terms*— Spike sorting, neural-recording front-end, high channel count, extracellular recording.**

I. Introduction

NEUROSCIENCE has greatly benefited from extracellular recording, a method that enhances our understanding of brain activity by locally capturing and processing the electrical signals emitted by neurons. This technique also provides researchers with detailed insights into the behavior of individual neurons. Depending on the research questions and experimental designs, various methods are used to analyze the neurons' action potentials (APs) or spikes, aiming to extract meaningful information about the properties and dynamics of the neuronal activity. One essential method is spike sorting, which involves classifying spikes into clusters based on their waveform characteristics to distinguish the signals originating from different neurons. Accurate classification and categorization of these spikes allow researchers to understand the firing and communication patterns of individual neurons, as well as their interactions within neural networks. High-performance spike sorting requires a meticulous analysis of the signal quality in extracellular recordings and the optimization of the algorithms used in the sorting process.

TABLE I
SPECIFICATIONS OF STATE-OF-THE-ART HIGH-CHANNEL-COUNT NEURAL-RECORDING-ONLY FRONT-ENDS

	[7]	Npix 2.0 [10]	SiNAPS [6]	Neuralink [8]	[9]
Application	LFP+AP	LFP+AP	LFP+AP	LFP+AP	LFP+AP
No. of Channels	1536	384	1024	1024	144
Bandwidth [Hz]	0.5-10k	0.5-10k	1-1k (LFP) 300-7.5k (AP)	5-1k (LFP) 300-10k (AP)	0-10k
Sampling Frequency [Hz]	30k	30k	25k	20k	20k
Input Noise [μV_{rms}]	6.01	7.74	6.5	8.9	13.43
Resolution [bits]	12	14	12	10	11
THD [%]	0.24%@ 10mVpp	0.17%@ 10mVpp	Not reported	0.57% @-0.79dBFS	Not reported

Recent research in neural-recording integrated circuits focuses on improving the area and power efficiencies while ensuring optimal signal quality [1], [2], [3], [4], [5]. As shown in Table I, typical specifications of published state-of-the-art high-channel-count recording systems [6], [7], [8], [9], [10] include a sampling frequency of 20-30 kHz, a resolution of 10-12 bits, an input-referred noise lower than 10 μV_{rms} in the AP band, and a total harmonic distortion (THD) lower than −40dB. Although these specifications are intended for high-quality signal capture, their necessity for effective spike sorting remains unclear, especially in multi-channel readouts

Received 6 November 2024; revised 6 April 2025 and 14 May 2025; accepted 25 May 2025. Date of publication 29 May 2025; date of current version 9 June 2025. *(Corresponding author: Carolina Mora Lopez.)*

Yunzhu Chen and Georges Gielen are with imec, 3001 Leuven, Belgium, and also with the Department of Electrical Engineering (ESAT), KU Leuven, 3000 Leuven, Belgium.

Xiaolin Yang and Carolina Mora Lopez are with imec, 3001 Leuven, Belgium (e-mail: Carolina.MoraLopez@imec.be).

Digital Object Identifier 10.1109/TNSRE.2025.3574917

where more relaxed requirement that minimizes the power and area are crucial.

In parallel, significant efforts have been dedicated to the development of accurate spike sorting algorithms. This research spans both software [11], [12], [13], [14], [15] and hardware [16], [17], [18], [19], [20], [21], [22], [23], [24] implementations, with numerous studies, such as [25], providing comprehensive comparisons and evaluations of the prevalent algorithms used in the spike-sorting workflows, including spike detection, feature extraction, and clustering.

Despite the advancements in these two areas, the relationship between the required neural recording front-end performance and the spike sorting accuracy has not been investigated thoroughly. Some initial studies have begun exploring this relationship. For instance, the work in [26] introduces a novel dual-path spike detection algorithm and analyzes the performance of various spike detectors across different sampling frequencies and bit resolutions. Similarly, [27] reports the development of a two-stage spike sorting algorithm, with an emphasis on determining the minimal requirements for the neural-recording front-end. Additionally, the study in [28] examines the hardware parameter relaxation for power savings in wireless brain-computer interfaces (BCIs), considering parameters such as noise, the filter cut-off frequency, and the ADC resolution. These studies, however, have limited scope since [26] focuses only on spike detection where spike morphology is not critical, [27] uses only a single algorithm (WAVEclus [15]) for spike sorting, and the decoding application in [28] primarily relies on binned spike counts where the spike waveform information is not important. Another limitation is that the datasets used in these studies are based on single-channel recordings or simplistic multi-channel extensions, thus failing to accurately represent the spatial correlations observed in actual neuronal spike patterns. Furthermore, the spike amplitudes of these datasets are normalized or simply divided into a few signal-to-noise ratio (SNR) scenarios, thus neglecting the up-to 10x amplitude variations that occur in real multi-channel neural recordings [29]. Also, nonlinearities such as harmonics and missing/sticky codes, which are important front-end design specifications, are not taken into account in the above studies. In summary, a comprehensive and systematic analysis of the interplay between the neural recording front-end and spike sorting performances is yet to be realized in full.

This work aims to systematically study how different front-end design choices affect the sorting accuracy of well-established spike-sorting algorithms. Our main goal is to derive the minimum required specifications of the analog front-end that can guarantee good spike-sorting accuracy, which is crucial to designing implantable high-density neural interfaces with minimum area and power. In this work, we focus on optimizing circuit performances specifically for high-density neural recording and spike-sorting applications. Therefore, aspects related to bidirectional or closed-loop neural interfaces, such as stimulation artifact tolerance, are not considered in this study. Additionally, this work does not seek to review or compare different recording architectures and their power/area trade-offs, as these aspects are covered in other dedicated studies [30], [31], [32]. Instead, our focus is on optimizing the performance requirements of the key building blocks of neural recording front-end design, ensuring that the proposed optimizations remain as applicable as possible across various recording architectures. Our study incorporates three key innovations:

1) A comprehensive neural recording front-end model is developed, including key electrical parameters that affect the neural signal integrity;
2) Multiple state-of-the-art spike-sorting algorithms are employed for the analysis, ensuring the reliability and universality of the findings;
3) High-channel-count datasets are used, incorporating the spatial relationships and redundancy among channels in high-density neural recording.

The remaining of the paper is organized as follows. The methodology, including the neural datasets used as ground truth, neural-recording front-end modeling and spike sorting algorithms, will be presented in Section II. Section III will benchmark the performance of different spike-sorting algorithms against different front-end requirements. The conclusions will be drawn in Section IV

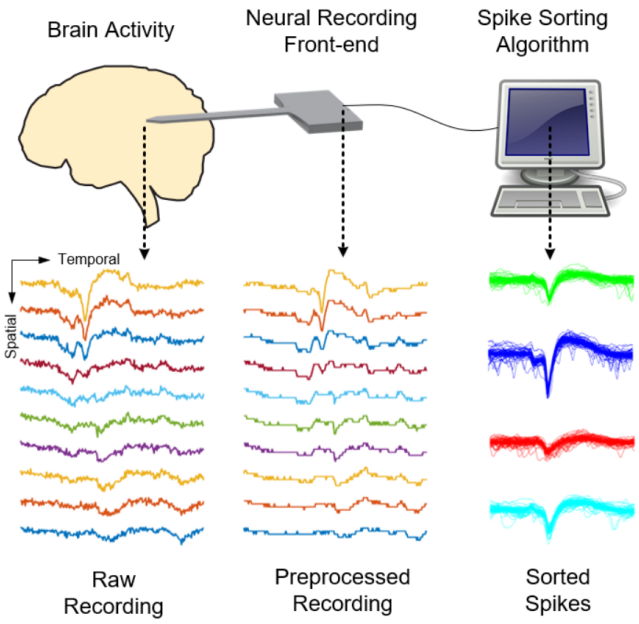

Fig. 1. Study methodology overview.

II. Methodology

A. Methodology Overview

Fig. 1 illustrates the methodology of the proposed study. Initially, we generate synthetic datasets to replicate raw extracellular recordings induced by brain activity, including complete ground truths. These raw recordings, together with real datasets, are then preprocessed through a parametrized model of a neural recording front-end. This step is designed to replicate the impact of various front-end design parameters, such as the filter order, quantization noise and nonlinearity, on the signal quality. These parameters have a direct impact on the area and power consumption of the front-end circuits.

Following this, we apply multiple established spike-sorting algorithms to the processed data to isolate and categorize the spikes. This process enables a detailed examination of the spike-sorting performance across different algorithms and front-end requirements. The insights gained from this study are intended to guide researchers and engineers in developing neural recording front-ends that have minimal requirements while achieving a consistent and reliable spike-sorting performance.

B. Ground-Truth Datasets

Ground-truth data is crucial to validate spike-sorting algorithms quantitatively. These data can be obtained with three different methods: paired recordings, hybrid recordings, and synthetic recordings [33]. In this work, we employ both synthetic and real datasets to comprehensively evaluate spike sorting performance. Synthetic datasets generated with biophysically-detailed simulators because of the following advantages: i) all the neurons making up the recordings are known, which allows for full and exhaustive benchmarks of the sorting results; ii) it is possible to model many neural sources at different firing rates and noise levels to emulate the complexity of real recordings; and iii) it is possible to generate high-channel-count datasets emulating the spatial relationships of well-stablished neural probes such as Neuropixels [5], [10]. Although synthetic datasets may not entirely reflect the intricacies of actual biological noise, their thoroughness and exceptional controllability make them a valuable tool for algorithm development. The inclusion of real datasets complements this approach by ensuring that the methods are also evaluated under biologically realistic conditions.

In our study, we use the open-source Python-based software MEArec [34] to create synthetic ground-truth datasets. MEArec is a versatile simulator that offers fast, user-friendly, and biophysically accurate extracellular recordings, ideal for testing and optimizing spike sorting algorithms. This tool supports multi-channel probes and can be integrated with the SpikeInterface platform [35], which will be discussed in detail in Section II-D. Within MEArec, we utilize the topology of a 128-channel Neuropixels 1.0 probe [5] to simulate a high-channel-count recording system. To assess the performance of various spike sorting algorithms, we have chosen a standard recording duration of 10 minutes. A cohort of 60 neurons is employed, comprising 48 excitatory and 12 inhibitory cells. Two distinct cell types are also characterized by two default spike firing rates of 5 Hz and 15 Hz. The datasets contain far-neuron colored noise, typically at a level of 10 μV_{rms}. The spike amplitudes vary from 40 to 300 μV_{pp}. The resulting SNR ranges from 8.5 to 48.5 (i.e., 18.6 dB to 33.7 dB), with a mean SNR of 19.2 (corresponding to 25.7 dB). The SNR is calculated following the definition in [36], after applying a commonly used bandpass filter (300–6000 Hz, 5^{th}-order Butterworth, zero-phase filter using filtfilt).

To further validate the robustness of our results and conclusions, we have extended our simulations to include real paired neural recordings. For this, we utilized the Paired-Boyden datasets available in the open-source platform SpikeForest [36]. The specific dataset used in our simulations

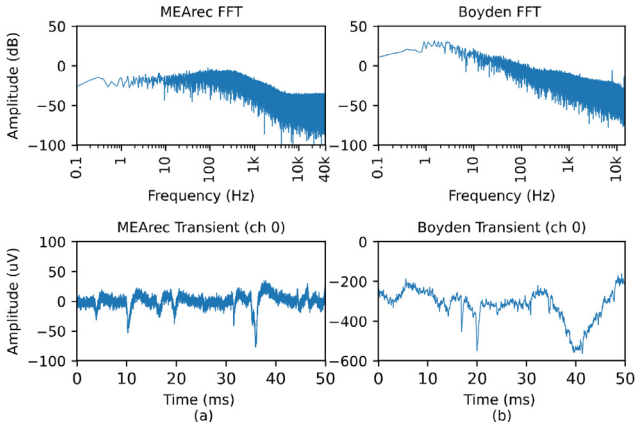

Fig. 2. (a) Spectrum and transient waveform of the MEArec datasets. (b) Spectrum and transient waveform of the real Paired-Boyden datasets.

(419_1_7) contains 32 channels, with a spike amplitude of 305 μV_{pp} and an SNR of 27 (i.e., 28.6 dB).

The FFT spectra and transient waveforms of both the MEArec and Paired-Boyden datasets are depicted in Fig. 2. As shown, the MEArec recordings contain only far-neuron noise and lack low-frequency local field potentials (LFPs). In contrast, the Paired-Boyden dataset, being a real neural recording, includes both high-frequency spike-related noise and low-frequency LFP components, reflecting the broader spectral characteristics of actual biological recordings.

C. Neural Recording Front-End Model

1) **Front-End Model Overview** We have developed a detailed generic MATLAB model of a typical neural recording front-end, consisting of three blocks: a low-noise amplifier (LNA), a band-pass filter (BPF), and an analog-to-digital converter (ADC). First, the neural signal is amplified by the LNA, then filtered to remove out-of-band noise, and finally digitized by the ADC for subsequent digital processing such as spike sorting. In such a neural recording front-end, the various blocks present different design choices, and introduce various types of nonidealities. These design choices and nonidealities are illustrated in Fig. 3 and will be explained in detail below.
Although various readout architectures exist (e.g. dedicated ADC per channel [37], [38], time-multiplexed ADC [5], [7], [8], [10], and direct-to-digital approaches [1], [4], [9], [39], [40]), each offering different area versus power trade-offs, our analysis is conducted at the building block level rather than the architectural level. As a result, our findings remain applicable across different architectures, beyond the typical configuration considered in this study.
2) **Amplifier Modeling** Amplifiers introduce two major nonidealities in the system: nonlinearity and noise. Harmonic distortion is the main factor to consider regarding nonlinearity, with a transfer function $y = f(x)$ typically represented by the following power series [41]

$$y = f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \dots \quad (1)$$

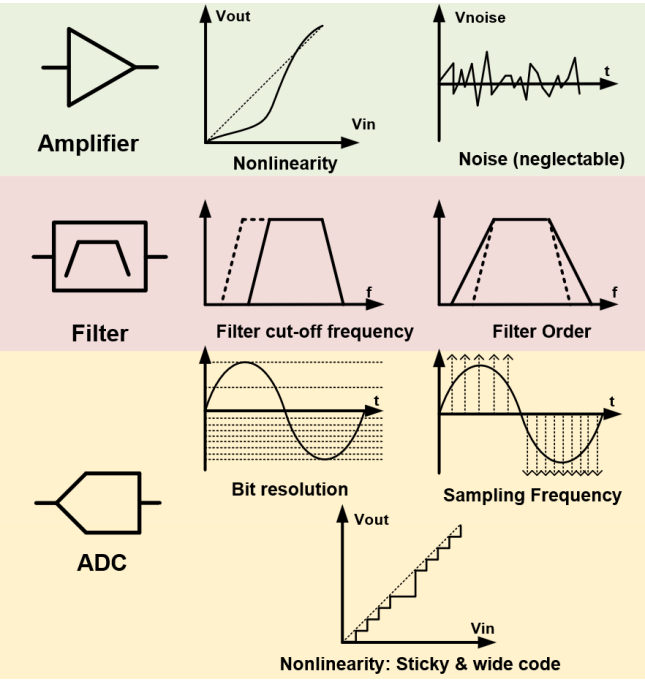

Fig. 3. Design choices and nonidealities considered in the neural recording front-end model.

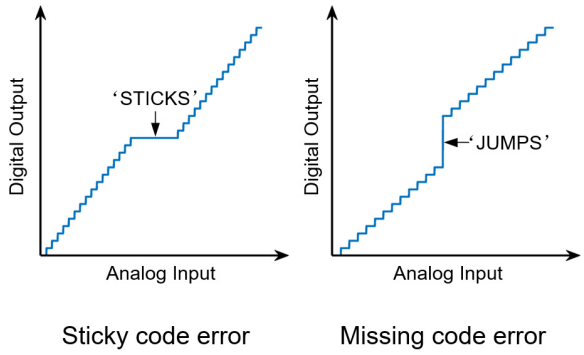

Fig. 4. Representation of sticky codes and missing codes in an ADC output.

In this series, the term a_0 is the DC component, a_1 is the small-signal gain, a_2x^2 represents 2$^{\text{nd}}$-order nonlinearity, and a_3x^3 represents 3$^{\text{rd}}$-order nonlinearity. Higher orders of harmonic distortion, such as fourth and beyond, are typically less significant and, therefore, often neglected. Since amplifiers are usually designed to be differential, the 2$^{\text{nd}}$-order harmonic distortion can be cancelled out. However, the 3$^{\text{rd}}$-order harmonic distortion (HD_3) remains. In harmonic studies, we typically use a sine wave as the input signal, $x = \sin(2\pi ft)$. When this signal is processed, distortion can create new frequencies, like $2f$and $3f$. The 3$^{\text{rd}}$-order harmonic distortion is measured by the ratio of the amplitude at $3f$ to the amplitude at the fundamental frequency f. This ratio is given by $HD_3 \approx \frac{a_3}{4a_1}A^2$ [41], where A represents the amplitude of the signal relative to the amplifier's full-scale input. We assume $A = 1$ to consider the worst-case scenario, even if the signal does not reach the system's limits.

In the design of neural-recording analog front-ends, a 3$^{\text{rd}}$-order harmonic distortion of up to 1% (i.e. -40 dB) is traditionally considered acceptable, but the rationale behind this limit is unclear. In our study, we examine a range of 0% to 5% for HD_3 to determine how much distortion can be tolerated without significantly affecting spike-sorting accuracy.

As will be detailed later, the front-end noise is not included in this study, as it is typically optimized such that it does not significantly contribute to the overall system noise.

3) **Filtering Modeling** To define the signal band in our neural-recording front-end model, we select a Butterworth filter because of its flat passband response, a common choice in analog filter design. Two crucial aspects of the filter design are the filter order and the passband cutoff frequencies. A higher filter order results in a sharper transition between the passband and the stopband, but it can increase the phase distortion. The cutoff frequency, which helps to eliminate unwanted signal components and noise, is usually selected as a trade-off between noise suppression and signal loss.

In this study, we compare 1$^{\text{st}}$-, 2$^{\text{nd}}$- and 3$^{\text{rd}}$- order Butterworth filters. The low-pass cutoff frequency is set at the Nyquist frequency, which is half of the sampling frequency f_s. For the high-pass cutoff frequency, we sweep it from 100 Hz to 2000 Hz to determine the optimal point for spike sorting performance. It is important to note that most spike sorters include their own filters in their preprocessing steps. To ensure a fair comparison in our study, these built-in filters are disabled.

4) **ADC Modeling** To model the impact of the ADC on the spike-sorting accuracy, we examine three main properties: the sampling rate, the resolution, and the nonlinearity.

Sampling rate: The sampling rate is usually chosen as a trade-off between the power consumption and the signal loss. Here, we compare different sampling rates ranging from 6 kHz to 40 kHz, aligning with the typical 20-30 kHz used in most AP recording systems.

Resolution: To assess the impact of the quantization error, we vary the ADC resolution (defined based on the SNR, as distortion is treated separately) from 6 to 14 bits, maintaining a fixed dynamic range of $\pm 500\ \mu$V, which corresponds to the typical maximum amplitude of the AP signals measured extracellularly. This ADC resolution range covers the common resolutions of 8 to 12 bits found in most neural recording front-ends.

Nonlinearity: Since harmonic distortion is already addressed in the amplifier model, our ADC modeling focuses on missing and sticky code errors [42] as illustrated in Fig. 4. Sticky code errors are depicted as horizontal 'sticks' in the transfer function, indicating that the output remains 'stuck' at a digital code over

TABLE II
DEFAULT MODEL SETTING FOR NEURAL-RECORDING FRONT-ENDS

Circuit Component	Characteristic	Default setting
Amplifier	Harmonic Distortion	0%
Filter	High-pass order	1st order
	High-pass cutoff	300 Hz
	Low-pass order	1st order
	Low-pass cutoff	15 kHz
ADC	Sampling Frequency	30 kHz
	Resolution	10 bits
	Sticky-code error	0 codes
	Missing-code error	0 codes

a broad range of analog inputs. Conversely, missing code errors are shown as abrupt vertical 'jumps' or gaps where certain output values are skipped. These errors typically arise from component mismatches within the circuit and are mostly prevalent in the mid-range of the transfer function, particularly when all digital bits are active during conversion (e.g., transitioning from 0111 to 1000 in a 4-bit ADC). In this study, we evaluate the impact of 1, 2, 4, and 6 consecutive missing and sticky codes, occurring at seven transition points within the full ADC range: 1/8, 1/4, 3/8, 1/2, 5/8, 3/4, and 7/8.

5) **Default Model Settings** To compare how the various design choices in each of the blocks discussed above affect the spike sorting accuracy, we have established default settings for our analysis. These default settings, presented in Table II, are aligned with the most widely used design specifications in state-of-the-art analog front-ends. In the experiments described in Section III, these default values are used unless otherwise specified.

6) **Other Relevant Parameters Not Considered in This Study** While this study focuses on the impact of the front-end performance parameters on the spike-sorting accuracy, several other factors that mainly influence the neural recording performance and less the spike sorting are not explicitly analyzed here.

LNA Noise: Higher noise levels reduce the SNR, impacting the spike detection and sorting accuracy. However, the total system noise includes electrode noise and biological background activity noise, which often dominate over the amplifier noise. To prevent the LNA noise from becoming the limiting factor, a widely accepted design guideline is to keep it between 5 to 10 μV_{rms}. Given these well-established constraints, the LNA noise is not explicitly analyzed in this study.

Input impedance: A sufficiently high input impedance is critical to minimize the signal attenuation, preserving the SNR, and ensuring accurate spike detection. Additionally, high input impedance helps maintain a high common-mode rejection ratio (CMRR), which is essential for artifact suppression, particularly for stimulation and movement artifacts [39], [40]. A commonly accepted guideline is to keep the input impedance at least 100× higher than the electrode impedance. However, since the electrode properties (size, material, and chronic degradation) largely dictate this requirement, the input impedance is not a focus of this study.

Crosstalk: While crosstalk can be a concern in high-channel-count neural probes, worst-case reports in the literature indicate that crosstalk levels remain well below 1% [5], [43], [44], [45], which is too low to meaningfully affect spike sorting. Given that the crosstalk level in well-designed AFEs is below the noise floor of the system, its impact on the spike sorting accuracy is negligible, and thus it is not further considered in this study.

*Stimulation and movement artifacts:*This study focuses on recording-only AFEs, where stimulation artifacts are not a concern. In stimulation-based applications, artifact severity depends on multiple external factors (e.g., electrode impedance, reference schemes, and stimulation pulse characteristics), making it difficult to generalize in a model-based study. Various system-level techniques can be implemented to tolerate or mitigate stimulation artifacts, though they do not necessarily have a direct impact on spike-sorting accuracy. Movement artifacts also depend on external factors such as electrode matching and reference configurations, and they can be effectively mitigated through post-processing techniques such as common average referencing (CAR) [46], [47]. Since our study is concerned with optimizing front-end parameters under controlled recording conditions, artifacts are outside its scope.

D. Spike-Sorting Algorithm Platform and Performance Metrics

After the raw signals pass through the previously described neural recording front-end model, it is essential to employ well-established spike-sorting algorithms to evaluate spike-sorting performance in a coherent and consistent methodology.

1) **State-of-the-Art Spike Sorting Algorithms**
We use SpikeInterface [35] as a comprehensive framework for spike sorting, fulfilling two essential requirements. Firstly, it encapsulates a variety of spike-sorting methods, facilitating easy invocation and execution of these sorters. Secondly, it provides standardized and contemporary approaches for evaluating the sorting performance. The seamless integration of the MEArec datasets with SpikeInterface further streamlines the implementation and comparison of spike sorters.

In this study, we compare 5 different spike-sorting algorithms: HerdingSpikes [14], Tridesclous [48], Spyking Circus 2 [11], Kilosort (3 and 4) [12] and IronClust [13]. These algorithms are specifically designed for high-channel-count neural recording, supporting up to 5000 channels. They primarily employ two clustering methodologies: Kilosort, Tridesclous and Spyking Circus 2 are based on template matching [49],

whereas IronClust and HerdingSpikes rely on density-based clustering [50]. While Kilosort3 is used for the MEArec datasets, Kilosort4 is employed for the Paired-Boyden dataset, as Kilosort3 was unable to reliably sort this dataset and resulted in inconsistent results.

2) **Performance Metrics**

To assess the performance of the spike sorting, ground-truth neurons and detected units are matched using the "Best Match" method with a best score threshold of 0.1, following the evaluation methodology outlined in SpikeForest [36]. The Best Match method pairs each ground-truth neuron with the detected unit that exhibits the highest accuracy. The 0.1 best-score threshold stipulates that only units with an accuracy greater than 0.1 are included in the calculations. The performance is evaluated using three common metrics: accuracy, good units, and bad units. Accuracy can be computed individually for each neuron, and it is defined as follows [35]:

$$ACC_{N(i)} = \frac{TP}{FP + TP + FN}, \tag{2}$$

where TP, FP and FN refer to true positives, false positives, and false negatives, respectively. Fig.4(a) illustrates the concept of TP, FP and FN using a Venn diagram with two circles, N and U. Circle N represents all the spikes from one ground truth neuron, while circle U represents all the spikes in a sorted cluster. The intersection of N and U shows the TP, which are spikes correctly classified into the cluster. The area of N outside U represents the FN, which are spikes in the ground truth neuron that were not sorted into the cluster. The area of U outside N represents the FP, which are spikes in the sorted cluster that do not belong to the ground-truth neuron.

While accuracy evaluates performance from the perspective of ground-truth neurons, the "good" and "bad" unit metrics assess the performance at the population level [35]. Fig. 5(b) illustrates this distinction. The ground truth contains 5 neurons (*N1-N5*), which are sorted by an algorithm into 8 units (*U1-U8*). "Good" units, like *U1* and U5, have high accuracy (>0.8) with matched neurons. "Bad" units include over-merged, redundant, and FP units. For example, *U4* is over-merged, showing relatively high accuracy (>0.2) for multiple ground-truth neurons *(N3, N4)*. *U8* is redundant, with relatively high accuracy (>0.2) but not the best match (since *U5* is the best match for *N5*). FP units, such as *U3*, *U2* and *U7*, have low accuracy (<0.2). Notably, *U2* and *U7*, consisting solely of noise without ground-truth neurons, are excluded from accuracy calculations. This means that sorters can sometimes exhibit a high accuracy while still having many bad units, indicating potential issues in unit classification.

Fig. 6 presents the baseline performance of the five selected sorters and the combination method on the MEArec dataset. The performance is evaluated using the metrics of accuracy, good units, and bad units. This dataset features a 30-kHz sampling frequency,

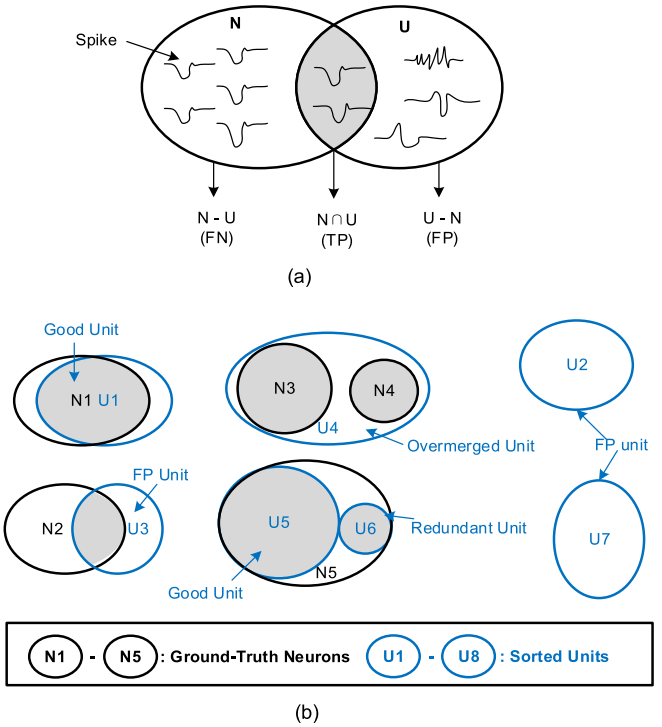

Fig. 5. (a): Venn diagram illustrating true positives (TP), false positives (FP), and false negatives (FN) in spike sorting. (b) Venn diagram illustrating the concepts of good, over-merged, redundant, and FP units.

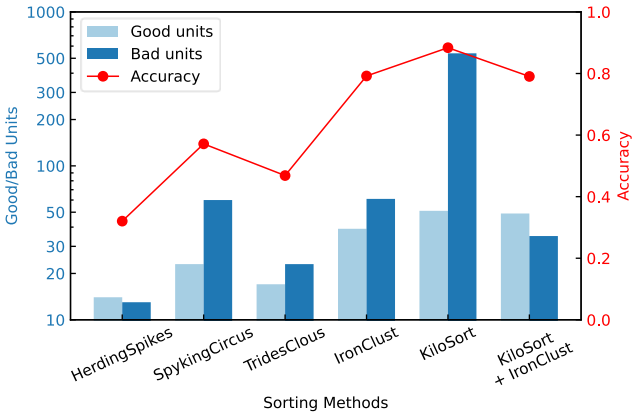

Fig. 6. MEArec dataset baseline performance comparison of the five selected sorters and a combination method based on accuracy, good units, and bad units. Kilosort shows high accuracy but many bad units, while others have lower accuracy with fewer bad units.

floating-point precision, no filtering, and no nonlinearity. Notably, Kilosort shows exceptionally high accuracy and a large number of good units, but also a significant number of bad units, mostly FP. This underscores the need for post-processing, such as manual curation, to refine the results by rejecting, merging, or splitting clusters based on spike features. Tridesclous, Spyking Circus 2 and IronClust, while yielding lower accuracy and fewer good units, have considerably fewer bad units. According to SpikeInterface, combining different spike-sorting techniques can help identify and eliminate FP units, which are the main contributors to bad units.

Therefore, we introduce the combination of Kilosort + IronClust in our study, demonstrating that the number of bad units can be reduced by more than 10x.

As shown in Fig. 6, the five sorters exhibit significant variation in performance on the same dataset. Therefore, we included all sorters in this study to better demonstrate the overall impact of the neural recording front-end characteristics. Additionally, since the good-units and accuracy metrics generally follow the same trend, we selected accuracy as our benchmark because it is a normalized metric, which makes it more suitable for consistent comparisons.

III. Results and Analysis

In this section, we evaluate the performance of the 5 spike sorting algorithms (HS: HerdingSpikes, TDC: Tridesclous, SC: Spyking Circus 2, KS: Kilosort 3 and 4 and IC: IronClust) as well as the combination method (KS+IC) across various specifications of the neural recording front-end. Our analysis covers: i) the orders and cutoff frequencies of the high-pass and low-pass filters; ii) the ADC sampling frequency and the associated quantization noise; and iii) the nonlinearities induced by the ADC and the amplifiers, which include harmonic distortion, missing-code errors and sticky-code errors. The results are presented for both the MEArec (M) and the Paired-Boyden (B) datasets.

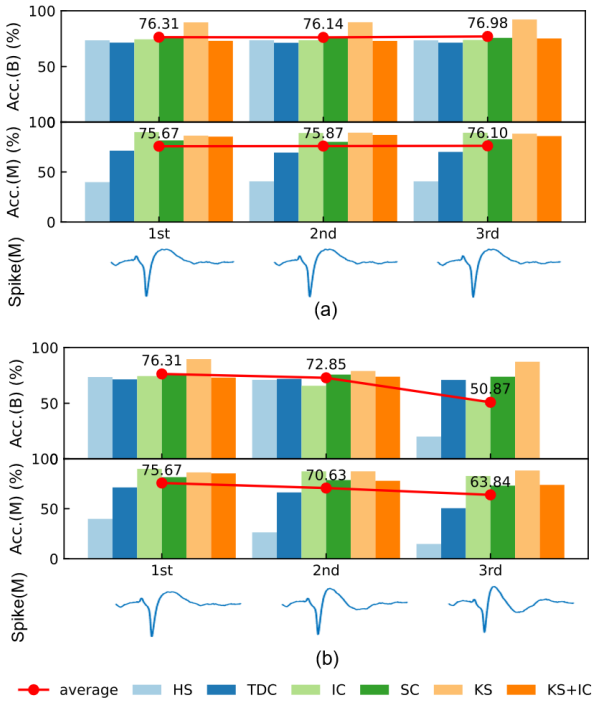

Fig. 7. (a) Comparison of the accuracy and spike waveforms across different low-pass filter orders, with a cutoff frequency of 15 kHz and a fixed 1st-order high-pass filter at 300 Hz. (b) Comparison of the accuracy and spike waveforms across different high-pass filter orders, with a cutoff frequency of 300 Hz and a fixed 1st-order low-pass filter at 15 kHz.

A. Filter Order and Cutoff Frequency

The assessment of the filter order is conducted separately for the high-pass and low-pass filters. Fig. 7 (a) shows the results of the low-pass filter order test, with a fixed 1st-order high-pass filter set at a typical cutoff frequency of 300 Hz. Interestingly, the performance of the sorters shows minimal variation, with only 0.67% (Paired-Boyden) and 0.43% (MEArec) increases in average accuracy across sorters when going from 1st to 3rd order filtering. This suggests that the order of the low-pass filter does not significantly impact the sorting accuracy. Therefore, when the goal is to minimize the design complexity, a 1st-order filter is sufficient. Fig. 7 (b) shows the results of the high-pass filter order evaluation, with a fixed 1st-order low-pass filter set at 15 kHz. Surprisingly, the average accuracy across sorters declines by 25.44% (Paired-Boyden) and 11.83% (MEArec) as the filter order increases from 1st to 3rd. Upon analyzing the spike waveforms in the MEArec dataset, we arbitrarily selected the average waveform of one neuron for demonstration in Fig. 7. It is evident that higher high-pass filter orders effectively suppress the low-frequency noise, but they also introduce significant distortion to the spike waveforms [51]. This distortion is caused by the inherent group delay associated with these filters. As the filter order increases, so does the group delay, causing different frequency components to experience different delays and leading to waveform distortion.

To verify that the waveform distortion is caused by group delay, we conducted an experiment comparing causal filters (using MATLAB's 'filter' function) and non-causal filters (using MATLAB's 'filtfilt' function). The non-causal filter, also known as the zero-group-delay filter, processes data by applying the filter twice: once from the beginning to the end and then from the end to the beginning, effectively nullifying any group delay. It is important to note that such a filter can only be implemented with even orders due to its bidirectional nature. Although a non-causal filter is impractical for hardware implementation because it requires bidirectional filtering, we use it in this experiment to isolate the impact of the group delay. We compare a 2nd-order bandpass causal filter with a 2nd-order bandpass non-causal filter, both ranging from 300 Hz to 15 kHz. We maintain a constant magnitude response, varying only the phase response and the group delay. Fig. 8 shows that the waveform after the non-causal filter resembles the raw spike more closely, and the accuracy using the non-causal filter exceeds that of the causal filter by 0.64% (Paired-Boyden) and 5.13% (MEArec). This confirms that the group delay significantly contributes to the decline in sorter performance with higher filter orders.

We have also conducted tests on the filter cutoff frequency. Fig. 9 shows the average accuracy of all sorters across various high-pass filter orders and cutoff frequencies. The average accuracy initially increases but then begins to decline around 1000 Hz for the MEArec dataset. In contrast, for the Paired-Boyden dataset, the accuracy increases by around 1% from 100 Hz to 300 Hz, remains relatively stable, and then drops more abruptly around 700 Hz. This trend can be attributed to the fact that both the spikes and the background far-neuron noise typically follow a 1/f spectrum. Thus, setting a high-pass filter at a higher frequency effectively filters out

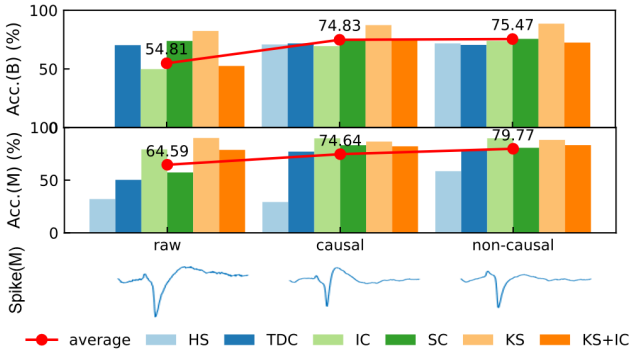

Fig. 8. Comparison of the accuracy and spike waveforms with no filter (raw), 2nd-order bandpass causal filter (300 Hz – 15 kHz), and 2nd-order non-causal filter (300 Hz – 15 kHz).

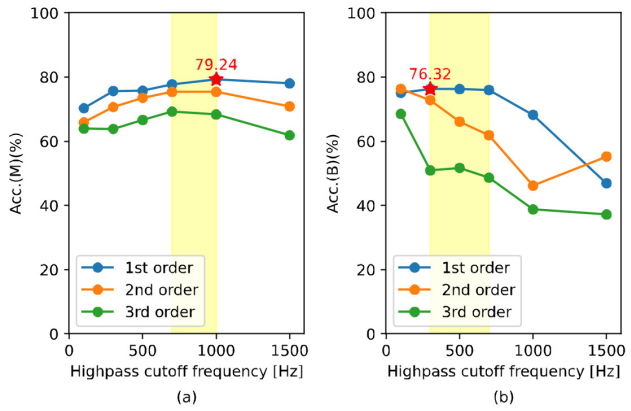

Fig. 9. Average accuracy across different high-pass-filter orders and cutoff frequencies for (a) the MEArec dataset and (b) the Paired-Boyden dataset, using a fixed 1st-order low-pass filter at 15 kHz.

significant portions of both the signal and the noise. As the filter order increases, the average accuracy decreases, and the optimal cutoff frequency shifts to lower values. This occurs because higher-order filters provide more noise suppression around the cutoff frequency. For cutoff frequencies higher than the optimal frequency, the spike amplitude starts to be attenuated by the filters.

The results of these experiments highlight the expected trade-off between noise reduction and signal preservation, which is crucial for achieving optimal spike-detection accuracy. In conclusion, the best high-pass filter setting to achieve high average accuracy in both datasets is a 1st-order filter at around 700 Hz. Finally, it is interesting to observe that Kilosort consistently performs the best and demonstrates less sensitivity to variations in the filter order and cutoff frequency.

B. ADC Sampling Frequency

Fig. 10 (top) compares the accuracy of the different sorters across varying ADC sampling frequencies. The MEArec dataset, being synthetic, allows for adjustable sampling frequencies. The Paired-Boyden dataset, however, has a fixed sampling rate of 30 kHz, thus tests at higher frequencies are not possible. As previously mentioned, the low-pass filter cutoff is set to the Nyquist frequency, so it varies together with the sampling frequency. As shown in the figure, the average

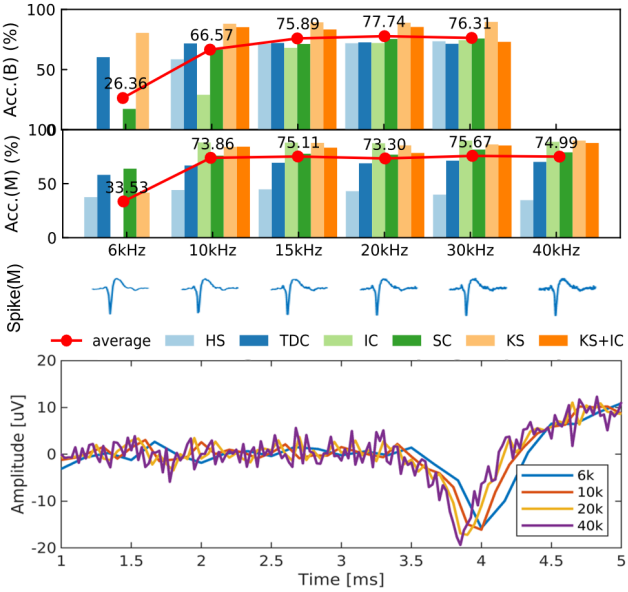

Fig. 10. Comparison of the accuracy and spike waveforms across different ADC sampling frequencies (top) and dataset snippets for the MEArec dataset (bottom). The 1st-order low-pass-filter cutoff frequency scales with the sampling frequency (Nyquist). A 1st-order 300-Hz high-pass filter is used.

accuracy increases sharply when going from 6 kHz to 10 kHz, with an improvement of approximately 40% for both datasets. Beyond this point, the rate of improvement slows down, and after 15 kHz, the accuracy appears to level off. This pattern suggests that a sampling frequency of 15 kHz, which yields accuracies of 75.89% (Paired-Boyden) and 75.11% (MEArec), should be adequate for spike sorting. Notably, at sampling frequencies of 10 kHz and above, Kilosort (for Paired-Boyden and MEArec) and IronClust (for MEArec only) consistently outperform other sorting algorithms. However, at lower sampling frequencies, such as 6 kHz, Tridesclous exhibits a higher accuracy, indicating its lower sensitivity to variations in the sampling frequency. It is also important to note that IronClust does not function properly at a 6-kHz sampling frequency, which is also the reason why the combination KS+IC cannot be simulated at this frequency.

The MEArec spike waveforms at the different sampling frequencies are also shown in Fig. 10 (bottom). At higher sampling rates, the spike waveforms are more detailed, but also include an increased amount of high-frequency noise. This explains why the accuracy plateaus or even begins to decline, as observed with HerdingSpikes, for sampling frequencies beyond 15 kHz.

C. ADC Resolution

A comparison of the accuracy and the spike waveforms across different ADC resolutions is shown in Fig. 11. The Paired-Boyden dataset shows sorting accuracy saturation from a resolution of 6 bits, while the MEArec dataset saturates after 8 bits. For the Paired-Boyden dataset, all sorters maintain a stable accuracy, resulting in $<1\%$ change when going from 6 bits to float point. In contrast, the MEArec dataset is more sensitive to the ADC resolution. In this case, the accuracy

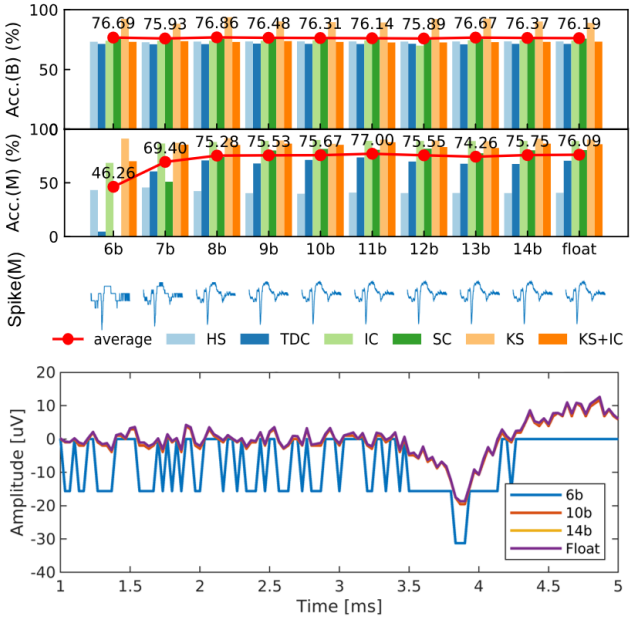

Fig. 11. Comparison of the accuracy and spike waveforms across different ADC resolutions (top) and dataset snippets for the MEArec dataset (bottom). Fixed 1st-order low-pass (15 kHz) and high-pass (300 Hz) filters are used.

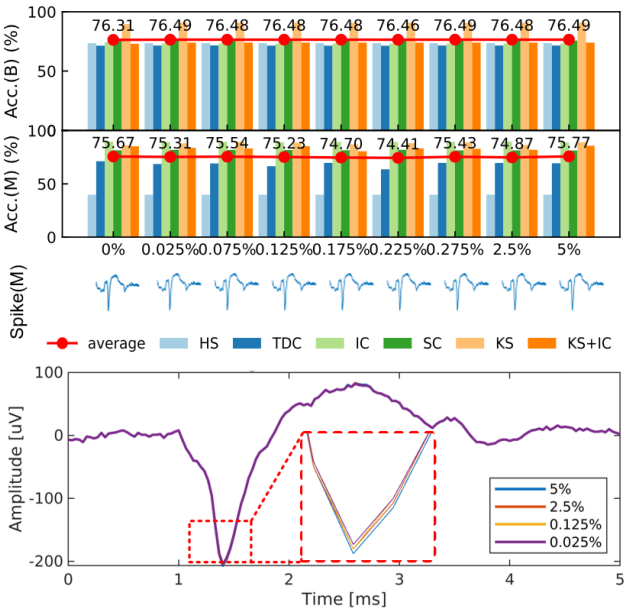

Fig. 12. Comparison of the accuracy and spike waveforms across different 3rd-order harmonic distortion percentages using a 10-bit ADC (top) and dataset snippets for the MEArec dataset (bottom).

increases by 23% when going from 6 to 7 bits, followed by a 6% gain from 7 to 8 bits, after which it levels off. At 6 bits, Spyking Circus fails, and Tridesclous achieves nearly 0% accuracy, indicating their high sensitivity to quantization errors. Kilosort, however, maintains the highest accuracy and shows robustness to reduced resolution.

Although the overall trend is similar for both datasets, the optimal ADC resolution differs due to variations in SNR, as discussed in Section II-B. The MEArec dataset has a lower average SNR and a broader SNR distribution, making it more susceptible to quantization noise. This likely explains why a higher resolution of 8 bits is optimal for the MEArec dataset, compared to the 6 bits of the Paired-Boyden dataset. The analysis of the spike waveforms in Fig. 11 shows that the signals maintain their details at resolutions above 8 bits for the MEArec dataset.

D. Nonlinearity Analysis

The accuracy results and spike waveforms obtained with different percentages of 3rd-order harmonic distortion are shown in Fig. 12. Remarkably, for both datasets, the accuracy of all sorters remains consistent, with less than 2% variation, even when subjected to 5% of HD_3. Additionally, the spike waveforms exhibit minimal variation, and only a slight difference is noticeable in the spikes with maximum amplitude. These results indicate that harmonic distortion does not significantly impact the performance of spike sorting.

The comparison of the accuracy and spike waveforms across different ADC sticky-code errors is shown in Fig. 13. The results indicate that, in both datasets, errors of up to 6 sticky codes do not significantly reduce the accuracy. This is because errors occurring at the mid code (0 V) blend into the noise floor, causing no significant information loss, while errors

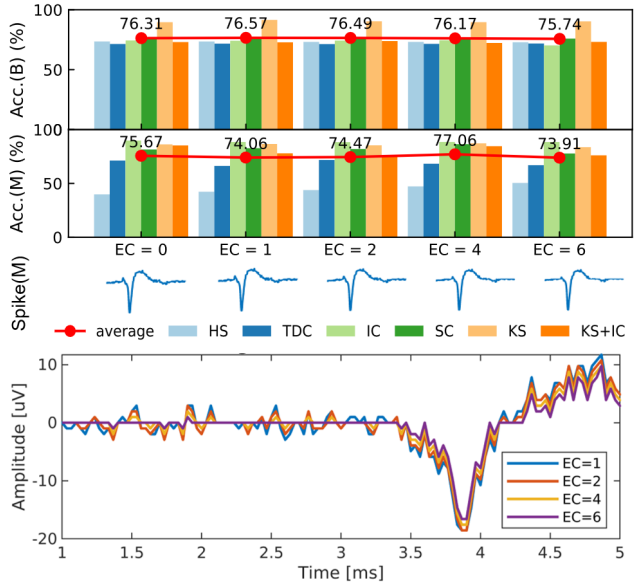

Fig. 13. Comparison of the accuracy and spike waveforms across different sticky-code errors in a 10-bit ADC (top) and dataset snippets for the MEArec dataset (bottom).

at other transition points do not noticeably distort the spike waveforms or obscure any spike features.

Fig. 14 presents a comparison of the accuracy and spike waveforms under different ADC missing-code error conditions. Interestingly, the results with the Paired-Boyden and the MEArec datasets show different trends. With the MEArec dataset, missing codes have a significant impact on the accuracy of the sorters. When there are no missing codes, the average accuracy is 75.67%. This accuracy decreases to 74.34% with just one missing code and drops further to 63.84% when there are 6 missing codes. Fig. 14 shows that,

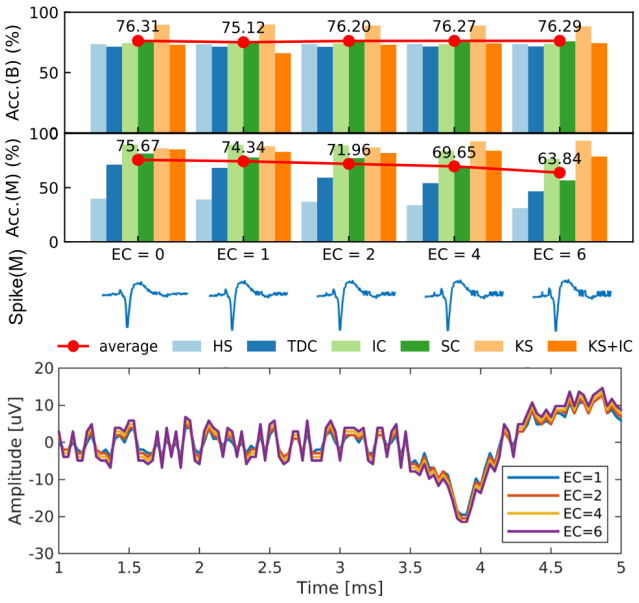

Fig. 14. Comparison of the accuracy and spike waveforms across different missing-code errors in a 10-bit ADC (top) and dataset snippets for the MEArec dataset (bottom).

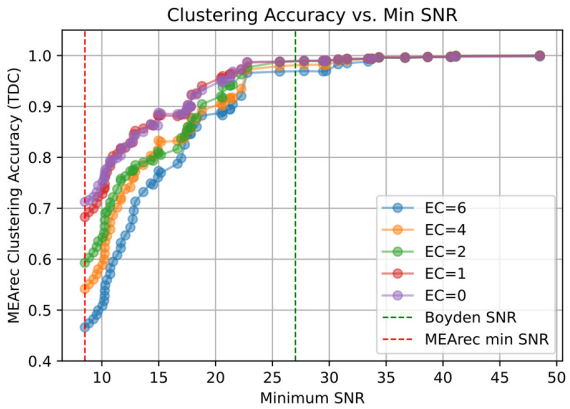

Fig. 15. Clustering accuracy of Tridesclous on the MEArec dataset as a function of minimum SNR, evaluated under varying numbers of ADC missing codes.

as the number of missing codes increases, there is a significant rise in the noise levels of the spike waveforms in the MEArec dataset. Notably, while most sorters are affected by this issue, Kilosort3 remains unaffected by missing codes. In contrast, the Paired-Boyden dataset shows less than a 2% change in accuracy from 0 to 6 missing codes, indicating minimal sensitivity to these errors.

The difference between the two datasets is primarily due to their SNR variations. This is illustrated in Fig. 15, which shows the relationship between the clustering accuracy and the minimum SNR for the MEArec dataset under different numbers of ADC missing codes (EC = 0, 1, 2, 4, 6), where EC = 0 represents no errors. As all sorters except KS follow a similar trend, the TDC accuracy is shown as a representative example. As mentioned earlier, the minimum SNR of the MEArec dataset is 8.5 (left red dashed line). Below an SNR of 15, the accuracy declines significantly, particularly for higher EC values. As the SNR increases, the accuracy improves, and the differences between the EC levels become less pronounced. Above an SNR of 30, missing codes have little to no impact. The green dashed line represents the SNR of the Paired-Boyden dataset (27), where the influence of missing codes is minimal.

TABLE III
RECOMMENDED SPECIFICATIONS FOR NEURAL-RECORDING FRONT-ENDS

Circuit Component	Characteristic	Recommended specification
Amplifier	Harmonic Distortion	$\leqslant$5%
Filter	High-pass order	1st order
	High-pass cutoff	700 Hz
	Low-pass order	1st order
	Low-pass cutoff	$\frac{1}{2}$ Sampling Frequency
ADC	Sampling Frequency	15 kHz
	Resolution	8 bits
	Sticky-code error	$\leqslant$6 codes
	Missing-code error	0 codes

E. Summary

Based on the results described in Section III-A-III-D, we can derive the optimal specifications of a neural recording front-end designed to achieve both high power and area efficiencies, as well as high spike-sorting accuracy. These optimal specifications are summarized in Table III.

IV. CONCLUSION

This paper has analyzed the critical interaction between the electrical performance specifications of the neural recording front-end and the performance of established spike-sorting algorithms. By analyzing relevant synthetic and real datasets with various state-of-the-art sorting algorithms, this paper has thoroughly examined the requirements for the amplifier, filter and ADC design, including the filter order, cutoff frequency, ADC resolution, ADC sampling frequency, and nonlinearity. Although some spike-sorting algorithms exhibit a low accuracy, our study demonstrates that front-end design parameters still have a significant impact on their performance. In contrast, high-accuracy algorithms like Kilosort show a greater robustness, displaying a reduced sensitivity to front-end imperfections.

Our findings indicate that optimal sorting performance can already be achieved with a 1st-order Butterworth bandpass filter with a band from 700 Hz to half the sampling rate (Nyquist frequency). Regarding the ADC design, a minimum sampling frequency of 15 kHz, a resolution of 8 bits, and zero missing codes ensure a stable sorting performance, while sticky-code errors of up to 6 codes can be tolerated. Additionally, it has been found that harmonic distortion up to 5% does not impact the spike sorting accuracy. These insights offer a comprehensive understanding of the minimum building-block requirements essential for developing power-efficient and compact neural recording front-ends tailored for accurate spike-sorting analysis.

ACKNOWLEDGMENT

The authors would like to thank Hua-Peng Liaw for his meticulous and comprehensive technical support in biomedical data analysis, Bernardo Tacca for his valuable suggestions about the signal processing, and Alessio Buccino and other developers for their support and timely updates on the SpikeInterface and MEArec platforms.

REFERENCES

[1] X. Yang et al., "An AC-coupled 1st-order Δ-ΔΣ readout IC for area-efficient neural signal acquisition," *IEEE J. Solid-State Circuits*, vol. 58, no. 4, pp. 949–960, Apr. 2023, doi: 10.1109/JSSC.2023.3234612.

[2] K. Sahasrabuddhe et al., "The Argo: A high channel count recording system for neural recording in vivo," *J. Neural Eng.*, vol. 18, no. 1, Feb. 2021, Art. no. 015002, doi: 10.1088/1741-2552/abd0ce.

[3] E. Musk and Neuralink, "An integrated brain-machine interface platform with thousands of channels," *J. Med. Internet Res.*, vol. 21, no. 10, Oct. 2019, Art. no. e16194, doi: 10.2196/16194.

[4] S. Wang et al., "A compact chopper stabilized Δ-ΔΣ neural readout IC with input impedance boosting," *IEEE Open J. Solid-State Circuits Soc.*, vol. 1, pp. 67–78, 2021, doi: 10.1109/OJSSCS.2021.3113887.

[5] C. Mora Lopez et al., "A neural probe with up to 966 electrodes and up to 384 configurable channels in 0.13 μm SOI CMOS," *IEEE Trans. Biomed. Circuits Syst.*, vol. 11, no. 3, pp. 510–522, Jun. 2017, doi: 10.1109/TBCAS.2016.2646901.

[6] G. N. Angotzi et al., "SiNAPS: An implantable active pixel sensor CMOS-probe for simultaneous large-scale neural recordings," *Biosensors Bioelectron.*, vol. 126, pp. 355–364, Feb. 2019, doi: 10.1016/j.bios.2018.10.032.

[7] X. Yang et al., "A highly-integrated 1536-channel quad-shank monolithic neural probe in 55nm CMOS for full-band raw-signal recording," in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2024, pp. 1–2, doi: 10.1109/VLSITechnologyandCir46783.2024.10631430.

[8] D. Yoon, S. Pinto, S. Chung, P. Merolla, T. Koh, and D. Seo, "A 1024-channel simultaneous recording neural SoC with stimulation and real-time spike detection," in *Proc. Symp. VLSI Circuits*, Jun. 2021, pp. 1–2, doi: 10.23919/VLSICircuits52068.2021.9492480.

[9] D. De Dorigo et al., "Fully immersible subcortical neural probes with modular architecture and a delta-sigma ADC integrated under each electrode for parallel readout of 144 recording sites," *IEEE J. Solid-State Circuits*, vol. 53, no. 11, pp. 3111–3125, Nov. 2018, doi: 10.1109/JSSC.2018.2873180.

[10] S. Wang et al., "A compact quad-shank CMOS neural probe with 5,120 addressable recording sites and 384 fully differential parallel channels," *IEEE Trans. Biomed. Circuits Syst.*, vol. 13, no. 6, pp. 1625–1634, Dec. 2019, doi: 10.1109/TBCAS.2019.2942450.

[11] P. Yger et al., "A spike sorting toolbox for up to thousands of electrodes validated with ground truth recordings in vitro and in vivo," *eLife*, vol. 7, Mar. 2018, Art. no. e34518, doi: 10.7554/eLife.34518.

[12] M. Pachitariu, S. Sridhar, and C. Stringer, "Solving the spike sorting problem with kilosort," *BioRxiv*, Jan. 2023, Art. no. 523036, doi: 10.1101/2023.01.07.523036.

[13] J. J. Jun, J. F. Magland, C. Mitelut, and A. H. Barnett, "IronClust: Scalable and drift-resistant spike sorting for long-duration, high-channel count recordings," *Elife*, vol. 9, Jun. 2020, Art. no. e61834.

[14] G. Hilgen et al., "Unsupervised spike sorting for large-scale, high-density multielectrode arrays," *Cell Rep.*, vol. 18, no. 10, pp. 2521–2532, Mar. 2017, doi: 10.1016/j.celrep.2017.02.038.

[15] R. Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural Comput.*, vol. 16, no. 8, pp. 1661–1687, Aug. 2004, doi: 10.1162/089976604774201631.

[16] Y. Chen et al., "A 384-channel online-spike-sorting IC using unsupervised geo-OSort clustering and achieving 0.0013 mm^2/Ch and 1.78μW/ch," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2023, pp. 486–488, doi: 10.1109/ISSCC42615.2023.10067264.

[17] F. Kalantari, H. Hosseini-Nejad, and A. M. Sodagar, "Hardware-efficient, on-the-fly, on-implant spike sorter dedicated to brain-implantable microsystems," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 30, no. 8, pp. 1098–1106, Aug. 2022, doi: 10.1109/TVLSI.2022.3170596.

[18] D. Valencia and A. Alimohammad, "Neural spike sorting using binarized neural networks," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 206–214, 2021, doi: 10.1109/TNSRE.2020.3043403.

[19] H. Hao, J. Chen, A. G. Richardson, J. Van der Spiegel, and F. Aflatouni, "A 10.8 μW neural signal recorder and processor with unsupervised analog classifier for spike sorting," *IEEE Trans. Biomed. Circuits Syst.*, vol. 15, no. 2, pp. 351–364, Apr. 2021, doi: 10.1109/TBCAS.2021.3076147.

[20] C. Seong, W. Lee, and D. Jeon, "A multi-channel spike sorting processor with accurate clustering algorithm using convolutional autoencoder," *IEEE Trans. Biomed. Circuits Syst.*, vol. 15, no. 6, pp. 1441–1453, Dec. 2021, doi: 10.1109/TBCAS.2021.3134660.

[21] S. M. A. Zeinolabedin et al., "A 16-channel fully configurable neural SoC with 1.52 μW/Ch signal acquisition, 2.79 μW/Ch real-time spike classifier, and 1.79 TOPS/W deep neural network accelerator in 22 nm FDSOI," *IEEE Trans. Biomed. Circuits Syst.*, vol. 16, no. 1, pp. 94–107, Feb. 2022, doi: 10.1109/TBCAS.2022.3142987.

[22] A. T. Do, S. M. A. Zeinolabedin, D. Jeon, D. Sylvester, and T. T. Kim, "An area-efficient 128-channel spike sorting processor for real-time neural recording with 0.175 μ W/channel in 65-nm CMOS," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 1, pp. 126–137, Jan. 2019, doi: 10.1109/TVLSI.2018.2875934.

[23] L. Schäffer, Z. Nagy, Z. Kincses, R. Fiáth, and I. Ulbert, "Spatial information based OSort for real-time spike sorting using FPGA," *IEEE Trans. Biomed. Eng.*, vol. 68, no. 1, pp. 99–108, Jan. 2021, doi: 10.1109/TBME.2020.2996281.

[24] V. Karkare, S. Gibson, and D. Markovic, "A 75-μW, 16-channel neural spike-sorting processor with unsupervised clustering," *IEEE J. Solid-State Circuits*, vol. 48, no. 9, pp. 2230–2238, Sep. 2013, doi: 10.1109/JSSC.2013.2264616.

[25] G. Regalia, S. Coelli, E. Biffi, G. Ferrigno, and A. Pedrocchi, "A framework for the comparative assessment of neuronal spike sorting algorithms towards more accurate off-line and on-line microelectrode arrays data analysis," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–19, Mar. 2016, doi: 10.1155/2016/8416237.

[26] X. Guo, M. Shaeri, and M. Shoaran, "An accurate and hardware-efficient dual spike detector for implantable neural interfaces," in *Proc. IEEE Biomed. Circuits Syst. Conf. (BioCAS)*, Oct. 2022, pp. 70–74, doi: 10.1109/BioCAS54905.2022.9948602.

[27] J. Navajas, D. Y. Barsakcioglu, A. Eftekhar, A. Jackson, T. G. Constandinou, and R. Quian Quiroga, "Minimum requirements for accurate and efficient real-time on-chip spike sorting," *J. Neurosci. Methods*, vol. 230, pp. 51–64, Jun. 2014, doi: 10.1016/j.jneumeth.2014.04.018.

[28] N. Even-Chen et al., "Power-saving design opportunities for wireless intracortical brain–computer interfaces," *Nature Biomed. Eng.*, vol. 4, no. 10, pp. 984–996, Aug. 2020, doi: 10.1038/s41551-020-0595-9.

[29] X. Jia et al., "High-density extracellular probes reveal dendritic backpropagation and facilitate neuron classification," *J Neurophysiol*, vol. 121, no. 5, pp. 1831–1847, Mar. 2019, doi: 10.1152/jn.00680.2018.

[30] M. S. Chae, W. Liu, and M. Sivaprakasam, "Design optimization for integrated neural recording systems," *IEEE J. Solid-State Circuits*, vol. 43, no. 9, pp. 1931–1939, Sep. 2008, doi: 10.1109/JSSC.2008.2001877.

[31] H.-S. Lee, K. Eom, M. Park, S.-B. Ku, K. Lee, and H.-M. Lee, "High-density neural recording system design," *Biomed. Eng. Lett.*, vol. 12, no. 3, pp. 251–261, Aug. 2022, doi: 10.1007/s13534-022-00233-z.

[32] F. Hashemi Noshahr, M. Nabavi, and M. Sawan, "Multi-channel neural recording implants: A review," *Sensors*, vol. 20, no. 3, p. 904, Feb. 2020, doi: 10.3390/s20030904.

[33] A. P. Buccino, S. Garcia, and P. Yger, "Spike sorting: New trends and challenges of the era of high-density probes," *Prog. Biomed. Eng.*, vol. 4, no. 2, Apr. 2022, Art. no. 022005, doi: 10.1088/2516-1091/ac6b96.

[34] A. P. Buccino and G. T. Einevoll, "MEArec: A fast and customizable testbench simulator for ground-truth extracellular spiking activity," *Neuroinformatics*, vol. 19, no. 1, pp. 185–204, Jan. 2021, doi: 10.1007/s12021-020-09467-7.

[35] A. P. Buccino et al., "SpikeInterface, a unified framework for spike sorting," *eLife*, vol. 9, pp. 1–24, Nov. 2020, doi: 10.7554/elife.61834.

[36] J. Magland et al., "SpikeForest, reproducible web-facing ground-truth validation of automated neural spike sorters," *eLife*, vol. 9, May 2020, Art. no. e55167, doi: 10.7554/eLife.55167.

[37] R. Müller, S. Gambini, and J. M. Rabaey, "A 0.013 mm^2 5μW DC-coupled neural signal acquisition IC with 0.5 V supply," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2011, pp. 302–304, doi: 10.1109/ISSCC.2011.5746328.

[38] W. Biederman et al., "A fully-integrated, miniaturized (0.125 mm^2) 10.5 μW wireless neural sensor," *IEEE J. Solid-State Circuits*, vol. 48, no. 4, pp. 960–970, Apr. 2013, doi: 10.1109/JSSC.2013.2238994.

[39] H. Kassiri et al., "Rail-to-rail-input dual-radio 64-channel closed-loop neurostimulator," *IEEE J. Solid-State Circuits*, vol. 52, no. 11, pp. 2793–2810, Nov. 2017, doi: 10.1109/JSSC.2017.2749426.

[40] J. Xu et al., "Fascicle-selective ultrasound-powered bidirectional wireless peripheral nerve interface IC," *IEEE Trans. Biomed. Circuits Syst.*, vol. 17, no. 6, pp. 1237–1256, Dec. 2023, doi: 10.1109/TBCAS.2023.3332258.

[41] W. M. Sansen, *Analog Design Essentials*, vol. 859. Cham, Switzerland: Springer, 2007.

[42] X. Zhu, "High performance SAR A/D converter with calibration techniques," Ph.D. dissertation, Sch. of Integrated Design Eng., Keio Univ., Yokohama, Japan, Sep. 2012.

[43] J. Du, T. J. Blanche, R. R. Harrison, H. A. Lester, and S. C. Masmanidis, "Multiplexed, high density electrophysiology with nanofabricated neural probes," *PLoS ONE*, vol. 6, no. 10, Oct. 2011, Art. no. e26204, doi: 10.1371/journal.pone.0026204.

[44] C. M. Lopez et al., "An implantable 455-active-electrode 52-channel CMOS neural probe," *IEEE J. Solid-State Circuits*, vol. 49, no. 1, pp. 248–261, Jan. 2014, doi: 10.1109/JSSC.2013.2284347.

[45] B. C. Raducanu et al., "Time multiplexed active neural probe with 678 parallel recording sites," in *Proc. 46th Eur. Solid-State Device Res. Conf. (ESSDERC)*, Sep. 2016, pp. 385–388, doi: 10.1109/ESSDERC.2016.7599667.

[46] A. E. Mendrela et al., "A bidirectional neural interface circuit with active stimulation artifact cancellation and cross-channel common-mode noise suppression," *IEEE J. Solid-State Circuits*, vol. 51, no. 4, pp. 955–965, Apr. 2016, doi: 10.1109/JSSC.2015.2506651.

[47] K. A. Ludwig, R. M. Miriani, N. B. Langhals, M. D. Joseph, D. J. Anderson, and D. R. Kipke, "Using a common average reference to improve cortical neuron recordings from microelectrode arrays," *J. Neurophysiol.*, vol. 101, no. 3, pp. 1679–1689, Mar. 2009, doi: 10.1152/jn.90989.2008.

[48] *Tridesclous*. Accessed: Apr. 29, 2024. [Online]. Available: https://tridesclous.readthedocs.io/en/latest/

[49] M. Abeles and M. H. Goldstein, "Multispike train analysis," *Proc. IEEE*, vol. 65, no. 5, pp. 762–773, 1977, doi: 10.1109/PROC.1977.10559.

[50] P. Bhattacharjee and P. Mitra, "A survey of density based clustering algorithms," *Frontiers Comput. Sci.*, vol. 15, no. 1, Feb. 2021, Art. no. 151308, doi: 10.1007/s11704-019-9059-3.

[51] R. Pal, "Comparison of the design of FIR and IIR filters for a given specification and removal of phase distortion from IIR filters," in *Proc. Int. Conf. Adv. Comput., Commun. Control (ICAC3)*, Dec. 2017, pp. 1–3, doi: 10.1109/ICAC3.2017.8318772.