Validating spike sorting toolkits against ground-truth electrophysiological data

Jano Horváth

MInf Project (Part 2) Report

Master of Informatics
School of Informatics
University of Edinburgh

2018

Abstract

The past 10 years have seen a rapid development of neurophysiological technology, which now enables researchers to detect neural activity from thousands of distinct cells with very high resolution and frame rates. To analyse the large amounts of raw data these devices produce, so-called spike-sorting algorithms have been designed; algorithms which detect distinct events (spikes), localise them spatially, and sort them according to the cell of their origin. To evaluate the correctness of these algorithms, a ground-truth dataset needs to be obtained, which contains a well-validated recording of an activity of a single cell in the proximity of the device.

Such ground-truth recording by Neto et al. [2016] has been used in this project to validate a spike detection algorithm by Muthmann et al. [2015] and a spike sorting algorithm by Hilgen et al. [2017]. Bayesian Optimisation was introduced as a means to bypass human factor in parameter tuning of these algorithms. Using the optimal parameters, the detection algorithm performed a perfect detection. The sorting algorithm has been validated, and although a sub-optimal performance was observed, the cause of this imperfect performance was identified, validating the implementation even further. These promising results present a strong case for using Bayesian Optimisation as a standard framework to both bypass the human factor and validate correctness of performance in the domain of spike sorting.

Acknowledgements

I would like to thank my parents, Eva and Tomáš, for their unceasing encouragement, emotional support and fundamentally, for making this University journey possible at all. Without you, I would have never made it this far!

I would also like to thank Dr Matthias Hennig for his support and supervision throughout this year. None of the work on this project would happen if it weren't for his ongoing guidance and help.

Lastly, a special thanks goes to Robin "The Honeybear" Nelson for his unique perspective during our countless discussions and for encouragement during those long hours full of work.

Table of Contents

Chapter 1

Introduction

The ability *'to film what is going on inside my head'* has traditionally been attributed to the realm of science fiction cinematography (John Malkovich!). However, with the latest advancements in the field of electrophysiology, this concept is gradually becoming less fiction and increasingly more science! High-density micro-electrode array (MEA) chips based on the CMOS technology, commonly found in high-end video cameras, are the latest step in the rapid hardware evolution of electrophysiological probes. The purpose of these devices is to record weak electric activity between the cells of the nervous systems (i.e. record neurons extracellularly), either "in a Petri dish" using extracted tissue preparations (*in-vitro*) or directly inside a living organism (*in-vivo*). With the latest models of high-resolution MEA probes featuring 1000s of recording sites (also called channels) and each channel sampling voltage with rates greater than $10kHz$, the bandwidth of these recordings easily exceeds orders of GigaBytes per minute. Because of this, fast, scalable algorithms, which can extract meaningful information from these large streams of raw data, a task dubbed 'spike sorting', are needed.

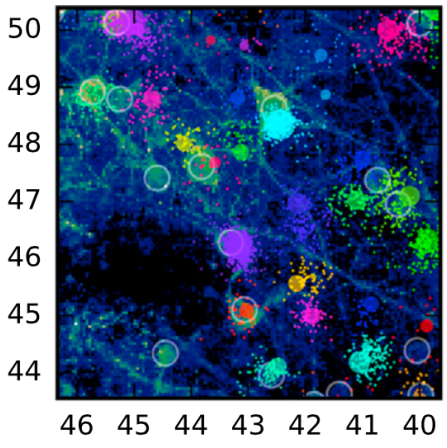

Figure 1.1: The output of a spike sorting algorithm overlaid on a confocal image of neural tissue (dark blue and green). The algorithm clusters and localises detected events (colourful dots). Notice the correspondence of cluster centres to the cell bodies (annotated with circles). Axes are in probe coordinates. [Figure by: Hilgen et al., 2017]

Due to the microscopic dimensions of the recording sites (μm), short timespans of observed events ($10^{-1} ms$) and generally high volumes of data, the evaluation of spike sorting algorithms in terms of correctness is a non-trivial task requiring insight into the electrical activity of the neural tissue. This poses a paradox: How can we validate tools which record neural activity if in order to do so, we need an insight into the very neural activity we are recording? A so-called ground-truth paired recording has to be obtained, which simultaneously records the activity of a single neural cell in the proximity of the MEA using a different, well-validated technique. Paired recordings have proven invaluable for validation of the previous generation of low-resolution electrophysiological probes [e.g. Wehr et al., 1999, Harris et al., 2000], but in the 15 years since the invention of MEA, the first and **only** paired ground-truth recording has been produced only months before the start of this project [Neto et al., 2016] (discussed in detail in Section 2.4). For the first time, it is therefore possible to validate spike sorting algorithms for *in-vivo* MEA data automatically.

In light of this recent development, the objectives of this project are two-fold:

- First, to adapt a set of algorithms designed by Muthmann et al. [2015] and Hilgen et al. [2017] so that they can be used with the paired data by Neto et al. [2016].
- Second, to design a framework for evaluating the correctness of spike-sorting algorithms and then use it to validate the aforementioned algorithms using the ground-truth paired recordings.

1.1 Work carried out in this project

The focus of Part 1 of the project was to adapt the existing algorithms to the ground-truth data. The following contributions have been made last academic year:

- Created tools to reformat paired ground-truth datasets to industry standards, increasing memory efficiency of subsequent data reads.
- Modified existing detection algorithms on a theoretical level and then implemented the changes in the existing C++11 code.
- Attempted to validate the modified algorithms against the paired ground-truth datasets.

The pipeline of the spike sorting toolkit was ready for validation, however, I could not relate the ground-truth recording to the MEA probe data. Upon manual inspection, it transpired that some neural activity is simply not recorded by the MEA. I have proposed a few hypotheses for this condition, but was not able to prove any of them conclusively. In a particularly passionate moment of the Discussion chapter of my last year's report, I therefore exclaimed: *'Is it then even possible to achieve satisfactory performance, where the algorithm would recall all (> 99.5%)* [neural activity]*?'* [Horváth, 2017]

The focus of Part 2 of the project, carried out this year, was to find an answer to that question. The original scope of the project was only concerned with implementing a solution, where after choosing suitable parameters for the detection and sorting algorithms, it could be shown that all ground-truth events are detected and sorted correctly. This has been implemented successfully. However the success of this method relied heavily on manual parameter tuning, and was hence reliant on the experience of the researcher (and often a bit of luck). To mitigate the human factor, I have introduced the idea of Bayesian Parameter Optimisation and integrated it with the algorithms. Therefore, not only the correctness of the algorithms can now be determined objectively, also the optimal configuration of parameters for that particular set of data can be obtained automatically.

In summary, the following contributions have been made this academic year:

- Performed a detailed analysis of the paired datasets by Neto et al. [2016] trying to address the challenges of Part 1 of this project.
- Designed a theoretical framework for validation and parameter optimisation of the spike sorting algorithms.
- Implemented multi-threaded evaluation methods for the detection algorithm in Python & implemented the parameter optimisation framework using the `scikit-optimize` Python library.
- Validated both the detection algorithm by Muthmann et al. [2015] and the sorting algorithm by Hilgen et al. [2017].

Chapter 2

Background

2.1 The field of Electrophysiology

To thoroughly record the inner workings of the human brain has always been the holy grail of electrophysiology. Since its inception, 150 years ago [Bernstein, 1868], electrophysiology closely followed the evolution of microelectronics, always finding a use for the latest, smallest circuitry. The 1950s saw the first attempts of an *in-vivo* recording of an arbitrary neural brain cell using a single microelectrode [Dowben and Rose, 1953]. Then, the precision of recording increased dramatically when Sakmann and Neher [1984] introduced their Nobel-prize winning patch clamp technique for recording individual ionic channels of a cell *in-vitro*. Next came the tetrode, which combined four electrodes approximately $30\mu m$ in diameter into one probe and was able to record multiple cells extracellularly by detecting voltage in the space between the cells (i.e. field potential in the synapse) [Recce, 1989]. Having been thoroughly validated [e.g. Wehr et al., 1999, Henze et al., 2000], the tetrode has become the golden standard in electrophysiology in the late 1990s and is being used in almost every neurophysiological laboratory around the world to this day.

The invention of the microelectrode array (MEA), the latest step in this rapid hardware evolution, has triggered an explosive growth in the number of recording sites. Using the complementary metal-oxid semiconductor (CMOS) technology, commonly found in high-end cameras, even the very first designs were able to record from hundreds of sites simultaneously [Eversmann et al., 2003]. *In-vitro*, this number quickly grew to thousands of electrodes at sampling rate $7kHz$ [Berdondini et al., 2005], and continues to grow both in terms of the sampling rates: $18kHz$ on 4K active electrodes [Ruther and Paul, 2015] and in terms of the numbers of electrodes: 60K multiplexed electrodes at $10kHz$ [Dragas et al., 2017].

The *in-vivo* probes follow this development with the latest designs by the European NeuroSeeker research group comprising of 1.4K electrodes with 7.5kHz sampling rate [Raducanu et al., 2017], and the US counterpart, the NeuroPixel probe, with 1K electrodes and a thinner diameter [Jun et al., 2017]. Although the CMOS technology has shrunk the circuitry significantly, allowing *in-vivo* probes to include the amplification,

digitisation, and multiplexing on the probe itself, it is still the case that one output connection is required for each recording site. This poses a significant challenge for the design of *in-vivo* probes which, contrary to their *in-vitro* counterparts, have to minimise tissue damage of the surrounding cells, which constrains the dimensions and prohibits the scaling of these devices (Figure 2.1). A potential solution to scaling of *in-vivo* probes is currently under discussion. One approach, suggested by the NeuroSeeker team in a preprint released less then a month before the submission of this project, is to not record from every electrode, but dynamically multiplex which electrodes are recorded from instead [Unpublished Dimitriadis et al., 2018].

Looking into the future, some claim the number of recorded cells will keep growing exponentially for at least the next two decades [Stevenson and Kording, 2011], and some are more cautious, e.g. Marblestone et al. [2013] who in their sobering review remind of the limits of the elementary physics we are getting close to with the current designs. However, both the optimists and the sceptics agree that the holy grail, to thoroughly record the inner workings of the human brain, is not even in reach.

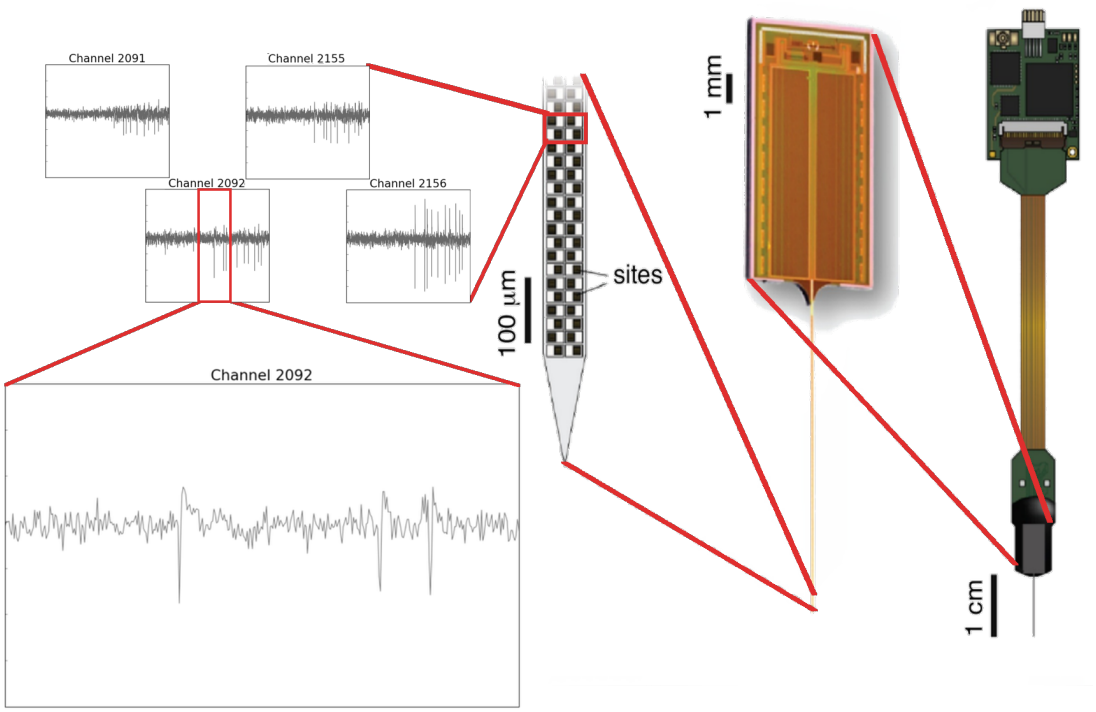

Figure 2.1: A detailed look at the *in-vivo* NeuroPixel probe. Right to left: The headstage with a flex cable and a PC interface. The CMOS element containing circuits for amplification, digitisation and multiplexing. A high-level schema of the tip of the shank with individual recording sites. Signal sampled at four neighbouring channels. A detailed examination of signal from a single channel showing three distinct spiking events. Images of hardware components by Steinmetz et al. [2018]

2.2 Spike sorting toolkits

Sampling from even modest MEA devices, e.g. $\sim$ 1K channels, with 12-bit resolution, at $\sim 10kHz$ already results in bandwidths in orders of GigaBytes per minute. A number which is only going to increase as the hardware improves. This creates a need for highly scalable software that would be capable of handling and extracting meaningful information from such volumes of data. The relevant information is encoded in a form of sudden bursts (spikes) in voltage level (Figure 2.1) which correspond to a neuron transmitting a signal. From the biological standpoint, it has been shown that each neuron has a different voltage 'footprint': its spikes have a distinct shape compared to spikes produced by other neurons [Gold et al., 2006], and that sequences of these spikes can be propagated through whole networks of neurons in the brain [Diesmann et al., 1999]. The occurrence of these events is sparse: e.g. in the auditory cortex part of the brain, Hromádka et al. [2008] have observed that the rate of relevant information can be up to 1000 times smaller than the rate of recording.

The task is then to design a software, which will first **detect** the spike events and then attribute each event to a specific cell in a process called **sorting**. Without going into too much detail, the main approaches to spike sorting can be broadly categorised into two groups:

- **Template based**. Where the algorithm first extracts [Prentice et al., 2011], or learns [Pachitariu et al., 2016] templates of spike shapes and then scans the raw data, detects events, and assigns them to a particular cell using some similarity measures.

- **Threshold based.** In this approach, the algorithm detects deviations from the baseline signal values by (dynamically) setting a threshold value. Threshold crossings are then identified as events [e.g. Muthmann et al., 2015, Rossant et al., 2016, Chung et al., 2017]. The next step is to then cluster the events based on their spatial and waveform properties, where each cluster corresponds to a single cell. The variation between the methods comes in their approach to clustering, where some use well-established algorithms [Hilgen et al., 2017, Shoham et al., 2003], some invent their own [Quiroga et al., 2004, Chung et al., 2017] and some generate templates from the clusters and perform additional template based sorting [Yger et al., 2016]. Notably, Kim and Kim [2000] used the results of clustering to train a supervised (artificial) neural network classifier (radial basis function network) which could then sort spikes recorded by a single electrode with above average accuracy.

The sorted events are then fed into the last stage of the pipeline, the visualisation framework, which is then used by a neurobiologist in their research.

2.3 Validation of spike sorting toolkits

When choosing between the spike sorting algorithms, it might be desirable to benchmark them in terms of speed or accuracy. However to prove the correctness of a spike sorting algorithm, independently recorded data from the same area (i.e. ground-truth data) needs to be paired with the recording; a surprisingly challenging task. Figure 2.1 serves as a reminder that even to obtain one *in-vivo* MEA recording, we already have to operate on a microscopic scale, at high frequencies and with limited physical accessibility. Nevertheless, it can still be surprising that in the 15 years since the invention of MEA, the first and **only** paired ground-truth recording has been produced only very recently [Neto et al., 2016] (discussed in detail in Section 2.4). For the first time, it is therefore possible to validate spike sorting algorithms for *in-vivo* MEA data automatically.

A note on 'automation': The field of neurophysiology is in a peculiar state, where years after the introduction of the first MEA, "*most laboratories still rely on manual intervention* [for spike sorting]" [page 2. Chung et al., 2017], whether it is the extensive manual parameter tuning, or the semi-automatic sorting algorithms, where human intervention is part of the design of the sorting process (e.g. manually choosing which clusters to merge). This practice is prevalent even after it transpired that human performance in spike sorting is not consistent [Harris et al., 2000, Pedreira et al., 2012] or varies significantly depending on the person [Rossant et al., 2016]. The need for human intervention in spike sorting is introduced due to the lack of ground-truth data, as the accuracy of the algorithms cannot be properly benchmarked on real data, nor can these algorithms be properly optimised. However, in order to validate spike sorting algorithms using ground-truth data, it is essential to remove the human factor from the procedure to ensure correctness and reproducibility.

In the absence of paired ground-truth recording, algorithms are being validated against synthetic data. The data can be either derived analytically by modelling signal for each channel (e.g. triangle waveform with Gaussian white noise [Biffi et al., 2010]), by modelling the spiking activity of a whole network of neurons [Smith and Mtetwa, 2007], or by modelling individual cells in terms of their specific biological components using the Neuron simulation software [Maccione et al., 2009] and even injecting an intracellular recording into the modelled soma [Einevoll et al., 2012]. Alternatively, the analytically generated data is merged with a recording of an empty probe [Muthmann et al., 2015] or with the low-frequency fluctuations (LFP) of a recording, creating a 'hybrid dataset', a term coined by Rossant et al. [2016].

Alternatively, a consensus-based approach to validation has been proposed which does not require ground-truth data. First, the spike sorting algorithm is rerun multiple times with slight alternations between the runs. The spikes which always get sorted into the same cluster (consensus) are assumed to be sorted correctly and a probabilistic distribution is generated for the rest. This is then repeated multiple times to either validate the sorting algorithm itself by altering the data [Barnett et al., 2016] or to optimise the sorting performance by altering the clustering parameters [Fournier et al., 2016].

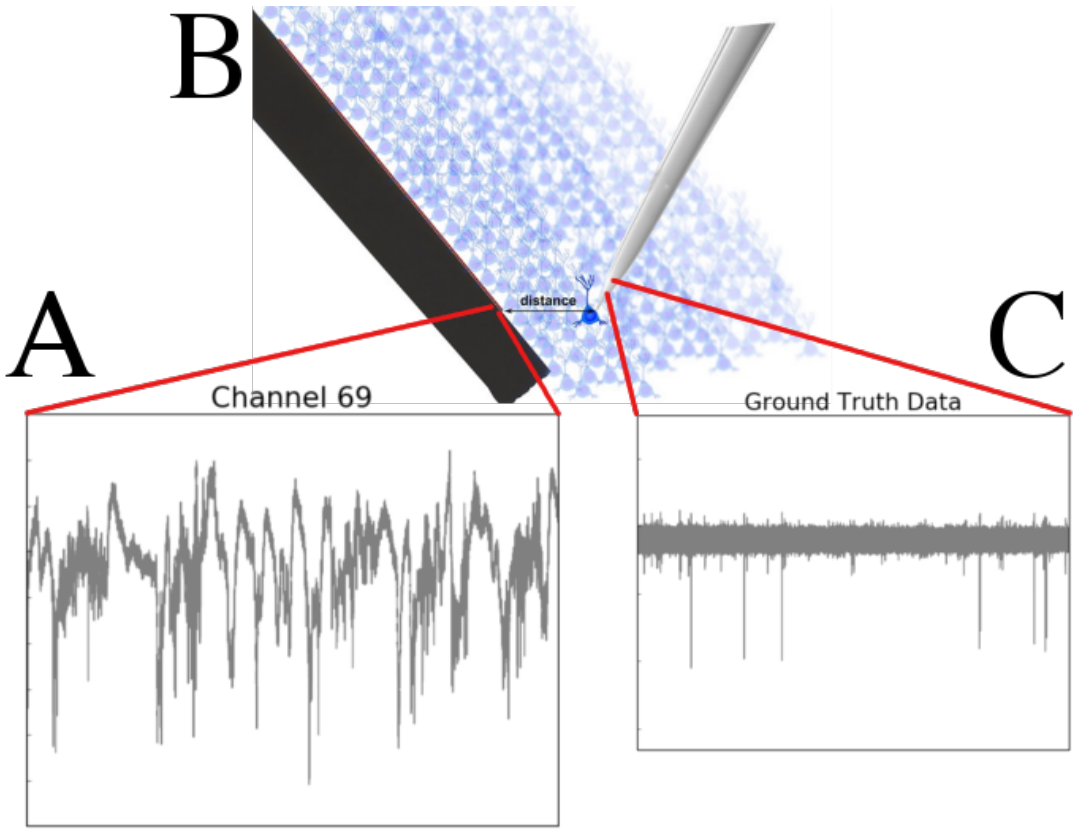

Figure 2.2: The setup designed by Neto et al. [2016]. A. Raw signal from the MEA probe with low-frequency fluctuations. B. The ground-truth pipette (grey) is recording from a single cell in the proximity of the MEA probe (black) C. Band-passed raw signal from the ground-truth electrode. [Figure 3B. by: Neto et al., 2016]

2.4 Data used in this project

To obtain the rare paired dataset, Neto et al. [2016] performed, with extreme precision, a blind insertion of both the MEA probe and a ground-truth pipette using high-accuracy mechanical manipulators (Figure 2.2 B). First, the MEA probe, a 128-channel early iteration of Neuroseeker [Pothof et al., 2015], is inserted into the hippocampus region of a mouse's brain. Each electrode on this probe is $20\mu m \times 20\mu m$ wide and the electrodes are arranged in a 4×32 layout. Next, the pipette is inserted blindly towards a probe until it reaches a nearby cell. It is then attached to the membrane of the cell using the loose-patch technique. Although the scale of this operation is microscopic, Neto et al. [2016] managed to position the pipette on average only $96\mu m$ away from the MEA ($n = 10, min = 29\mu m, max = 150\mu m$).

Raw signal from the MEA probe was sampled at $30kHz$ with 16-bit resolution, band-pass filtered in the frequency range $0.1 - 7500Hz$ (Figure 2.2 A). The same sampling was used for the pipette, but the recording was filtered in the frequency range $300 - 8000Hz$ (Figure 2.2C). The difference in the low-frequency band can be observed in Figure 2.2, as the significant low-frequency fluctuations present in A. are not present

in C.

The data was retrieved from the official repository (http://www.kampff-lab.org/validating-electrodes/). It was then repackaged from the original cumbersome format into the industry-standard HDF5 and fed into the spike detection and spike sorting algorithms described in detail in Chapter 3.

2.5 Distance of probe from the ground-truth pipette as a factor in recordings

At the end of Part 1 of this project, I had to present inconclusive results. The accuracy of the spike detection algorithm was poor, and manual inspection of the raw data revealed that some ground-truth spikes simply were not present in the MEA recording [Figure 4.3F, Horváth, 2017]. The hypothesis I proposed was that the distance of the MEA probe from the ground-truth electrode could be playing a role. The two datasets used had similar probe-pipette distance of 77.8μm and 78μm. Could this play a role? Could 78μm be too far away for the MEA to record significant signal extracellularly?

Some research suggests the answer is positive, with the maximum recording distance found to be 50μm [Henze et al., 2000], 60μm [Somogyvari et al., 2012], and 50μm *in-vitro* [Anastassiou et al., 2015], whereas others report recording from up to 100μm [Henze and Buzsáki, 2007] and even 140μm [Du et al., 2011]. To factor out any influence of the probe-pipette distance on performance, the recording with the lowest probe-pipette distance (29μm) was used in this year of the project.

Chapter 3

Methodology

3.1 Detection

To detect events from the raw voltage data, a thresholding algorithm with running baseline designed by Muthmann et al. [2015] was used (the non-interpolated variant). Frame-by-frame, this algorithm reads the voltage x_t at the frame t, updates the variable baseline values and then uses a threshold derived from the baseline values to determine whether an event has happened at t (Figure 3.1). Only the event with the highest amplitude from among neighbouring channels is kept. Lastly, the spatial coordinates are calculated for that event. This procedure is applied to all channels recorded in a frame, either in parallel or, as in this project, sequentially channel-by-channel.

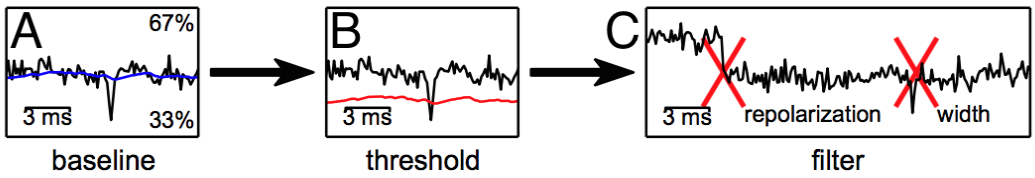

Figure 3.1: Three stages of the detection algorithm. A. Variable baseline value b is computed for every frame t B. An event is detected, when the voltage crosses a threshold value derived from the baseline b. C. Every threshold crossing is checked against strict criteria, to ensure biological plausibility of the detected events. Some candidates are rejected. [Figure by: Muthmann et al., 2015]

The variable baseline b is updated dynamically, using the variability estimate v from the previous frame. The variability estimate is itself updated depending on how much the signal amplitude x_t varies from the current variability estimate (Equation 3.1).

$$b_{t+1} = \begin{cases} b_t + \frac{v_t}{4} & x_t > b_t + v_t \\ b_t - \frac{v_t}{2} & x_t < b_t - v_t \\ b_t & otherwise \end{cases}$$
$$v_{t+1} = \begin{cases} v_t + 1 & x_t \in [b_t - v_t, b_t - 5v_t) \\ v_t - 1 & x_t \in [b_t, b_t - v_t) \\ v_t - 1 & x_t \in [b_t - 6v_t, \infty) \\ v_t & otherwise \end{cases} \tag{3.1}$$

After the baseline is updated, the threshold for event detection is set to $b_t - \theta v_t$, where θ is a global parameter (Figure 3.1 B). An event is recorded at time t if the following three criteria are fulfilled:

1. For the next τ_{event} steps there was no larger minimum than the current value

$$\forall t', t' \in (t, t + \tau_{event}) : x_{t'} \geq x_t \tag{3.2}$$

2. The spikes repolarises in the next τ_{event} steps (Figure 3.1 C).

$$\exists t', t' \in (t, t + \tau_{event}) : x_{t'} > \theta_b v_t \tag{3.3}$$

3. Sum of all baseline-subtracted voltages is larger than $-\theta_{ev}$ indicating the event is wider than $\tau_{ev} - 1$ frames (Figure 3.1 C).

$$-\theta_{ev} < \sum_{t'=t}^{t+\tau_{ev}} x_{t'} - b_{t'} \tag{3.4}$$

The last step of the detection phase is the localisation of a detected event. Acknowledging the findings of Pettersen et al. [2008] that the electrical potential in the soma decays approximately quadratically with distance, that is $V \sim \frac{1}{r^2}$ where r denotes distance, Muthmann et al. designed a procedure which can localise events with precision higher than the channel resolution. The exact spatial coordinates are calculated as a barycentric average using amplitudes of the neighbouring channels.

The detection phase outputs a two dimensional array of values, where every row represents one event detected, and columns represent respective event's timestamp, amplitude, its spatial location, the id of a channel on which it was detected, and its spike shape.

3.2 Sorting

The aim of a spike sorting toolkit is not only to detect events in the stream of raw data but to attribute these events to specific neural cells. At first glance, it might seem sufficient to group events together based only on the spatial localisation, however, Prentice

et al. [2011] present a strong case concluding that for spatial-only localisation on large arrays (such as the MEAs discussed in this report) "there will inevitably be temporal collisions of spikes from distinct units" [Text & Fig. 2A, p3. Prentice et al., 2011]. Similarly, it might seem sufficient to attribute events to cells purely on the basis of the shape of the spike since, as stated in Section 2.1, it has been observed that distinct neurons produce events with unique shapes. However, the authors of the algorithm used in this project have observed that particularly for channels which record only weaker signals from multiple remote cells, shape-only sorting often yields incorrect results [Hilgen et al., 2017]. Under these constraints, spike sorting then becomes a high-dimensional clustering problem taking both the spatial and waveform properties of events into account.

The first step of the spike sorting algorithm designed by Hilgen et al. [2017] is to extract d most dominant waveform features, using principal-component analysis (PCA), which reduces dimensionality of the data and therefore also computational complexity of the clustering algorithm. These waveform features are then weighted by a parameter α which gives the ability to prioritise waveform over spatial information (or vice-versa) in the event representation.

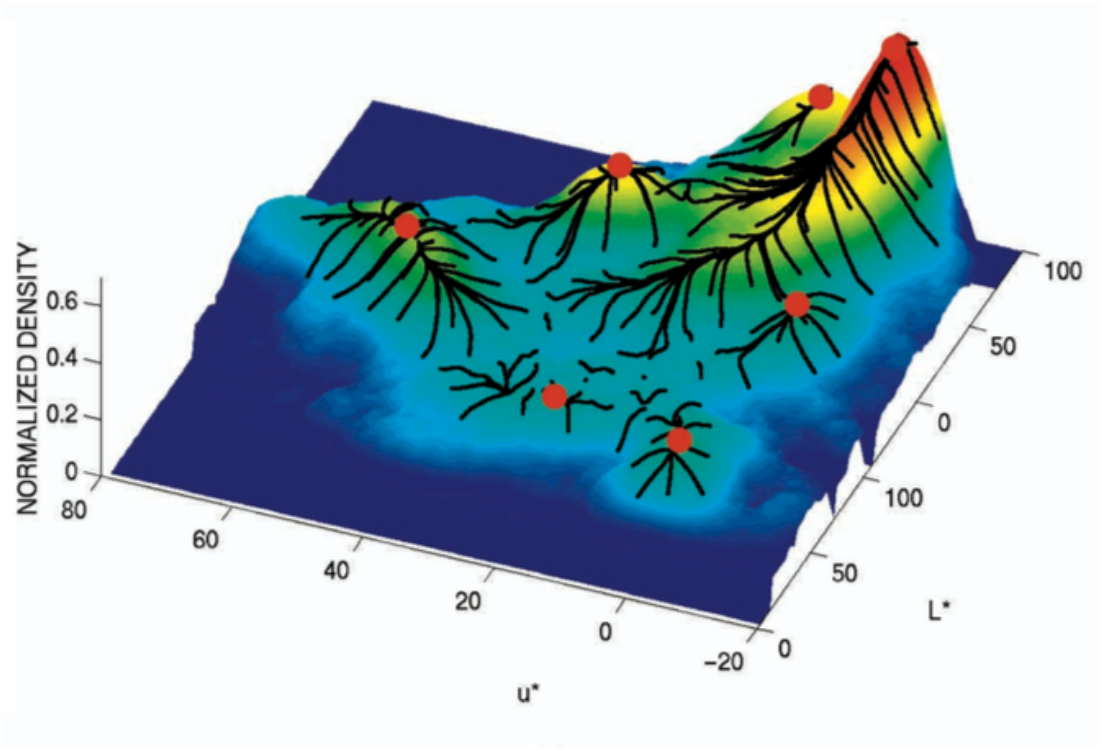

Figure 3.2: The trajectories of Mean Shift procedures (black lines) shown over a density estimate plot of a sample dataset. Red dots denote the final cluster centres. [Figure by: Comaniciu and Meer, 2002]

Lastly the events are clustered together using the MeanShift clustering algorithm [Comaniciu and Meer, 2002], which is based on Mean Shift: a mode-seeking procedure that locates the maxima of a density function defined by the sampled data [Fukunaga and Hostetler, 1975]. In the context of clustering, the algorithm iteratively moves the

centre of a proposed cluster into a higher-density region, until convergence. First, multiple clusters are initialised arbitrarily, requiring only a single parameter, the bandwidth of a kernel h, which describes the expected 'radius' or 'width' of a cluster. The number of initial clusters depends on the computational resource; one may even initialise one cluster per spike event. Then, in every subsequent iteration, the cluster centre is moved by the *mean shift vector*: a vector between the mean of all the points belonging to the cluster and the current cluster centre (Figure 3.2). Clusters whose centres are closer than the bandwidth h are merged together, and the one with the highest density is chosen. The algorithm converges when mean shift vectors for every cluster are zero in the given iteration.

The MeanShift clustering algorithm is particularly suitable for spike sorting:

1. Contrary to many other clustering methods, the number of clusters does not have to be defined in advance. This is crucial, as in spike sorting the exact number of cells is not know in advance.

2. There is only one parameter, the bandwidth, and even that can be estimated in advance as it corresponds to 'physical' width. This makes it suitable for use in a spike sorting toolkit which aspires to be automatic and hence non-parametric.

3. The algorithm is parallelisable by design, where every Mean Shift procedure can be run in one thread. Under the hood, the spike sorting toolkit by Hilgen et al. [2017] uses an implementation of MeanShift clustering from the scikit-learn library [Pedregosa et al., 2011], which is parallel out of the box.

Once the clustering algorithm converges, all events in a single cluster shall exhibit similar waveform and will have occurred at a similar spatial location, which indicates their common origin from one neuron. Clustered events are then fed into the visualisation framework, which is where the scope of this project ends and neuroscience research begins.

Symbol	Default value	Description
θ	17	Detection threshold
τ_{event}	$1ms$	Maximum depolarisation width
θ_b	0	Repolarisation threshold
α	0.3	Weight of the waveform component for clustering
h	3	Bandwidth of the MeanShift kernel
d	4	Number of PCs used to describe a spike shape

Table 3.1: Parameters of the spike sorting toolkit used in this project showing default values, as proposed by Muthmann et al. [2015] and Hilgen et al. [2017].

3.3 Bayesian Parameter Optimisation

3.3.1 Motivation

Any quantitative neuroscience research must be based on efficient and reproducible procedures, where only little subjective 'tweaking' and human intervention is required. The need for automatic spike sorting procedures have been voiced for a long time [Abeles and Goldstein, 1977], but reasonable fully automatic implementations have been proposed only very recently, e.g. the MountainSort toolkit [Chung et al., 2017].

Although the implementation used in this project does still involve manual parameter setting, the availability of the ground-truth parallel dataset changes the paradigm of the task. Full automation may be achieved by automating the 'parameter tweaking' part of the workflow, dubbed *hyperparameter optimisation*, a subject of significant interest in the Machine Learning community. Instead of a semi-automatic evaluation and a manual parameter setting performed by the researcher (Figure 3.3), an automatic procedure may be used, where the evaluation step is performed automatically using the ground-truth data and a new, more optimal parameter configuration is chosen by some policy. In the context of this project, the task then becomes to optimise parameters in Table 3.1 to maximise the detection and sorting performance.

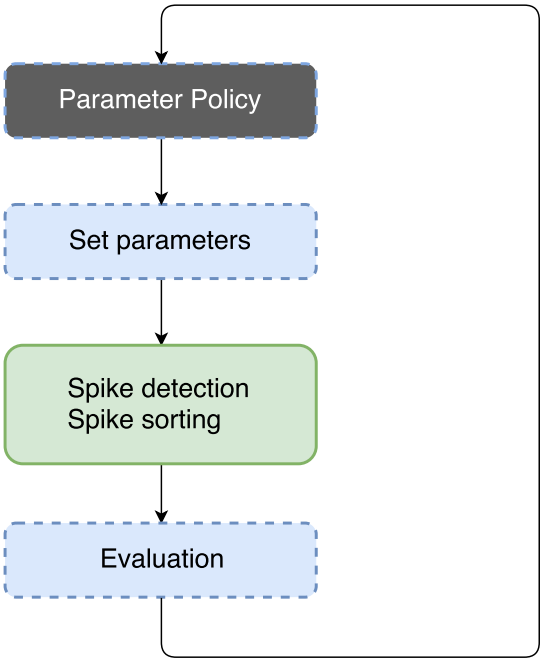

Figure 3.3: Typical flow of adapting a sorting algorithm to a new set of data. The blue rectangles represent parts often performed manually by the researcher, who follows an arbitrary black-box policy to set the parameters.

The problem can be then reframed analytically: let the vector $\mathbf{x}$ denote a configuration of parameters. We run the computationally expensive detection or sorting procedure, evaluate it using ground-truth data (Figure 3.3) and then obtain a scalar value. Let this

be denoted by a function $f(\mathbf{x})$, which essentially maps the performance of the algorithm (e.g. missed detections on a specific channel) over the input domain of algorithm parameters. The goal then becomes to find the arg minimum of f. There are various strategies to find this minimum. One could search the whole domain of input parameters with coarse granularity, trying out every permutation (the so-called grid search), or one could even search the input space randomly which, surprisingly, has been shown to outperform the grid search when optimising hyperparameters for neural networks [Bergstra and Bengio, 2012]. However, when choosing this strategy, it is important to realise that every evaluation of f takes a significant amount of computation; hence one can trade-off more elaborate optimisation algorithm if it leads to fewer evaluations of f. In this project, a well-known probabilistic approach was taken.

3.3.2 Bayesian optimisation with Gaussian Processes

In essence, Bayesian Optimisation consists of two components:

1. A prior over functions, which describes our beliefs about f. This gives us an analytical framework to reason about the behaviour of f, i.e. the behaviour of the performance of the sorting algorithm given some parameters.
2. An *acquisition function,* which given the posterior, identifies the next point (i.e. set of parameter values) to evaluate.

The following section outlines the optimisation strategy used in this project. For an in-depth statistical description and evaluation of Bayesian Optimisation see e.g. [Brochu et al., 2010].

3.3.2.1 Gaussian Process Prior

A Gaussian process (GP) is a collection of random variables such that every subset of those variables has a multivariate normal distribution. We can use GPs to approximate functions if we discretise their input space, which is the case if we allow certain granularity of our parameter values (e.g. the float datatype with IEEE precision [Kahan, 1996]). GPs are also a good fit for a prior, as joint Gaussians can be marginalised to then obtain the posterior. In the context of the parameter optimisation, given previous evaluations f_{prev} one can compute posterior for point $\mathbf{x}_{new}$ which will be a normal distribution defined by its mean and variance. We then not only get the expected value for $\mathbf{x}_{new}$ but also a measure of uncertainty of our model (the variance). Formally, given that we observed f_{prev} for some points $\mathbf{x}_{prev}$, we can reason about the behaviour of f on previously unseen $\mathbf{x}_*$ in a bayesian way:

$$p\left(f_* \mid \mathbf{x}_{prev}, \mathbf{x}_*, f_{prev}\right) \tag{3.5}$$

Since we are using GP to describe f, we know f_* and f_{prev} are jointly Gaussian:

$$\begin{pmatrix} f_{prev} \\ f_* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu_{prev} \\ \mu_* \end{pmatrix}, \begin{pmatrix} K_{prev} & K_* \\ K_*^T & K_{**} \end{pmatrix}\right) \tag{3.6}$$

where K is a positive definite covariance (or kernel) function, which defines a method to compare similarity of points in the input space. The Matérn kernel [Matérn, 1986] was used in this project, since it has been shown to yield more complex functions (and therefore better approximations) in neural-net hyperparameter optimisation setting [Snoek et al., 2012]. For an exhaustive overview of the inner workings of Gaussian processes see the definitive book on GP by Rasmussen [2004].

3.3.2.2 Expected Improvement

Given evaluations f_{prev} we can now compute the posterior for any $\mathbf{x}_{new}$. To choose the best $\mathbf{x}_{new}$ we need the second component of the Bayesian Optimisation method, the acquisition function. For this project, the expected improvement function was chosen [Mockus et al., 1978], as it addresses the Exploration-Exploitation dilemma nicely, and has been shown to perform well in wide variety of optimisation domains [Wagner et al., 2010]. The expected improvement is defined as follows:

$$EI(\mathbf{x}) = \mathbb{E}\left[max\{0, f(\mathbf{x}_{new}) - f(\mathbf{x}_{best})\right] \tag{3.7}$$

To see the Exploration-Exploitation dynamics, Mockus et al. [1978] evaluate Equation 3.7 analytically under the GP model:

$$EI(\mathbf{x}) = \begin{cases} (\mu_{new} - f_{best})\,\Phi(Z) + \sigma_{new}\phi(Z) & \sigma_{new} > 0 \\ 0 & \sigma_{new} = 0 \end{cases} \tag{3.8}$$
$$Z = \frac{\mu_{new} - f_{best}}{\sigma_{new}}$$

where ϕ and Φ denote the PDF and CDF of the multivariate standard normal distribution respectively. From this we can see that EI is high when a) the expected value of a new candidate μ_{new} is significantly higher then the previous best evaluation, or b) when the uncertainty (i.e. variance) around the point $\mathbf{x}_{new}$ is high, which corresponds to exploitation and exploration respectively.

The complete Bayesian Optimisation procedure is therefore as follows:

1. Evaluate f for some random initialisations to obtain f_{prev}.
2. Use all previous observations to update the posterior under the GP model.
3. Find the optimal $\mathbf{x}_{new}$ where $\mathbf{x}_{new} = argmax\, EI(\mathbf{x})$.
4. Compute $f(\mathbf{x}_{new})$ and repeat steps 2.– 4. until a stopping criterion is reached.

This procedure has been implemented using the `scikit-optimize` library [Scikit-optimize, 2016].

3.4 Parameter optimisation automatisation

The following section describes the design of the automatised parameter optimisation of the algorithms by Muthmann et al. [2015] and Hilgen et al. [2017]. These methods were run on servers hosted by a remote cloud service provider, on machines with 32 CPUs and 60 GB of RAM. In principal, these methods can be run locally, but would take several days of wall clock time to complete.

3.4.1 Detection

The first two steps of the feedback loop in Figure 3.3 were implemented using Bayesian Optimisation as described above. For evaluation, a method was created, which takes the ground-truth spike train, the spike train detected at a certain channel, and its neighbours, and counts how many ground-truth events were detected on the probe (true positives) and how many were missed (false negatives) for that particular channel and its neighbours. Since this task is embarrassingly parallel, the implementation can spawn new threads until it saturates either CPU or RAM. To calculate $f(\mathbf{x})$, it takes the false negatives count on the most pronounced channel in that recording (usually the channel closest to the ground-truth pipette):

$$\begin{aligned} ch_{pron} &= arg\,max\,(TP[ch]) \quad \forall ch \\ f(\mathbf{x}) &= FN[ch_{pron}] - \theta \end{aligned} \tag{3.9}$$

Regularisation is necessary to avoid a naïve solution, where the threshold is set extremely low, detecting every noisy deviation as a spike, hence minimising false negatives; a problem usually solved by including false positives in the 'cost' function. This regularisation elegantly solves the lack of false positives (since multiple neurons can be detected on a single channel, definition of false positive is impossible, given ground-truth from only a single neuron), a problem identified in the first year of this project [Chapter 4, p19. Horváth, 2017].

3.4.2 Sorting

To evaluate the sorting performance, we count how many of the ground-truth spikes get sorted into one cluster (true positives), and how many spikes in that cluster are not found in the ground-truth spiketrain (false positives). Note that in the ideal case, every cluster corresponds to the activity of one cell only. Hence all ground-truth spikes should get assigned to one exclusive cluster. We then calculate the function value:

$$f(\mathbf{x}) = -(1+\beta^2)\frac{p*r}{p\beta^2+r} \tag{3.10}$$

which corresponds to the F_β score, where p is precision and r is recall defined as $p = \frac{TP}{n\_in\_cluster}$ and $r = \frac{TP}{n\_ground\_truth}$. The value $\beta = 1$ was used.

Chapter 4

Results

The spike detection and sorting algorithms were validated using the *2015_09_03_Cell.9.0* paired recording from the dataset series published by Neto et al. [2016]. Since this 10 minute recording is obtained from a cell particularly close to the MEA probe, the signal amplitudes on the MEA are exceptionally high (Table 4.1). The timestamps of the ground-truth events were extracted from the juxtacellular recording using the detection algorithm outlined in Section 3.1.

To confirm that all ground-truth spikes originate from the same cell, a simple shape analysis was performed using the Principal Component Analysis. PCA is an orthogonal linear transformation, which projects multi-dimensional data into a new space, such that the first dimension of this space (the first principal component) describes the greatest variance in the data. Given a set of spike shapes, each represented as a multi-dimensional vector of raw voltage values, we can therefore quantify similarity of their shapes. Figure 4.1 shows, how inadequate values of the detection threshold θ resulted in detection of spikes with different shape, possibly from a different cell. Therefore, a stricter detection threshold ($\theta = 15$) has been chosen and the ground-truth channel was manually inspected to guarantee correctness.

2015_09_03_Cell.9.0	
$593.2s$	Duration of recording
$29\mu m$	MEA-Pipette distance
$419\mu V$	Average peak-to-peak amplitude on the MEA channel closest to the pipette
$27, 1$	Coordinates of the most pronounced channel (*row, column*)

Table 4.1: Properties of the paired recording used for validation.

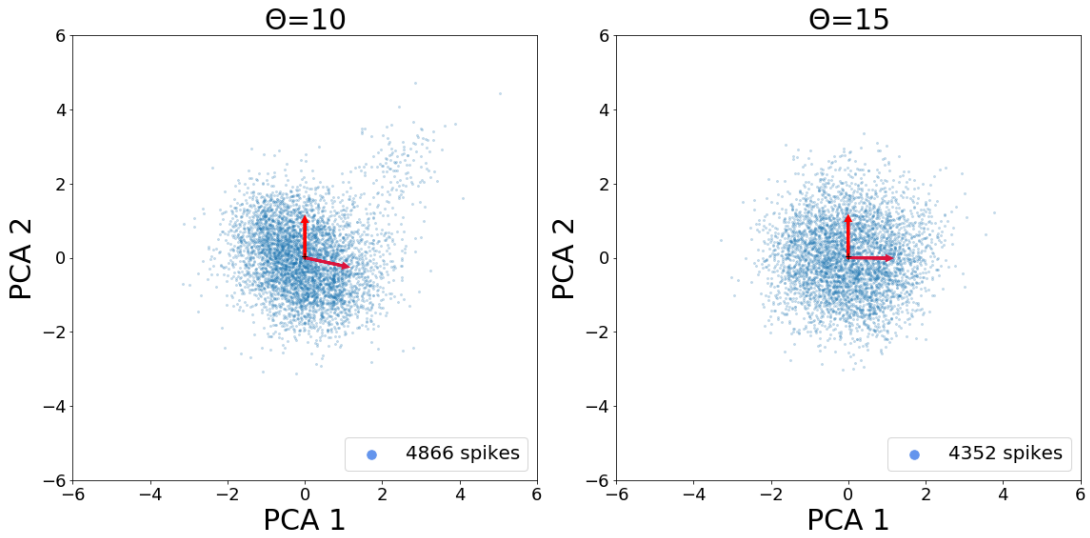

Figure 4.1: The shapes of ground-truth spikes projected into PC space, detected using two different thresholds θ. The shape of these PC clusters can be described by another eigenvalue decomposition of their covariance, eigenvectors of which are shown in red.

4.1 Validation of spike detection

To validate the spike detection algorithm the Bayesian Parameter Optimisation procedure was run for 50 iterations, out of which the first 30 were initialised with random parameters. As defined in Subsection 3.4.1, the procedure was minimising the number of missed detections, where 0 missed events on the observed channel is considered a perfect detection. Coincidently, already the first iteration selected such parameters that the detection missed less events than the value of the threshold θ, hence yielding negative values of $f(\mathbf{x})$. After the 41st iteration, the procedure converged to the optimal parameter settings: $\theta = 196, \theta_b = 17, \tau_{event} = 27$.

To understand the relationship between the individual parameters, and therefore to understand how the Bayesian Optimisation procedure found the optimum, the parameter space was examined in detail (Figure 4.3). Interestingly, the repolarisation threshold θ_b was found to have only a small variance in partial dependence, suggesting that this parameter impacts the overall detection performance only slightly. The partial dependence of detection threshold is almost linear along the whole domain, which is consistent with the fact that we use it as a regulariser and hence every change to this parameter has a small impact on the overall function value. The most interesting of the three however is the τ_{event} space. This parameter is used to discard events, which have not depolarised in τ_{event} seconds. The optimisation procedure was therefore able to discover that events in hippocampus take at least 12 frames ($= 0.4ms$) to depolarise, a value corresponding to the findings of biological research.

Using these parameters, the algorithm has performed a perfect detection on the most pronounced channel (Figure 4.2), and near-perfect detection on neighbouring channels. Conversely, the far side of the MEA detected only a handful, if any, of these events, proving that the parameters were strict enough to only detect relevant events.

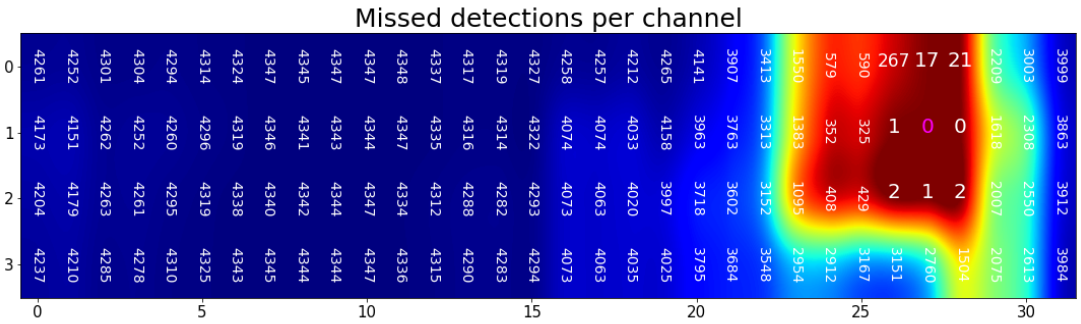

Figure 4.2: Missed detections (FNs) of the ground-truth events on MEA probe's channels. The most pronounced channel is marked pink. Axes are in probe coordinates.

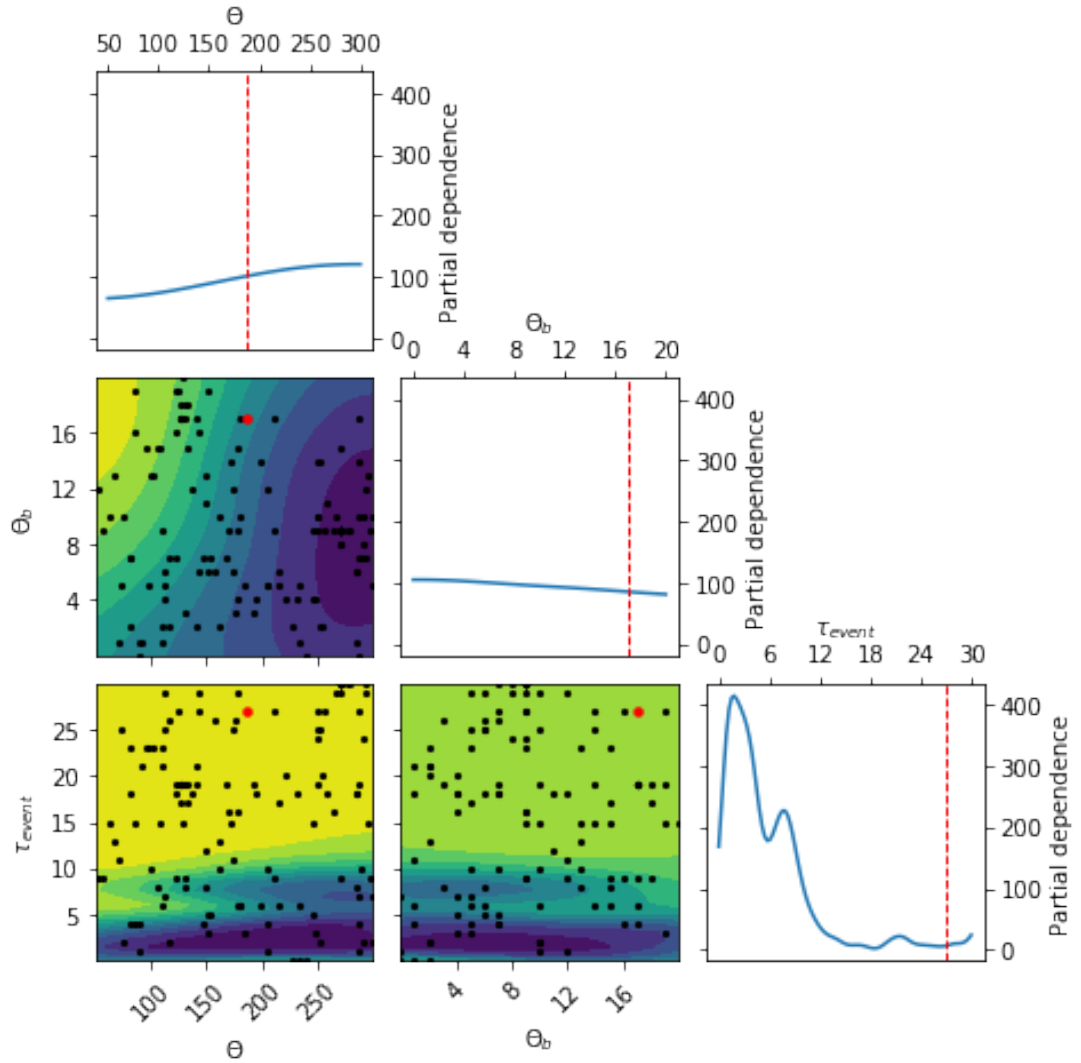

Figure 4.3: Exploration of the detection parameter space $(\theta, \theta_b, \tau_{event})$. Line-plots on the diagonal show the impact of the values of the respective parameter on $f(\mathbf{x})$ in a form of partial dependence plots, a standard measure proposed by Friedman [2001]. The lower triangle then visualises a space defined by pairs of these parameters. Every combination evaluated by the optimisation procedure (black point) and the optimal values found (red) are shown. Figure generated using the `scikit-optimize` library.

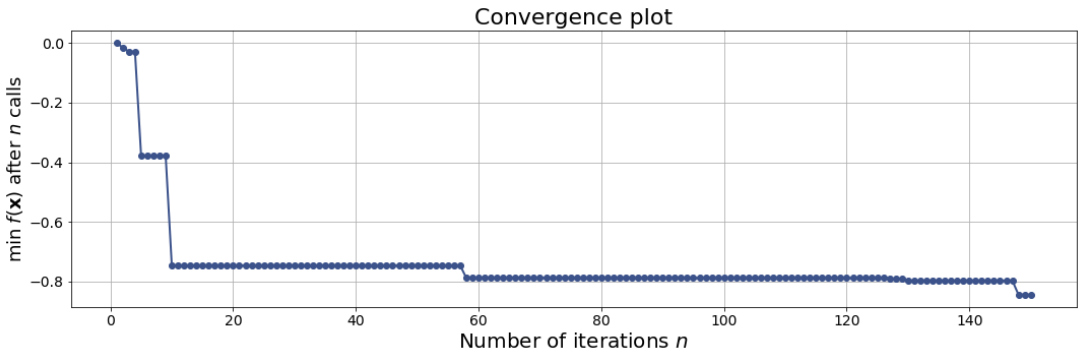

Figure 4.4: The convergence plot of the Bayesian Optimisation procedure over 150 iterations, minimising the negated F_1 score. First 70 iterations are initialised at random.

4.2 Validation of spike sorting

The spikes detected using the optimal parameters were then used to validate the sorting algorithm. The Bayesian Parameter Optimisation procedure was run for 150 iterations, with 70 random initialisations, trying to maximise (or minimise the negation of) the F_1 score (see Subsection 3.4.2), where the perfect sorting ($F_1 = 1.0$) would group all ground-truth spikes into one exclusive cluster containing no other spikes. The random initialisations managed to discover two configurations which increased the performance significantly (Figure 4.4). Afterwards, the procedure improved the performance only slightly and, surprisingly, converged to a sub-optimal configuration, with the parameter values: $\alpha = 0.0845, h = 0.2810, d = 8$. The most surprising is the α parameter, since such low value effectively means the sorting will not take the shape properties of the spikes into account, and only sort based upon location.

Detailed examination of the neighbourhood around the most pronounced channel revealed the cause of this sub-optimal performance (Figure 4.5).

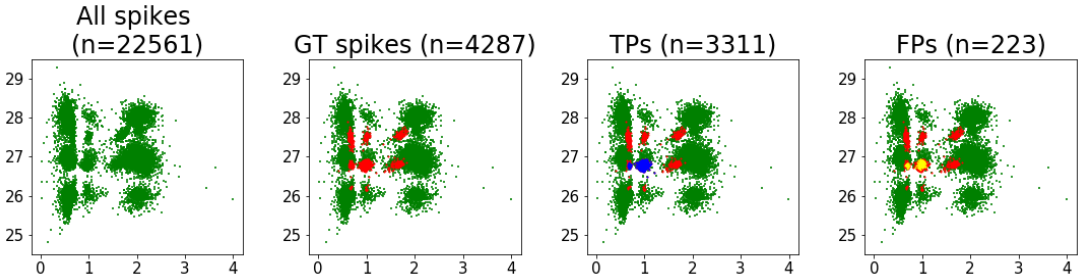

Figure 4.5: The neighbourhood of the most pronounced channel with all spikes detected (green), of which the spikes corresponding to the ground-truth (red), of which clustered correctly (blue). In yellow, False Positives are shown, which were assigned into the same cluster, however don't correspond to any ground truth spike. Both axes show the probe coordinates.

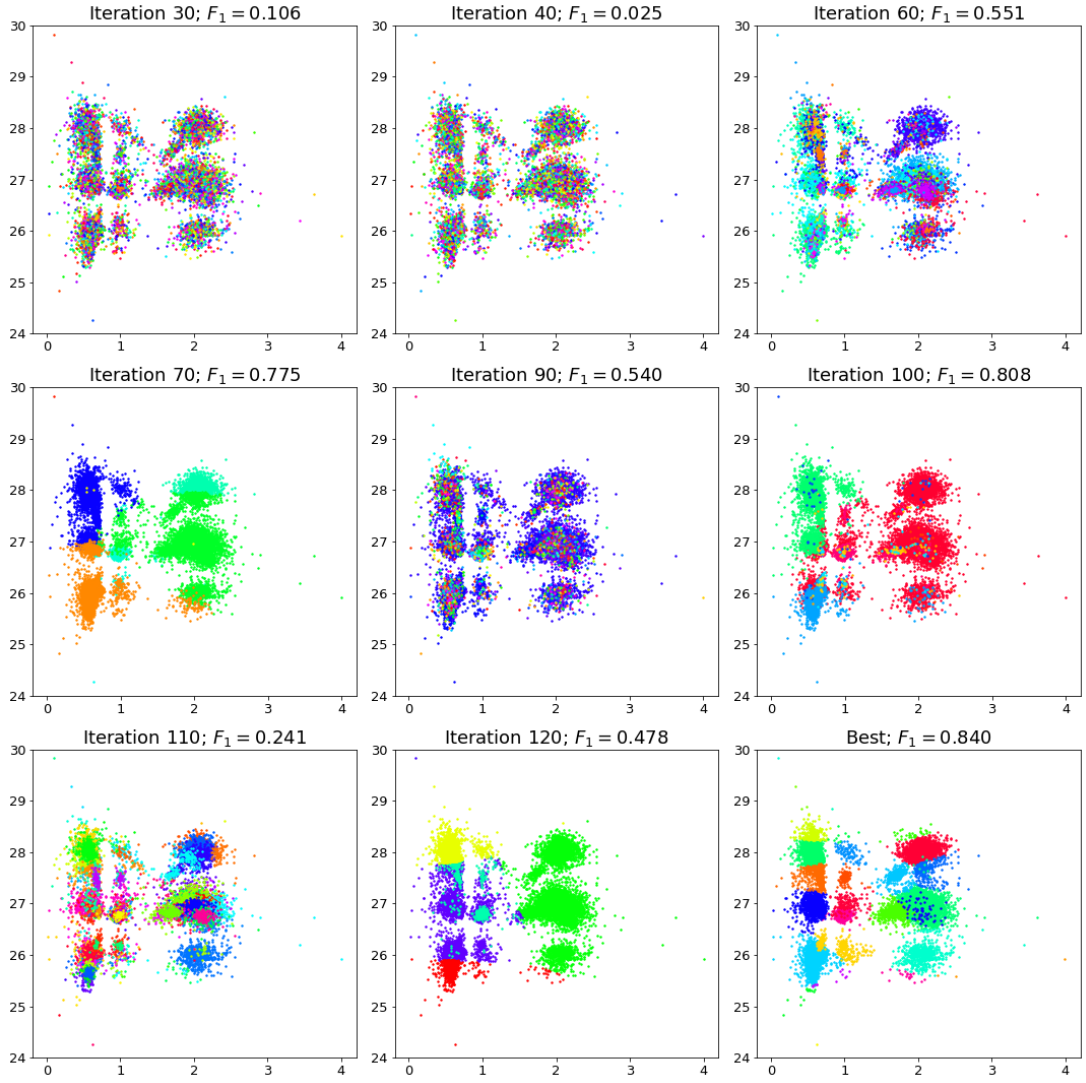

Figure 4.6: An overview of various clustering configurations attempted during the optimisation procedure. The colours are sampled arbitrarily for every configuration, hence no correspondence in between configurations can be assumed. Axes are in probe coordinates.

The localisation step, which is performed at the end of the detection phase, has misplaced $\sim 10\%$ of the ground truth spikes away from the most-pronounced channel (Figure 4.5 Red). Given this constraint, the optimisation procedure has converged on such parameters that only cluster spikes detected on the central channel, which contains the majority of the ground-truth spikes. To achieve such localised clustering, the procedure had to choose an extremely low value of the parameter α.

This prioritisation of spatial features can be also seen in the parameter configurations of the intermediate iterations of the optimisation procedure (Figure 4.6). In iteration 120, the priority is placed on clustering by shape (observe the teal cluster), which only yields an F_1 score of 0.478. On the contrary, the iteration 70, where the wide bandwidth parameter causes spikes to be clustered into big patches based on location,

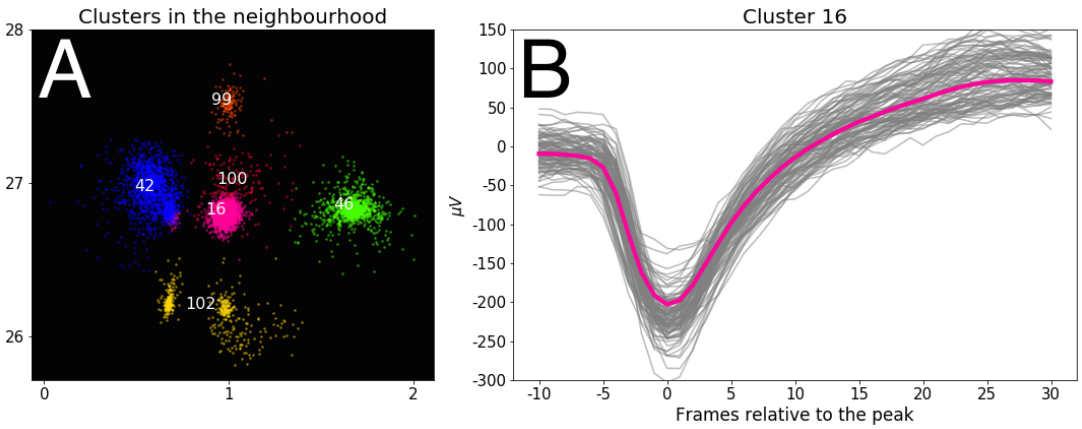

Figure 4.7: Detail view of the cluster containing the most ground-truth spikes (16) A. The 5 nearest clusters are shown in coordinate space i.e. as they are localised on the MEA probe, with any other spikes filtered out. B. Peak-aligned shapes of all spikes in the cluster are shown (grey), compared to the mean shape (magenta).

yields significantly higher score of $F_1 = 0.775$.

Despite the mis-localised spikes and the resulting unusual parameter configuration chosen by the optimisation procedure, the sorting algorithm has yielded a very plausible result. The clusters in the neighbourhood are well separated (Figure 4.7A) and the spikes in the cluster containing the most ground-truth spikes have a very pronounced shape (Figure 4.7B). Finally, both the amplitude and voltage of the mean shape of the spikes in the ground-truth cluster correspond to the findings presented by Neto et al. [2016].

Chapter 5

Discussion

Over the two years, the goals of this project have been accomplished successfully. The spike sorting toolkit has been analysed and adapted to be able to process the format of the novel paired recording datasets by Neto et al. [2016]. The performance of the detection and sorting algorithms has been analysed in depth and their correctness has been validated. What is more, an automated framework has been devised, such that any paired recording can be now used not only for validation of these algorithms but also for a fully-automatic parameter optimisation. This framework is implemented as a single, easy-to-use Python class, which is compatible with the latest iteration of the detection and sorting algorithms. The implementation takes three inputs: a path to a dataset with raw MEA data, a path to the ground-truth spiketrain file (effectively a list of timestamps) and a dictionary of parameter domains to be optimised over. The optimal parameter configurations for both detection and sorting are then returned. Therefore, the research community using these algorithms can now both easily obtain a guarantee of correctness of their electrophysiological results, and save considerable amounts of time by not having to perform tedious manual parameter search.

Validation of spike detection
A perfect detection has been achieved on the MEA data, with a surprisingly high threshold. It is therefore reasonable to assume that the detection algorithm could achieve similar correctness even for data with significantly lower amplitudes (e.g. recorded cells which are further away). Three out of five parameters of the detection algorithm were optimised. Inspecting the partial dependence of each, it has been observed that the θ_b parameter does not influence the detection performance significantly. This shows that the Bayesian Optimisation framework implemented in this project can also be used to gain invaluable insight into the design of spike detection algorithms, namely it can untangle the often complex interaction between individual parameters.

Localisation error revealed
During the evaluation of the sorting algorithm, a bug in the intermediate localisation step was discovered, where after detection, the spikes are incorrectly assigned to multiple locations. Although this obstructed any conclusive validation of the sorting performance, in a sense, it fulfils the validation objective of this project even more than

a perfect performance. Due to the high volumes of data involved in spike sorting, such accidental implementation mistakes are very difficult to spot without ground-truth data, especially since manual inspections (e.g. Figure 4.7) would show very plausible results. Bayesian Optimisation framework can therefore be used not only to evaluate performance, but also to aid development of spike sorting toolkits.

Validation of spike sorting
Apropos plausible results, even given such constraint as misplaced spikes (or in the future maybe an extremely noisy dataset), the optimisation framework was still able to find such parameters which included $\sim 77\%$ of the ground-truth spikes. It should be noted that these were very unusual parameters, which would very likely be never chosen by a human. Although the correctness of spike sorting algorithm could not be validated conclusively, the detailed inspection of its various modes of performance (Figure 4.6) shows that the algorithm is capable of prioritising either spatial or waveform properties of spikes, or balancing both.

Automation via Bayesian Optimisation
Lastly, this project has shown that Bayesian Optimisation is suitable for optimising spike sorting algorithms. Although the implementation of this project focused specifically on algorithms by Muthmann et al. [2015], Hilgen et al. [2017], the theoretical framework could be applied to any spike sorting algorithm to not only validate its performance but to transform it into a fully automated spike sorting toolkit.

5.1 Future work in the field

Automation of existing spike sorting toolkits (e.g. using Bayesian Optimisation) and the emergence of new automated toolkits in the near future will enable independent benchmarks to be constructed. Benchmarks, which will evaluate and compare the performance and correctness in plethora of situations, and will eventually lead to thorough validation of every spike sorting toolkit. However, in order to construct such benchmarks, many more paired ground-truth datasets have to be recorded. It is encouraging to see recent development in this area: only two weeks before the submission of this project, Yger et al. [2018] published a set of 20 paired recordings, using a near-identical setup to Neto et al. [2016] but with a MEA probe with twice as many recording channels.

To lay solid foundations for fully understanding how the human brain works, the current undesirable state where *"most laboratories still rely on manual intervention"* [page 2. Chung et al., 2017] needs to be resolved. Automated validation procedures are a step in that direction. A step in the direction of reliable and reproducible neurophysiological research.

Bibliography

Moshe Abeles and Moise H Goldstein. Multispike train analysis. *Proceedings of the IEEE*, 65(5):762–773, 1977.

Costas A Anastassiou, Rodrigo Perin, György Buzsáki, Henry Markram, and Christof Koch. Cell type-and activity-dependent extracellular correlates of intracellular spiking. *Journal of neurophysiology*, 114(1):608–623, 2015.

Alex H Barnett, Jeremy F Magland, and Leslie F Greengard. Validation of neural spike sorting algorithms without ground-truth information. *Journal of neuroscience methods*, 264:65–77, 2016.

Luca Berdondini, PD Van Der Wal, Olivier Guenat, Nicolaas F de Rooij, Milena Koudelka-Hep, P Seitz, R Kaufmann, P Metzler, N Blanc, and S Rohr. High-density electrode array for imaging in vitro electrophysiological activity. *Biosensors and bioelectronics*, 21(1):167–174, 2005.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.

Julius Bernstein. Ueber den zeitlichen verlauf der negativen schwankung des nervenstroms. *Archiv für die gesamte Physiologie des Menschen und der Tiere*, 1(1): 173–207, 1868.

Emilia Biffi, Diego Ghezzi, Alessandra Pedrocchi, and Giancarlo Ferrigno. Development and validation of a spike detection and classification algorithm aimed at implementation on hardware devices. *Computational intelligence and neuroscience*, 2010:8, 2010.

Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

Jason E Chung, Jeremy F Magland, Alex H Barnett, Vanessa M Tolosa, Angela C Tooker, Kye Y Lee, Kedar G Shah, Sarah H Felix, Loren M Frank, and Leslie F Greengard. A fully automated approach to spike sorting. *Neuron*, 95(6):1381–1394, 2017.

Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 24(5): 603–619, 2002.

Markus Diesmann, Marc-Oliver Gewaltig, and Ad Aertsen. Stable propagation of synchronous spiking in cortical neural networks. *Nature*, 402(6761):529, 1999.

George Dimitriadis, Joana P Neto, Arno Aarts, Andrei Alexandru, Marco Ballini, Francesco Battaglia, Lorenza Calcaterra, Francois David, Richard Fiath, Joao Frazao, et al. Why not record from every channel with a cmos scanning probe? 2018.

Robert M Dowben and Jerzy E Rose. A metal-filled microelectrode. *Science*, 118 (3053):22–24, 1953.

Jelena Dragas, Vijay Viswam, Amir Shadmani, Yihui Chen, Raziyeh Bounik, Alexander Stettler, Milos Radivojevic, Sydney Geissler, Marie Engelene J Obien, Jan Müller, et al. In vitro multi-functional microelectrode array featuring 59 760 electrodes, 2048 electrophysiology channels, stimulation, impedance measurement, and neurotransmitter detection channels. *IEEE journal of solid-state circuits*, 52(6): 1576–1590, 2017.

Jiangang Du, Timothy J Blanche, Reid R Harrison, Henry A Lester, and Sotiris C Masmanidis. Multiplexed, high density electrophysiology with nanofabricated neural probes. *PloS one*, 6(10):e26204, 2011.

Gaute T Einevoll, Felix Franke, Espen Hagen, Christophe Pouzat, and Kenneth D Harris. Towards reliable spike-train recordings from thousands of neurons with multielectrodes. *Current opinion in neurobiology*, 22(1):11–17, 2012.

Bjorn Eversmann, Martin Jenkner, Franz Hofmann, Christian Paulus, Ralf Brederlow, Birgit Holzapfl, Peter Fromherz, Matthias Merz, Markus Brenner, Matthias Schreiter, et al. A 128/spl times/128 cmos biosensor array for extracellular recording of neural activity. *IEEE Journal of Solid-State Circuits*, 38(12):2306–2317, 2003.

Julien Fournier, Christian M Mueller, Mark Shein-Idelson, Mike Hemberger, and Gilles Laurent. Consensus-based sorting of neuronal spike waveforms. *PloS one*, 11(8):e0160494, 2016.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

Keinosuke Fukunaga and Larry Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, 21(1):32–40, 1975.

Carl Gold, Darrell A Henze, Christof Koch, and Gyorgy Buzsaki. On the origin of the extracellular action potential waveform: a modeling study. *Journal of neurophysiology*, 95(5):3113–3128, 2006.

Kenneth D Harris, Darrell A Henze, Jozsef Csicsvari, Hajime Hirase, and Gyorgy Buzsaki. Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. *Journal of neurophysiology*, 84(1): 401–414, 2000.

Darrell A Henze and György Buzsáki. Hilar mossy cells: functional identification and activity in vivo. *Progress in brain research*, 163:199–810, 2007.

Darrell A Henze, Zsolt Borhegyi, Jozsef Csicsvari, Akira Mamiya, Kenneth D Harris, and György Buzsáki. Intracellular features predicted by extracellular recordings in the hippocampus in vivo. *Journal of neurophysiology*, 84(1):390–400, 2000.

Gerrit Hilgen, Martino Sorbaro, Sahar Pirmoradian, Jens-Oliver Muthmann, Ibolya Edit Kepiro, Simona Ullo, Cesar Juarez Ramirez, Albert Puente Encinas, Alessandro Maccione, Luca Berdondini, et al. Unsupervised spike sorting for large-scale, high-density multielectrode arrays. *Cell reports*, 18(10):2521–2532, 2017.

Jano Horváth. Validating spike detection algorithms against ground-truth electrophysiological data. Master's thesis, The University of Edinburgh, 2017.

Tomáš Hromádka, Michael R DeWeese, and Anthony M Zador. Sparse representation of sounds in the unanesthetized auditory cortex. *PLoS biology*, 6(1):e16, 2008.

James J Jun, Nicholas A Steinmetz, Joshua H Siegle, Daniel J Denman, Marius Bauza, Brian Barbarits, Albert K Lee, Costas A Anastassiou, Alexandru Andrei, Çağatay Aydın, et al. Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679):232, 2017.

William Kahan. Ieee standard 754 for binary floating-point arithmetic. *Lecture Notes on the Status of IEEE*, 754(94720-1776):11, 1996.

Kyung Hwan Kim and Sung June Kim. Neural spike sorting under nearly 0-db signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier. *IEEE Transactions on Biomedical Engineering*, 47(10):1406–1411, 2000.

Alessandro Maccione, Mauro Gandolfo, Paolo Massobrio, Antonio Novellino, Sergio Martinoia, and Michela Chiappalone. A novel algorithm for precise identification of spikes in extracellularly recorded neuronal signals. *Journal of neuroscience methods*, 177(1):241–249, 2009.

Adam H Marblestone, Bradley M Zamft, Yael G Maguire, Mikhail G Shapiro, Thaddeus R Cybulski, Joshua I Glaser, Dario Amodei, P Benjamin Stranges, Reza Kalhor, David A Dalrymple, Dongjin Seo, Elad Alon, et al. Physical principles for scalable neural recording. *Frontiers in computational neuroscience*, 7:137, 2013.

Bertil Matérn. Spatial variation, volume 36 of lecture notes in statistics, 1986.

J Mockus, V Tiesis, and A Zilinskas. Toward global optimization, volume 2, chapter bayesian methods for seeking the extremum. 1978.

Jens-Oliver Muthmann, Hayder Amin, Evelyne Sernagor, Alessandro Maccione, Dagmara Panas, Luca Berdondini, Upinder S Bhalla, and Matthias H Hennig. Spike detection for large neural populations using high density multielectrode arrays. *Frontiers in neuroinformatics*, 9:28, 2015.

Joana P Neto, Gonçalo Lopes, João Frazão, Joana Nogueira, Pedro Lacerda, Pedro Baião, Arno Aarts, Alexandru Andrei, Silke Musa, Elvira Fortunato, et al. Validating

silicon polytrodes with paired juxtacellular recordings: method and dataset. *Journal of neurophysiology*, 116(2):892–903, 2016.

Marius Pachitariu, Nicholas Steinmetz, Shabnam Kadir, Matteo Carandini, and Kenneth D Harris. Kilosort: realtime spike-sorting for extracellular electrophysiology with hundreds of channels. *BioRxiv*, page 061481, 2016.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

Carlos Pedreira, Juan Martinez, Matias J Ison, and Rodrigo Quian Quiroga. How many neurons can we see with current spike sorting algorithms? *Journal of neuroscience methods*, 211(1):58–65, 2012.

Klas H Pettersen, Espen Hagen, and Gaute T Einevoll. Estimation of population firing rates and current source densities from laminar electrode recordings. *Journal of computational neuroscience*, 24(3):291–313, 2008.

F Pothof, T Galchev, M Patel, A Sayed Herbawi, O Paul, and P Ruther. 128-channel deep brain recording probe with heterogenously integrated analog cmos readout for focal epilepsy localization. In *Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS), 2015 Transducers-2015 18th International Conference on*, pages 1711–1714. IEEE, 2015.

Jason S Prentice, Jan Homann, Kristina D Simmons, Gašper Tkačik, Vijay Balasubramanian, and Philip C Nelson. Fast, scalable, bayesian spike identification for multi-electrode arrays. *PloS one*, 6(7):e19884, 2011.

R Quian Quiroga, Zoltan Nadasdy, and Yoram Ben-Shaul. Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. *Neural computation*, 16(8):1661–1687, 2004.

Bogdan C Raducanu, Refet F Yazicioglu, Carolina M Lopez, Marco Ballini, Jan Putzeys, Shiwei Wang, Alexandru Andrei, Veronique Rochus, Marleen Welkenhuysen, Nick van Helleputte, et al. Time multiplexed active neural probe with 1356 parallel recording sites. *Sensors*, 17(10):2388, 2017.

Carl Edward Rasmussen. Gaussian processes in machine learning. In *Advanced lectures on machine learning*, pages 63–71. Springer, 2004.

M Recce. The tetrode: a new technique for multi-unit extracellular recording. In *Soc. Neurosci. Abstr.*, volume 15, page 1250, 1989.

Cyrille Rossant, Shabnam N Kadir, Dan FM Goodman, John Schulman, Maximilian LD Hunter, Aman B Saleem, Andres Grosmark, Mariano Belluscio, George H Denfield, Alexander S Ecker, et al. Spike sorting for large, dense electrode arrays. *Nature neuroscience*, 19(4):634, 2016.

Patrick Ruther and Oliver Paul. New approaches for cmos-based devices for large-scale neural recording. *Current opinion in neurobiology*, 32:31–37, 2015.

Bert Sakmann and Erwin Neher. Patch clamp techniques for studying ionic channels in excitable membranes. *Annual review of physiology*, 46(1):455–472, 1984.

Scikit-optimize. Scikit-optimize: a simple and efficient library to minimize (very) expensive and noisy black-box functions., March 2016. URL `https://github.com/scikit-optimize/scikit-optimize`.

Shy Shoham, Matthew R Fellows, and Richard A Normann. Robust, automatic spike sorting using mixtures of multivariate t-distributions. *Journal of neuroscience methods*, 127(2):111–122, 2003.

Leslie S Smith and Nhamoinesu Mtetwa. A tool for synthesizing spike trains with realistic interference. *Journal of Neuroscience Methods*, 159(1):170–180, 2007.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

Zoltan Somogyvari, Dorottya Cserpán, István Ulbert, and Péter Erdi. Localization of single-cell current sources based on extracellular potential patterns: the spike csd method. *European Journal of Neuroscience*, 36(10):3299–3313, 2012.

Nicholas A Steinmetz, Christof Koch, Kenneth D Harris, and Matteo Carandini. Challenges and opportunities for large-scale electrophysiology with neuropixels probes. *Current opinion in neurobiology*, 50:92–100, 2018.

Ian H Stevenson and Konrad P Kording. How advances in neural recording affect data analysis. *Nature neuroscience*, 14(2):139, 2011.

Tobias Wagner, Michael Emmerich, André Deutz, and Wolfgang Ponweiser. On expected-improvement criteria for model-based multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 718–727. Springer, 2010.

Michael Wehr, John S Pezaris, and Maneesh Sahani. Simultaneous paired intracellular and tetrode recordings for evaluating the performance of spike sorting algorithms. *Neurocomputing*, 26:1061–1068, 1999.

Pierre Yger, Giulia LB Spampinato, Elric Esposito, Baptiste Lefebvre, Stephane Deny, Christophe Gardella, Marcel Stimberg, Florian Jetter, Guenther Zeck, Serge Picaud, et al. Fast and accurate spike sorting in vitro and in vivo for up to thousands of electrodes. *bioRxiv*, page 067843, 2016.

Pierre Yger, Giulia LB Spampinato, Elric Esposito, Baptiste Lefebvre, Stéphane Deny, Christophe Gardella, Marcel Stimberg, Florian Jetter, Guenther Zeck, Serge Picaud, et al. A spike sorting toolbox for up to thousands of electrodes validated with ground truth recordings in vitro and in vivo. *eLife*, 7:e34518, 2018.