LUND UNIVERSITY

Signal Modeling and Data Reduction for Wireless Brain-Machine Interfaces

Thorbergsson, Palmi Thor

2012

Signal Modeling and Data Reduction for Wireless Brain-Machine Interfaces

Palmi Thor Thorbergsson

Doctoral Thesis
Radio Systems

Lund University
Lund, Sweden
2012

Department of Electrical and Information Technology
Lund University
Box 118, SE-221 00 LUND
SWEDEN

This thesis is set in Computer Modern 10pt
with the LaTeX Documentation System

Series of licentiate and doctoral theses
No. 45
ISSN 1654-790X
ISBN 978-91-7473-388-4

Printed in Sweden by *Tryckeriet i E-huset*, Lund.
September 2012.

Till Charlotte och Oliver

Abstract

Brain-machine interfaces (BMIs) provide a uni- or bidirectional communication link between the central nervous system and the outside world. This link facilitates the studying of neuronal mechanisms underlying behavior as well as the treatment of neurological disease. Wired BMIs are limited in the sense that they restrict the mobility of the subject and they increase the risks for post-surgical complications. While wireless BMIs ideally solve these problems, their designers face the challenge of combining high information throughput with limited wireless link capacity and energy resources. Therefore, measures have to be taken to maximize the utilization of the wireless link and energy resources by designing computationally efficient and reliable data reduction techniques. The design and validation of such techniques requires the presence of well defined test data, where the true information content is known a priori. This thesis deals with both the modeling of the neural signal to provide realistic and practical means of generating test data, as well as low-complexity methods for data reduction that lead to efficient utilization of the wireless link and the energy resources at hand. The main part of the thesis is a collection of papers that address these aspects.

Paper I presents the design and implementation of a simple telemetry system for the wireless transmission of neural data from four measurement channels. This paper highlights some of the design challenges that need to be considered and thereby serves as a pilot investigation for the following papers.

Paper II presents a recording model and a simulation tool for generating single-channel test recordings for the validation of algorithms for spike detection and spike sorting. Having set up the geometry of the recording, each neuron is assigned a random spike waveforms from a library of experimentally obtained templates. The contribution of each neuron is generated by adding the corresponding waveform at randomly generated spike times and the spike trains are added up to form the entire recording. Spike times are modeled by a renewal process. The model is evaluated in terms of realism by comparing the power spectral density and autocorrelation of synthetic biological noise generated by

the model, to noise obtained from real recordings.

Paper III extends the finite spike library provided in paper II in order to provide a greater, still realistic, variation in spike waveforms. Principal component analysis and Gaussian mixture models are used to model the statistical properties of the original spike library and the statistical model can then be used to generate an arbitrary number of spike wavforms with realistic properties. The extension is shown to be usable in providing access to arbitrarily large libraries of spikes with realistic properties.

Paper IV uses the models presented in papers II and III to explore the effects of sampling rate and resolution on the performance in spike detection and spike sorting at various noise levels and numbers of target neurons. Performance curves are analyzed to find sampling rate and resolution breakpoints for spike detection and spike sorting. These breakpoints serve as guidelines for selecting sampling parameters when dimensioning wireless BMIs. The paper presents methods for quantifying the accuracy in spike detection and spike sorting and provides general insight into how the performance of these processing tasks are influenced by sampling parameters, noise level and number of target units.

Paper V presents a preliminary study of the characteristic relationship between physical electrode movements and movements of detected spikes in feature space, using the signal models presented in paper VI. We then model this relationship as a linear transformation between two coordinate systems and show that given that a training procedure is introduced at the time of electrode insertion, future electrode movements can be estimated directly from the feature space representation of spikes.

Paper VI presents a new, computationally and memory efficient approach for modeling the extracellular signal. We use traditional compression techniques and polynomial fitting to derive a deterministic model that can be used for fast calculation of spike waveforms in arbitrary measurement points surrounding a compartment model of a neuron. Four different neuron models are derived and they are all shown to accurately predict the spike waveforms produced by the original compartment model, both in terms of spike shape and amplitude. The model is implemented into a simulation tool that efficiently and realistically synthesizes recordings with multielectrode arrays of arbitrary geometries.

Paper VII addresses low-complexity methods of compressing detected spike waveforms in wireless BMIs to ensure efficient use of the wireless link and energy resources at hand. The paper shows that given the correct choice of overall system architecture and spike detector, spike waveforms can be compressed with fixed generic compression bases, derived from experimentally obtained spike libraries, without significant loss in accuracy in spike reconstruction and sorting.

Preface

During the final part of my studies of electrical engineering at LTH, I was offered, by Anders J Johansson, the opportunity of writing my master's thesis on the topic of telemetric devices for neural recordings. The thesis, titled "*Nerve Telemetry System*" was handed in and presented about one year later and addressed the design and implementation of a wireless measurement system for the purpose of recording extracellular signals. The result was a prototype built on a "breadboard" and whose main components were an analog amplifier, a prototype-board for a commercial wireless transceiver including an A/D converter and a CPU and a simple PC interface to control the devices and collect measurement data.

The initial interest in such a system came from a then recently established research group at Lund University called the *Neuronano Research Center* – or *NRC*. This multidisciplinary group, composed of experts within neuroscience and engineering, set off with the following vision:

> *To improve quality of life for disabled people and individuals with neurodegenerative disease by listening to, understanding and talking to the nervous system by means of a neuroelectronic junction.*

"*Listening to*", "*understanding*" and "*talking to*" in this context involves measuring, interpreting and reacting upon the activities of neuronal circuits within the central nervous system (CNS). This kind of neuroelectronic junctions – brain-machine interfaces (BMI) – can provide insight into the neuronal mechanisms that govern memory, learning, pain and other aspects of our behavior and they provide the opportunity to "tap onto" the nervous system to extract control signals for prosthetic devices. Realizing this noble vision thus requires significant efforts within all fields of expertise covered by the group, including for instance biocompatible electrode designs, methods for signal analysis and the design of wireless interfaces. Wireless BMIs allow the subject or patient to move around without being influenced significantly by the measurement equipment and they minimize the risk for post-surgical complications.

My master's thesis was the first step within the group towards designing a

wireless BMI and resulted in me being offered a position as a Ph.D. student for continued research within the field, under the supervision of Anders J Johansson and Martin Garwicz. I soon realized that the challenge at hand not only involved designing and building the hardware for the wireless BMI, but also, and perhaps primarily, establishing a set of design parameters that ensured good system performance in terms of "listening to" and "understanding" the nervous system. This need for pre-investigations was primarily driven by the requirement that the implanted part of the wireless BMI should be self-contained and fully automatic, thus demanding minimal power consumption and efficient use of computational and wireless link resources. Testing the performance of system designs requires realistic and practical signal models that can be used to generate test signals with controllable properties. This provides access to "ground-truth" about the neural activity being measured, and that the system output can be compared with to provide a quantitative estimate of performance.

This doctoral thesis summarizes my research within signal modeling and data reduction for wireless BMIs where computational simplicity has been one of the key criteria. In the first part of the thesis, a general overview of the research field is provided and the contributions of my research to the field are briefly summarized. The second part contains an assembly of seven research papers that have been written during my five years as a PhD student. These are:

[1] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Implementation of a Telemetry System for Neurophysiological Signals", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 1254 – 1257, 2008.

[2] P. T. Thorbergsson, H. Jorntell, F. Bengtsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Spike Library Based Simulator for Extracellular Single Unit Neuronal Signals", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 6998 – 7001, 2009.

[3] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Statistical Modelling of Spike Libraries for Simulation of Extracellular Recordings in the Cerebellum", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 4250 – 4253, 2010.

[4] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Minimizing data transfer with sustained performance in wireless brain-machine interfaces", *Journal of neural engineering*, pp. 036005, 2012.

[5] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Spike-Feature Based Estimation of Electrode Position in Extracellular Neural Recordings", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 3380 – 3383, 2012.

[6] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Computationally efficient simulation of extracellular recordings with multielectrode arrays", *Journal of neuroscience methods*, vol. 211, pp. 133 – 144, 2012.

[7] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Compression of neural spikes with fixed generic bases for wireless brain-machine interfaces", manuscript to be submitted for publication, 2012.

Acknowledgments

There are so many people who have, in one way or another, given me the help and support that I needed to find my way through the jungle of challenges that the last five years have presented me with.

First of all, I thank my supervisor and co-author Dr. Anders J Johansson for giving me the opportunity to work on this exciting project, first as a master's student and then as a Ph.D. student. His supervision has been characterized from his side by trust in my personal judgment and his enthusiasm and openness to new ideas. His ability to raise his eyes above the details that a Ph.D. student can easily get tangled up in has been an inspiration for me and taught me about the value of never losing sight of the big picture while delving into the details. Through his supervision and personal presence, Anders has made a great contribution to making the last years a pleasant lesson in research and in life in general. Thank you Anders.

I thank my co-supervisor and co-author Prof. Martin Garwicz for always taking the time to provide me with thorough and constructive feedback and encouragement. I thank him for always making my contributions feel valuable and for being a good role-model for a young Ph.D. student starting a career as a researcher. I thank my co-author and NRC's coordinator Prof. Jens Schouenborg for welcoming me in the group and for contributing to the papers with new, interesting perspectives that the engineer in me sometimes needs to be reminded of when writing papers for a mixed audience. Thank you Anders, Martin and Jens for all the conversations we have had during our meetings about the articles. I thank Dr. Fredrik Tufvesson for taking on the role as my main supervisor during the last year and for helping me to prioritize wisely in order to get the work done in time.

I thank my dear colleagues and the administrative and technical staff at the Department of electrical and information technology (EIT) and the Neuronano Research Center (NRC) for creating a pleasant and stimulating environment to work in. My fellow Ph.D. students at both EIT and NRC have made me feel that I am not alone and taught me that the worries and anxiety that sometimes

show up uninvited are just normal "symptoms" of being a Ph.D. student. By always being available and ready to help, the administrative and technical staff at both EIT and NRC have helped to make the life as a Ph.D. student easier. Thank you all for being there and thank you for all the entertaining discussions at coffee breaks, lunches and at casual meetings in the hallways!

I am grateful for the funding that has made this work possible. The work was supported by a Linnaeus Grant from the Swedish Research Council (no. 60012701), a grant from the Knut and Alice Wallenberg Foundation (no. 2004.0119) and the Medical and Engineering Faculties at Lund University.

I thank my loving family, my parents Thorbergur and Sigurborg, my sisters Gunnthorunn and Kristveig and my niece Sara Björt, for their unconditional love and support. By always showing me that they believe in me, they have helped me believe in myself. When times have been tough, they have always helped me to put things into perspective and to turn hopelessness into hope. Takk elsku fjölskylda!

Finally, I am forever thankful to my beloved wife, Charlotte, for always standing by my side, and to our newborn son, Oliver, for bringing a whole new dimension to my life. With her love, patience and ability to see things for what they truly are, Charlotte has kept me reminded of the importance of knowing what truly matters in life. Charlotte and Oliver have given me the invaluable feeling of already having achieved the greatest of all achievements. Besides being a wonderful feeling in itself, this feeling has provided me with extra energy when taking on new challenges, such as writing a Ph.D. thesis. Tack Charlotte och Oliver, ni betyder allt för mig!

Pálmi Þór Þorbergsson, September 27, 2012

Contents

Part I

Overview of the Research Field

Chapter 1

Introduction

1.1 Background

As humans, we receive constant input from the world around us through our senses. The processing of the inputs can lead to various outputs, such as the experience of a mental mode, the retrieval or forming of a memory or the elicitation of a physical reaction. We constantly interact with the world around us by feeling emotions, feeling pain, learning, forgetting, seeing, hearing, speaking and by performing complex movements. But what are the underlying mechanisms that govern all the information processing that constitutes the biological foundations of behavior? Neuroscientists take on the challenge of answering this kind of questions by studying the network dynamics of the immensely complex organ that is the brain.

"Listening to" the brain with the right kind of tools can help us reveal some of the mysteries behind our behavior. By probing the central nervous system (CNS) for its neuronal activities while simultaneously observing behavior, we can make associations and correlations and successively build up models that describe the neural circuits that drive our behavior. Not only do such insights quench our thirst for understanding normal behavior. They also provide us with knowledge that can be used for clinical treatment of neurological disease or for regaining physical functions that have been lost due to neural injury, usually by "talking back to" the nervous system by means of electrical stimulation.

Brain-machine interfaces (BMIs) constitute a class of platforms that provide a uni- or bidirectional connection to the central nervous system, and thereby allow research and treatments as the ones mentioned above. Unidirectional BMIs allow either the measurement of neural activity or the induction of some kind of action, for instance the control of prosthetic devices or a wheelchair.

Bidirectional BMIs communicate with the CNS in both directions by measuring neural activity, extracting information from the measurements and then reacting in a predefined manner. For example, the extracted information can reflect a patient's intended action and the reaction can be in the form of sending a control signal via an actuator unit to an electric wheelchair or a prosthetic limb.

Various types of BMIs exist, varying in their level of invasiveness and the type of signal that they measure [1]. One type of BMIs uses chronically implanted microelectrodes to measure the changes in electric potential that follows the activation of single neurons in the vicinity of the electrodes. Due to both the spatial and temporal resolution that can be captured with such recordings, the amount of information produced by them is vast and the most practical way of conveying the measured data to the outside world is through wires. However, wired connections to chronically implanted measurement devices come with risks of infections due to transcutaneous wires, and they obviously restrict the subject's freedom to move around. Not only are these restrictions disadvantageous from the subject's standpoint, but also from the point of view of answering research questions regarding behavior, since having a wire bundle connected to its head is likely to influence the behavior of the subject. Wireless BMIs ideally resolve these restrictions.

Wireless BMIs come with new challenges which mostly arise due to limitations in wireless link capacity and energy resources. Implanted self-contained measurement devices must have a long lifetime in order to maximize their reliability and minimize the frequency of surgical interventions to replace batteries. Low power consumption is therefore a key feature of the implanted part of a wireless BMI. However, wireless transmission of data is energy consuming. The most straight-forward ways of increasing the capacity of a wireless link are to increase the transmitted power and to increase bandwidth. However, due to the power-constraints on the implant and the heavy utilization of the radio frequency spectrum, both of these approaches are expensive, considering the high data rates provided by multi-channel neural recording devices.

A more feasible way to approach the problem is to decrease the data rate into the wireless link by minimal computational efforts and thereby simultaneously ensuring power efficiency and efficient use of the wireless link capacity at hand. This demands the implementation of energy efficient data reduction or compression techniques on the implant, that make sure that only the relevant information is transmitted over the wireless link. However, discarding data may reduce the quality of the subsequent analysis and thereby it may influence the validity of the conclusions drawn either by the researcher studying behavior or the actuator generating prosthetic control signals. It is therefore essential to consider the performance in extracting information from the neural data

when studying ways of discarding data. Such consideration requires the use of reliable and realistic signal models that allow the output from the analysis procedure to be compared with the true information content of the signal.

This thesis addresses realistic and computationally efficient modeling of the extracellular neural signal, as well as techniques for data reduction for efficient use of computational and wireless resources. Signal models of various complexities and degrees of realism have been implemented, the latest one constituting a new class of models, referred to in this thesis as *compressed compartment models*. The models facilitate computationally and memory efficient synthesis of multi-electrode recordings with realistic properties and are used to study the effects of data reduction on the conclusions that can be drawn from the data analysis.

1.2 Aims

The overall aims of the thesis work have been to:

- Implement a simple wireless brain-machine interface using off-the-shelf components to identify bottlenecks to consider in future designs.
- Establish realistic, controllable and computationally efficient signal models that can be used to synthesize test data for evaluating performance in the analysis of single- and multi channel neurophysiological data.
- Characterize the influence of sampling rate and sampling resolution on the performance in the analysis of neurophysiological data using various algorithms under various circumstances and establish guidelines for the selection of sampling parameters.
- Use signal models to characterize the effects of electrode movements on detected spike waveforms.
- Study how compression of spike waveforms with low-complexity architectures influences performance in the analysis of neurophysiological data.

1.3 Methods at a Glance

Following is a brief summary of the most essential methods applied in the papers of Part II. The methods are described in more detail in the papers and in Chapters 2 to 4.

Hardware (Paper I) was designed and simulated in *NI Multisim* and PCB layout was done in *CadSoft Eagle*. The hardware was built by manual soldering of surface mounted components of sizes down to 0201, i.e. 0.6×0.3 mm.

Programming was done in *Matlab* (Papers I to VII) and *C* (Papers I and II). The simulator in Paper VI was written using object oriented programming in *Matlab*.

Extracellular neural signal models were developed (Papers II, III and VI) and used to provide synthetic test recordings with ground truth. Spike times were generated assuming spontaneous firing with gamma distributed inter-spike intervals (Papers II and VI), bursty firing with gamma distributed inter-burst intervals and Poisson distributed number of spikes per burst (Paper VI) and correlated firing (Paper VI). Neuronal spike waveforms were obtained from an experimentally obtained library of mean spike waveforms from the cat cerebellum (Paper II), by a statistical model involving principal component analysis and Gaussian mixture models (Papers III and IV) and through compressed compartment models obtained through the use of the *NEURON* simulation environment, the line source approximation (LSA), singular value decomposition (SVD) and trivariate polynomial fitting with a multivariate Vandermonde matrix (Papers V, VI and VII). Physiological background noise was modeled as the spiking activity of distant neurons (Papers II to VII) and thermal noise was assumed to be zero-mean Gaussian distributed. Spike libraries and models were validated in terms of spike duration (Papers II and III), distribution of Euclidean distances between spikes, a double blind test for discrimination between real and modeled spikes and sample intensity (Paper III) and comparison of shape and amplitude of true and modeled spikes (Paper VI). Signal models were validated in terms of power spectral density using Welch's method (Papers II, III and VI), sample histogram and level of background noise (Paper VI).

Neurophysiological analysis was performed with various algorithms for spike detection (Papers II, IV and VII), spike alignment (Paper VII), feature extraction (Papers IV to VII), clustering (Papers IV, VI and VII) and automatic spike sorting (Paper II). A method for quantifying performance in spike detection in terms of true and false positive detection rates was described and used (Paper IV). Methods for quantifying performance in spike sorting, in terms of overall and per-neuron clustering accuracies, were described and used (Papers IV, VI and VII). The PCA feature space representation of spikes was examined as a potential predictor for performance in spike sorting (Paper IV).

Electrode movements were modeled by simulating arrays of electrodes following the assumed movement paths and extracting spikes from a time window of given duration while sweeping the window across the recording channels (Paper V). Electrode position was estimated using a linear model involving a

transformation matrix obtained through singular value decomposition (Paper V). Electrode movements in physical space and spike movements in feature space were quantified using a path measure involving the Euclidean distance of points on the path to the mean point on the path (Paper V).

Data reduction was performed by minimizing sampling rate and resolution while sustaining performance in spike detection and sorting (Paper IV) and by compression with fixed generic compression bases in combination with various spike detectors and system architectures (Paper VII).

1.4 Outline

Chapters 2 to 4 provide a general overview of the research field and thereby set the stage for the discussion carried out in the papers of Part II. Chapter 2 starts with a basic presentation of the neuron and the action potential, followed by a discussion about extracellular neural recordings and the essential system components involved in their acquisition and analysis.

Chapter 3 describes various approaches to extracellular signal modeling with the purpose of providing test signals with a priori known characteristics which allows the quantitative assessment of performance in the analysis of neurophysiological data. The models are discussed in terms of their realism, controllability and computational efficiency.

Chapter 4 begins by introducing the wireless BMI as a feasible solution to some of the problems inherent with wired BMIs, such as the risk for post-surgical complications and limited freedom of movement for the subject or the patient. The major challenges involved in the design of wireless BMIs are presented in relation to theoretical limits for the information transfer capacity of the wireless channel and the limited energy supply that typically characterizes self-contained implants. The problem is discussed both from the perspective of the wireless system and the neural data that is to be transmitted, in terms of capacity and data reduction techniques, respectively. The second part of the chapter addresses data reduction in more detail, focusing on simple approaches to efficiently utilizing the capacity of the wireless link at hand.

In Chapter 5 the results and contributions of the papers are summarized and the papers are discussed within the context of the field as presented in Chapters 2 to 4. The chapter concludes with a general discussion about the results and possible future work.

Chapter 2

Extracellular Neural Recordings

Neurons are the basic signaling elements of the central nervous system (CNS) and they communicate with each other through interfaces called *synapses*. The neurons on the transmitting and receiving sides of the synapse are referred to as the *presynaptic* and *postsynaptic* neurons, respectively. A network of neurons that processes information is in many ways similar to a circuit of logic elements that constantly evaluate the sum of all input signals from all presynaptic neurons. If the gathered inputs from the network to a given neuron add up sufficiently, the neuron gets activated and transmits the signal to downstream neurons. Since the signaling is electrochemical in nature, governed by the flow of ions across the neuronal membranes, the activation pattern of a neuron can be monitored from its outside by measuring the change in electric potential associated with its activation. These types of measurements are referred to as *extracellular neural recordings* and they generally allow the activities of multiple individual neurons to be assessed through one or more measurement channels. By monitoring multiple neurons in a circuit that is involved in a behavioral task, it is possible to characterize the major circuit components and build models that explain the neuronal activities underlying the behavior.

In this chapter, the neuron is introduced from a cytological and electrophysiological perspective and the action potential is introduced as the basic signaling unit in neuronal circuits [2, 3]. The major steps involved in acquiring and and analyzing the extracellular signal are then discussed, assuming that the task is to characterize either the compound activities of a group of

neurons (*multi-unit activity*) or the activities of individual neurons (*single-unit activity*).

2.1 Neurophysiological Background

2.1.1 The Neuron

Neurons and glia cells are the building blocks of the CNS. In principle, the information processing carried out by the CNS is carried out by neurons. The basic signal unit for most neurons is the action potential, a rapid all-or-nothing change of the neuronal membrane potential. The total number of neurons in the brain is in the order of 10^{11} and despite the existence of a large number of neuronal types, they all share the same basic architecture and their common role is to convey and process the information that governs our behavior.

A typical neuron can be roughly divided into four functional regions – *soma*, *dendrites*, *axon* and *presynaptic terminals* (see Fig. 2.1). The soma (or cell body) contains the cell nucleus and is the neuron's center for metabolic functions and protein synthesis. The dendrites serve as the input terminals of the neuron – receiving incoming signals from presynaptic neurons. The axon starts at the base of the soma – at the *axon hillock* – and serves as a pathway for the signal from the neuron to reach postsynaptic neurons. The axon hillock is where the input signals to the neuron are integrated and if the total input exceeds a certain threshold, an action potential is generated and propagates along the axon to reach other neurons. This function of the axon hillock depends on its high density of voltage gated sodium channels (see following section). The axon can be myelinated, i.e. wrapped within a sheath of isolating *oligodenrocytes*, which is one type of glial cells found within the nervous system. The sections of the myelin sheath are separated by the *nodes of Ranvier*, at which the action potential is regenerated, providing an efficient way of transmitting the signal a long way effectively without being attenuated. Presynaptic terminals are where the neuron terminates and connects to postsynaptic neurons. Synapses are the connection terminals between neurons across which signaling substances flow and allow the signal from one neuron to influence the state of another neuron.

2.1.2 The Membrane Potential

As touched upon already, the state of the neuron is reflected in its *membrane potential*, i.e. the potential difference between the inside and outside of the cell

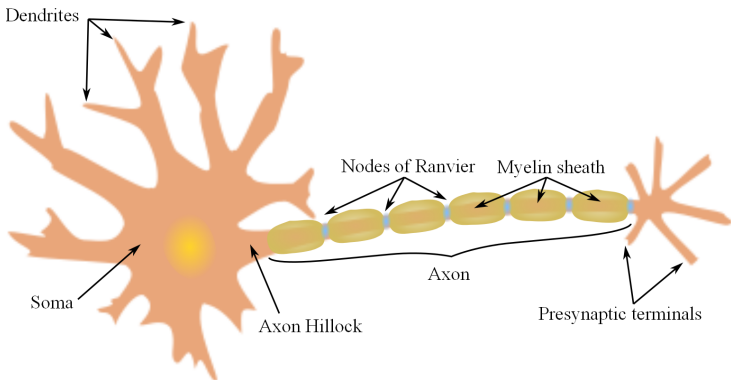


Figure 2.1: An illustration of the building blocks of a typical neuron (partially adopted from http://www.openclipart.com)

membrane. The membrane potential V_m is thus given by

$$V_m = V_{in} - V_{out} \tag{2.1}$$

where V_{in} and V_{out} are the electric potentials at the inside and outside of the cell membrane respectively.

While the action potential, generated in the axon hillock and traveling down the axon, is in principle an all-or-nothing phenomenon, the sub-threshold membrane potential of the cell soma or dendrites can be modulated in a graded fashion. The membrane potential depends on differences in concentrations of ions on either side of the membrane, which are maintained by energy consuming ion pumps located in the membrane and pumping the ions against their concentration gradients. Membrane current is generated as the ions flow through passive ion channels, thereby displacing electrical charges across the membrane.

The most significant ions in determining the neuronal membrane potential are sodium (Na^+), potassium (K^+) and chlorine (Cl^-). At any given time, their *equilibrium potentials*, together with the membrane's permeability to the respective ion, determine the *resting potential* of the cell, i.e. the membrane potential in the resting – or inactive – neuron. The equilibrium potential of a given ion X can be calculated by Nernst's equation as

$$E_X = \frac{RT}{zF} \ln \frac{[X]_o}{[X]_i} \tag{2.2}$$

where R is the gas constant, T is the temperature (in Kelvin), z is the valence of the ion, F is the Faraday constant and $[X]_{i/o}$ is the intra-/extracellular concentration of the ion. Under steady-state conditions, i.e. when the membrane

potential V_m is not changing, it can be calculated according to Goldman's equation as

$$V_m = \frac{RT}{F} \ln \frac{P_{K^+}[K^+]_o + P_{Na^+}[Na^+]_o + P_{Cl^-}[Cl^-]_i}{P_{K^+}[K^+]_i + P_{Na^+}[Na^+]_i + P_{Cl^-}[Cl^-]_o} \tag{2.3}$$

where P_X is the membrane permeability to ion X (in cm/s). Thus, when the membrane permeability of one ion dominates over those of other ions, Goldman's equation approaches Nernst's equation for that ion and the membrane potential approaches the equilibrium potential for that ion.

In the resting neuron, the net ion flow across the membrane is at equilibrium and the membrane potential is given by the neuron's resting membrane potential (usually between -50 and -90 mV). At rest, there is an influx of Na^+ ions and an outward flux of K^+ ions. When this equilibrium is disturbed, the membrane becomes either depolarized or hyperpolarized, i.e. the membrane potential becomes either less or more negative.

It is through locally altering the neuronal membrane properties and thereby the permeability to different ions that neurons are able to communicate with each other. When a neuron receives input from a presynaptic neuron, the (postsynaptic) neuron's membrane potential is changed locally, usually at the dendrites and this change then spreads towards the cell soma and attenuates passively along the cell membrane. The membrane's response to this change depends on the membrane capacitance, the membrane input resistance and the axial resistance of the dendrite(s).

A single neuron can receive excitatory and inhibitory inputs from several thousands of presynaptic neurons. Excitatory and inhibitory inputs strive towards depolarizing and hyperpolarizing the membrane, respectively, and thus counteract each other. When the sum of these inputs leads to a depolarization of the membrane beyond a given threshold, the cell gets activated and an action potential is generated.

2.1.3 The Action Potential

In 1952 Alan Lloyd Hodgkin and Andrew Fielding Huxley explained in detail the interaction between ion permeabilities, ion flows and the membrane potential. Using their models they provided a thorough mathematical description of the generation of action potentials [4, 5].

If the inputs from all presynaptic neurons sum up to depolarize the membrane potential of the postsynaptic cell beyond a given threshold, an action potential is generated (see Fig. 2.2). Voltage gated Na^+ channels open when the threshold is reached, giving rise to an influx of Na^+ ions that is larger than the outflux of K^+ ions. This altered flux of ions continues to further depolarize

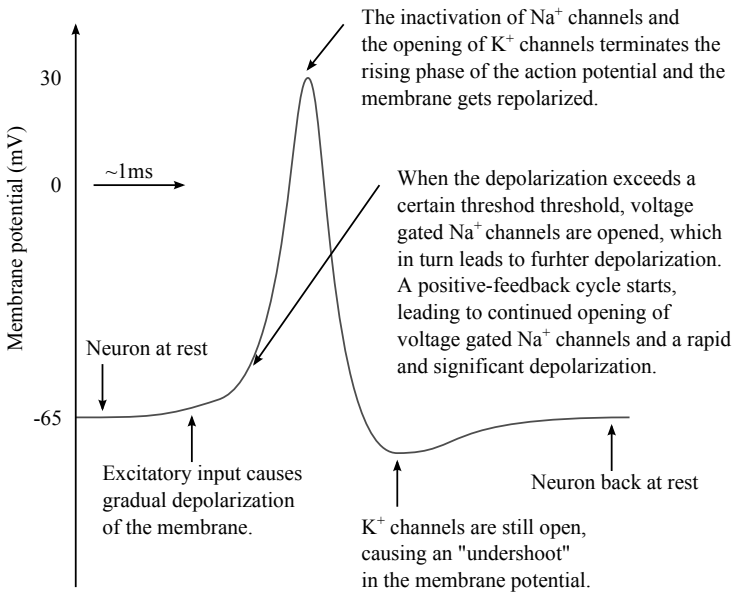


Figure 2.2: An illustration of the action potential and the underlying mechanisms.

the membrane, which in turn leads to the opening of yet more voltage gated Na^+ channels. This positive-feedback procedure continues to rapidly drive the membrane potential towards the equilibrium potential of Na^+, which is in the range of some tens of millivolts. Finally, when the peak of the action potential is reached, the voltage gated Na^+ channels start to close by inactivation and voltage gated K^+ channels start to open. Thus the Na^+ influx decreases and a K^+ influx begins, resulting in the repolarization of the membrane. Before the K^+ ion channels close, the membrane potential usually gets briefly polarized beyond the resting membrane potential.

During a short time period (a few milliseconds) after the termination of the action potential, there is a residual inactivation of Na^+ channels and opening of K^+ channels. This short time period is referred to as the *refractory period*. The refractory period can be divided into two phases – the *absolute* and the *relative* refractory periods. During the absolute refractory period, the neuron cannot be activated and during the relative refractory period, the activation threshold is higher than when the cell is at equilibrium and thus a stronger stimulus is required for activation. The refractory period thus gives rise to an upper limit to the firing rate of the neuron.

2.2 Recording and Analysis of the Extracellular Neural Signal

Extracellular neural recordings have become important tools for providing insight into the neuronal circuits that govern our behavior as well as to provide means of extracting control signals for prosthetic control [6, 7, 8, 9, 10]. In contrast to intracellular measurements, which directly measure the membrane potential of a single neuron, extracellular neuronal recordings measure the electric potential within the extracellular space, resulting from the flow of ions across the cell membranes of a large number of neurons simultaneously. The extracellular signal is measured with an implanted *microelectrode*, consisting of one or more *electrode sites*.

The extracellular signal consists of several components. The first of these, which is the one addressed throughout this thesis, is the *spiking* component that reflects the action potentials elicited in neurons that are close to the recording electrode, typically within a distance of approximately 50 μm [8]. The shape of the extracellular spike depends on the morphology of the neuron and the spatial relationship between the neuron and the electrode [11, 12]. This dependency is due to the spatial distribution of membrane current sources as seen from the electrode site, and it allows the detected spikes from different neurons to be classified and assigned to their neurons of origin through a procedure referred to as *spike sorting* [13, 14]. By this procedure, the spike trains from individual neurons can be separated and the firing characteristics of the neurons can be characterized individually.

Other components of the extracellular signal are physiological noise, representing spiking activity from distant neurons, and low-frequency local field potentials (LFPs) that are believed to represent synaptic input to neurons close to the recording electrode [15, 3]. In contrast to the spiking component, most of the LFP energy resides in the lower part of the frequency spectrum, making it straightforward to remove from the recorded signal by means of high-pass filtering and thereby isolating the spiking component. Since the physiological noise component is composed of spiking activity of distant cells, it cannot be removed from the recording by simple filtering without influencing the spiking component as well.

In addition to the above components of the extracellular signal, which all stem from physiological processes, are those that originate from the measurement system or it's electrical environment. The most significant of those are thermal noise generated in the analog front end and 50/60 Hz powerline interference. Thermal noise can be minimized by considering it during the design of electrodes and amplifying circuits and if present, powerline interference can

be removed by digital filtering.

Figure 2.3 illustrates the general structure of brain-machine interface that uses extracellular neuronal recordings to extract information about the activities of multiple or single neurons – the multi- or single unit activity. Figure 2.4 illustrates the procedure of isolating the spike trains from the individual neurons through spike detection, alignment, feature extraction and clustering. In the following sections, we introduce the essential components of the system and briefly discuss the major design considerations.

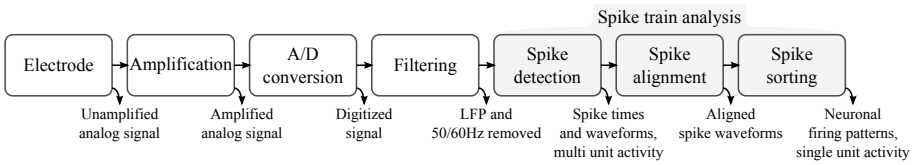


Figure 2.3: An illustration of the building blocks of a typical brain-machine interface that uses extracellular neuronal recordings to characterize multi- or single unit activity.

2.2.1 Electrode

Traditional chronically implanted electrodes for extracellular neuronal recordings can be roughly categorized into wire electrodes and semiconductor based electrodes [16]. Wire electrodes are the oldest type of electrodes used in this context and they consist of thin insulated wires whose implanted tip is either cut off or etched in order to expose the conductor to the tissue. Semiconductor based electrodes are manufactured on semiconductor substrates and can be designed to have a variety of geometries. They also offer the possibility of directly manufacturing the front end electronics on top of the electrode array. Examples of such electrodes are the well known Utah and Michigan arrays that have numerous electrode sites arranged on a planar and linear array, respectively. A recent type of electrodes has been developed at Neuronano Research Center (NRC), specifically designed to target the neuronal layers of the cerebellum [17]. The NRC electrode is based on a polymer foil, with platinum coated gold electrode sites and offers increased mechanical flexibility and the manufacturing procedure can be adapted to manufacture electrodes to target specific brain structures.

Several factors influence the recording quality for a given electrode. In [18], the authors concluded that the recording SNR was mainly determined by the digital filtering, the impedance of the electrode-tissue interface, neuronal

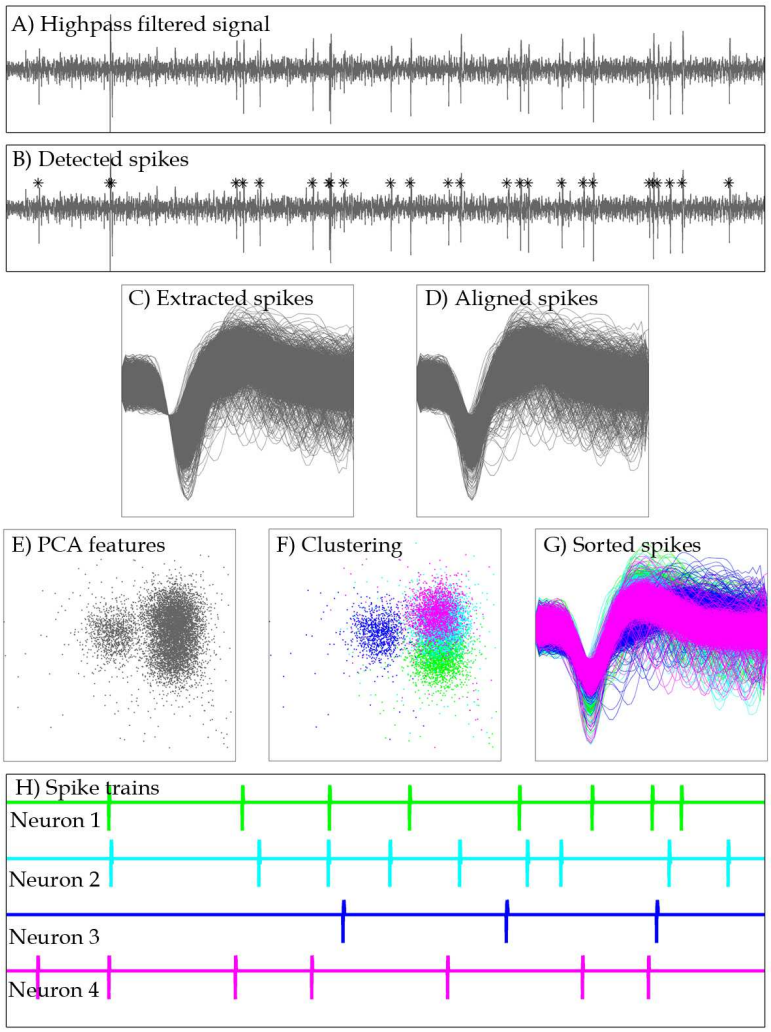


Figure 2.4: An illustration of the procedure of isolating spike trains from individual neurons in the highpass filtered signal. The procedure typically involves spike detection, spike alignment, feature extraction and clustering. In the example shown here, absolute value threshold detection, maximum absolute value alignment and principal component analysis are used for spike detection, alignment and feature extraction, respectively.

density and firing rates. They concluded that the size of electrode sites had little impact on the SNR.

2.2.2 Amplification

Due to the small amplitude of the recorded neural signal (typically tens to hundreds of microvolts), amplification is needed at the input of the neural recordings system. Some of the primary design factors to consider are gain, input offset, noise and power consumption [19, 20]. Gain and input offset need to be set to match the signal amplitude with the dynamic range of the subsequent block (usually the A/D converter). Noise should be minimized since that is a significantly limiting factor in the analysis of the recorded signal. Maximizing the input impedance is important in order to minimize the amplifier's electrical loading of the electrode.

In Paper I, a four-channel amplifier with variable gain was designed and simulated in *National Instruments Multisim* and the PCB layout was designed using *CadSoft Eagle*. With the A/D converter's dynamic range of 1.22 V in mind, the gain was set to be variable between 55 and 70 dB, thus allowing full scale amplification of spikes with peak-to-peak amplitudes between 270 μV and 1.5 mV.

2.2.3 A/D Conversion

The A/D converter digitizes the amplified signal at a sampling rate and resolution that ideally provide an adequate representation of the signal components for subsequent data analysis to be carried out reliably. Correct dimensioning of the A/D converter is especially important in wireless BMIs since excessive data acquisition increases the demand for computational resources and the data rate into the wireless link, both of which increase the power consumption of the implanted part of the BMI.

The sampling parameters may be selected either by considering the frequency contents of the extracellular signal [13, 21] or by minimization with regard to a performance measure [22, 23]. However, the estimation of signal bandwidth, and thereby the Nyquist rate, depends on subjective selection of an upper bound for the frequencies that represent relevant signal components. Thus, from the point of view of reproducability, an approach that only relates to the objective estimation of accuracy at the system's output, as a function of the sampling parameters employed, is likely to be a superior choice.

The influence of sampling rate and resolution on the performance in the analysis of neurophysiological data (spike detection and spike sorting) at various noise levels and numbers of target units was addressed in Paper IV. The

simulator reported in Paper II in combination with the statistically modeled spike libraries reported in Paper III was used to provide 150 test recordings – 50 recordings containing each number of target neurons, 1, 2 and 4 respectively. The performance curves were analyzed and sampling rate and resolution breakpoints for each processing task were taken as the sampling rate and resolution at which the accuracy fell beyond 1% or 5% below the maximum achievable accuracy in each case. Spike detection accuracy was estimated in terms of true and false positive detection rates and spike sorting accuracy was estimated in terms of true positive classification rate (see Sections 2.2.5 and 2.2.7). Spike detection with absolute value threshold detection and spike sorting with principal component analysis and fuzzy c-means required a sampling rate and resolution of 16 or 31 kHz (5% or 1% error tolerance) at 9 bits and 5 kHz at 5 bits, respectively. Besides providing guidelines for minimizing sampling rate and resolution while sustaining performance, the results provided an overall characterization of the relationships between performance, sampling parameters, noise level and number of target units for various algorithms for spike detection and spike sorting.

2.2.4 Filtering

Highpass filtering is usually applied to the sampled signal in order to remove the low-frequency LFP from the higher-frequency spiking component (see Figure 2.4 A). This filtering can be performed in various ways, but should be considered carefully when comparing results from different studies since it influences the shape of spikes [18]. The spiking component is typically isolated by bandpass filtering the signal from around 300 Hz to 5 kHz.

2.2.5 Spike Detection

Spike detection is the task of detecting (and extracting) spike waveforms in the recording and it is usually based on applying a threshold to a pre-processed version of the signal. The output of the spike detector therefore constitutes a series of timestamps indicating the occurrences of action potentials in neurons close to the electrode, referred to as multi-unit activity (see Figure 2.4 B), and extracted spike waveforms (see Figure 2.4 C). Although located after the A/D converter block in the system architecture presented here, spike detection can also be carried out in the analog domain [24]. In this thesis, we assume spike detection to be performed in the digital domain.

In Paper IV, we compared the performances of several spike detection algorithms while varying sampling rate, resolution, noise level and number of target neurons. The algorithms that were implemented and compared were

ABSolute value threshold detection (ABS) [25], *Nonlinear Energy Operator detection* (NEO) [26], *Stationary Wavelet Transform Product Detection* (SWTP) [27] and *Matched Filter detection* (MF) [28].

In Paper VII, we compared the performance in spike reconstruction and spike sorting when preceded by spike detection and compression with the ABS and NEO detectors and various compression bases (see Chapter 4). In Paper VII, spike detection was simulated by first using known spike times for extracting spikes from the synthetic signals, provided by the simulator described in Paper VI, and then discarding spikes that would not pass the detection threshold or that would occur during the recovery period of the detector[1]. Spike detection jitter (see Section 2.2.6) was introduced afterwards using discrete time delay filters. This procedure was chosen in order to ensure that false positive detections would not influence the quantification of performance in spike reconstruction and sorting.

The following sections briefly describe the spike detectors that have been implemented and used in the papers. The descriptions are followed by a review of a method for estimating spike detection accuracy, introduced in Paper IV.

Absolute Value Threshold Detection (ABS)

ABS is the simplest form of spike detection, in which a detection occurs whenever the absolute value of the signal exceeds a given threshold. The threshold is usually set as a multiple of an estimate of the background noise level, for instance as

$$T = 4\hat{\sigma}_N = 4 \cdot \text{median}\left\{\frac{|v|}{0.6745}\right\} \tag{2.4}$$

where $\hat{\sigma}_N$ is the estimated standard deviation of background noise and $|v|$ is the absolute value of the digitized signal's amplitude [25].

Nonlinear Energy Operator Detection (NEO)

In NEO detection a threshold is applied to the nonlinear energy operator (NEO) of the signal. The NEO estimates the instantaneous energy of the highpass filtered signal and therefore emphasizes high-amplitude, high-frequency spikes. The NEO $\Psi(n)$ of the signal $v(n)$ is given by

$$\Psi(n) = v^2(n) - v(n+1) \cdot v(n-1). \tag{2.5}$$

[1] A spike detector's recovery period refers to the time period after threshold crossing during which spike detection can not occur and is introduced in order to avoid multiple detections of single spikes.

and the threshold T is set as

$$T = C\overline{\Psi(n)} \tag{2.6}$$

where $\overline{\Psi(n)}$ is the mean of the $\Psi(n)$ and C is a scaling factor adjusted empirically and then used as a constant [26]. C can be found by optimizing the ratio between true and false positive detection rates when performing spike detection on synthetic recordings with known spike times.

Stationary Wavelet Transform Product Detection (SWTP)

The stationary wavelet transform (SWT) of the signal $v(n)$ is calculated at 5 consecutive dyadic scales ($W(2^j, n)$, $j \in [1, 5]$). The scale with the maximum sum of absolute values is found ($2^{j_{max}}$) and the point-wise product $P(n)$ of wavelet coefficients over three consecutive scales up to $2^{j_{max}}$ is calculated as

$$P(n) = \prod_{j=j_{max-2}}^{j_{max}} |W(2^j, n)|. \tag{2.7}$$

$P(n)$ is then smoothed by convolution with a Bartlett window $w(n)$ (half the spike length) and a threshold T is applied to the smoothed $P_s(n)$. T is set as

$$T = C\overline{P_s(n)} \tag{2.8}$$

where C is a scaling factor and $\overline{P_s(n)}$ is the mean of $P_s(n)$ [27]. As for NEO detection, the scaling factor C can be set empirically by maximizing the ratio between true and false positive detection rates.

Matched Filter Detection

In MF detection, the signal is convolved with a template spike waveform that can, for instance, be taken as the mean of detected waveforms during an initial training period. A threshold T is applied to the absolute value of the resulting signal. T is set to a percentage of the input range of the (convolved) signal and can be optimized in the same way as C for NEO and SWTP.

Estimation of Spike Detection Accuracy

In Paper IV, we presented a method for estimating spike detection accuracy, based on the comparison of detected spike times and true spike time frames according to the ground truth provided with the test signal. True spike time frames were determined by the true onsets and durations of spikes. Based

on this comparison, detection scores were assigned to true spikes depending on the number of true spike time frames covering each detected spike time. Special cases, for instance where multiple detections occurred during a single true spike time frame, were also accounted for. Based on these scores and the number of false positive detections, the true and false positive detection rates were calculated according to

$$\hat{P}_{TP} = \frac{\sum_{m=1}^{M} d_m}{M} \cdot 100\% \tag{2.9}$$

and

$$\hat{P}_{FP} = \frac{N_{FP}}{N_{ns}/N_{recovery}} \cdot 100\%, \tag{2.10}$$

respectively, where M is the number of true spikes and d_m is the detection score assigned to the mth true spike, N_{FP} is the number of false positive detections, N_{ns} is the number of samples in the recording not covered by a true spike time frame and $N_{recovery}$ is the recovery time of the spike detector (in samples).

2.2.6 Spike Alignment

Depending on the spike detector employed, spike alignment may be needed to ensure maximum accuracy in spike sorting. Due to noise and asynchronicity between the sampling of the signal and the firing of action potentials, detected spikes from a given neuron are typically not sampled at the same time instances within the noise-free spike waveform. Therefore, the spike detection threshold is typically not crossed at the same relative time instance between spikes and once extracted, spike waveforms are misaligned to one another. This phenomenon is referred to as *spike detection jitter* and is disadvantageous in spike sorting since it introduces an apparent deviation in shape between the spike waveforms from a given neuron (see Figure 2.4 C).

Spike detection jitter is removed by spike alignment by identifying the location of a reference point (for instance a maximum value or center of mass [3]) and then shifting the detected waveforms to have that reference point occur at the same absolute point in time for all waveforms (see Figure 2.4 D). Since the true reference point often does not coincide with a specific sample within the spike waveforms, the alignment can be preceded by upsampling and proceeded by downsampling to the original sampling rate. Single-point based alignment is usually more sensitive to noise than approaches that consider the entire waveform, such as the center-of-mass approach [3], but is computationally less demanding.

In Paper IV, the influence of spike detection jitter, introduced by asynchronous sampling and firing, on the accuracy in spike sorting was quantified

and found to be insignificant at sampling rates above 11.5 kHz when sorting spikes with principal component analysis and fuzzy c-means (see following section). In Paper VII, maximum value alignment was implemented and applied prior to spike sorting. First, the absolute maximum value of the detected peak or valley was identified and the waveform was then shifted in time by a discrete time delay filter to have the absolute maximum value occur at a given point in time. This procedure was implemented with a variable upsampling factor to facilitate a more precise alignment.

2.2.7 Spike Sorting

Spike sorting is the task of discriminating between spikes coming from different neurons and thereby, in combination with the detected spike times, establishing an estimation of the individual neuronal spike trains (see Figure 2.4 H). The combined output from spike detection and spike sorting is the single unit activity. In its most common form, spike sorting involves two steps – feature extraction and clustering. However, there are algorithms that combine the spike detection and spike sorting steps and do not explicitly include the feature extraction step. An example of such an algorithm, that employs linear filters to optimize the signal to noise ratio of the recordings and then uses a procedure called "deconfusion" to separate the activities of the contributing neurons, is given in [29]. In this thesis, focus is on algorithms where spike detection, feature extraction and clustering are carried out separately.

An important aspect to consider in relation to spike sorting is the algorithm's ability to identify neurons in recordings that are separated in time. Due to electrode movements, spikes coming from a given neuron at a given recording instance might differ from the spike coming from the same neuron at another recording instance. The task of pairing spikes from separate recording instances as coming from the same neurons is sometimes referred to as *spike tracking* and has been address in e.g. [30]. In Paper VI, a method for realistically simulating the changes in spike shapes followed by varying the electrode's position was introduced (see Chapter 3). In Paper V, this method was used to provide test signals for studying the relationship between electrode movements and the feature space representation of spikes and a linear model was used to estimate the electrode position directly from the features of detected spikes.

The following sections briefly describe the feature extraction and clustering algorithms that have been used in the papers. The descriptions are followed by a brief review of the methods for estimating spike sorting accuracy, used in Papers IV, VI and VII.

Feature Extraction

Feature extraction is the step of extracting characteristic features from the detected spike waveforms that deviate between the spikes coming from different neurons (see Figure 2.4 E). Various approaches exist, varying in complexity and applicability to different cases. Here we consider a few feature extraction methods whose performances have been well characterized. Those are *Principal Component Analysis* (PCA) and the *Discrete Wavelet Transform* (DWT) [14, 25]. PCA was used for spike sorting in Papers IV, VI and VII and the DWT was used in combination with two different feature selection methods in Paper IV. PCA has been shown to perform better than the DWT when the wavelet basis is badly tuned to the data and when the discriminative features of the waveforms are encountered on a large time scale [31].

Principal Component Analysis:
In PCA, an ordered set of orthonormal basis waveforms (principal components), that describe the variation in the set of spike waveforms presented to the algorithm, is found. The spike waveforms can then be completely described as linear combinations of the basis waveforms and the basis waveforms weights can be used as spike features in spike sorting. Since the set of basis waveforms is ordered by significance – the first waveform describing the most variation – the first N weights are selected as spike features. N is often set to 2 or 3, but this choice is often made since it provides natural means of visualizing the feature space representation of spikes.

Discrete Wavelet Transform:
The wavelet transform of each spike is calculated using, for instance, Haar wavelets. The wavelet coefficients then describe the spike waveforms at various scales and times [25]. The number of wavelet coefficients is the same as the number of samples in the spike waveforms. In contrast to PCA, the coefficients are not ordered according to significance and the set of coefficients used in clustering needs to be identified. Thus, every coefficient distribution needs to be tested and selecting the coefficients that provide the best cluster separation is based on the obtained test statistics. Selecting coefficients with multimodal distributions is beneficial since good mode separation indicates good cluster separation. In Paper IV, we implemented and used the *maximum-difference test* [32] and a modified version of the *Lilliefors test for normality* [25]. In the modified Lilliefors test, a measure for mode separation is included in the overall test statistic and the features with the three highest test statistics are selected.

Clustering

Clustering is the task of grouping together spikes with similar features, and thereby assigning spikes to their neurons of origin (see Figure 2.4 F and G). The clustering algorithms used in the papers are *K-means* (Papers VI and VII) and *Fuzzy C-means* (Paper IV) [33]. Both of these algorithms require that the number of clusters is known a priori, which it seldom is in reality. However, since the work accounted for in the papers in all cases involved comparing the performances of the algorithms while varying some external properties, such as sampling rate, noise level etc., the true number of clusters was provided as input for the sake of consistency. The true number of clusters was known from the ground truth provided with the synthetic test recordings.

In K-means, K clusters are assumed to be present and the points in feature space are successively assigned to clusters based on their distances to the mean point in each cluster. The cluster means are updated as points are assigned to them. Fuzzy C-means is similar to K-means, but instead of returning cluster identities for each point as K-means does, C-means returns the probabilities of each point belonging to each cluster.

Estimation of Spike Sorting Accuracy

Spike sorting accuracy can be estimated by comparing assigned neuronal identities with the true identities provided by the ground truth when processing test data. The accuracy can be estimated for either the entire set of spikes as an overall measure (Paper IV, VI and VII), or for the individual neurons (Paper VI). Since the cluster identities of true and detected clusters are not necessarily the same, measures need to be taken in order to link together true and detected clusters.

In Paper IV, a method for estimating overall clustering accuracy was introduced and used. First, an evidence matrix E was constructed, where the entries $e_{i,j}$ corresponded to the number of spikes truly belonging to cluster j that were assigned to cluster i. By hierarchical examination of the evidence matrix, the most likely cluster mappings were successively revealed and the true positive classification rate was calculated as the total percentage of spikes assigned to the correct cluster.

In Paper VI, a similar method was used, but instead the entries $c_{i,j}$ of the evidence matrix C corresponded to the pairwise linear correlation coefficients between the mean spike waveforms of true cluster i and detected cluster j. Having revealed the cluster mappings from the evidence matrix, the *leakage matrix* L was constructed where element $l_{i,j}$ corresponded to the number of spikes truly belonging to neuron i classified as belonging to neuron j. Thus, the entries of the leakage matrix provided direct information about the number

of correct and false classifications on a per-neuron basis, allowing the clustering results to be evaluated both for each neuron and for the gathered assembly of neurons. This procedure was not accounted for in detail in Paper VI since the purpose of the analysis carried out in the paper was only to demonstrate the usability of the modeling method presented in the paper.

Chapter 3

Extracellular Neural Signal Modeling

As discussed in Chapter 2, the main components of the extracellular signal are (1) the spiking component, representing neuronal activity in the vicinity of the recording electrode, (2) local field potentials, representing synaptic input into the region and (3) physiological noise, representing spiking activity from distant neurons. When developing and testing algorithms for extracting information from such recordings, the access to test data with a priori known characteristics is of great importance [34]. Since the true neuronal activities that contribute to real recordings are usually not known or difficult to characterize completely, synthetic model based test recordings are a feasible alternative.

Realism, controllability and computational efficiency are attractive properties of extracellular signal models for test signal generation. Realism can refer to neuronal firing patterns, spike amplitude and spike shape and it increases the level to which the model mimics real-life scenarios that arise once the system is implemented. Realism is thus a highly important property since it increases the likeliness that the estimated algorithm performance reflects the true performance after implementation in a future application. The controllability of a model refers to, for instance, to which extent the firing properties of neurons can be controlled by the experimenter and the degree of freedom in setting up the recording geometry. It might also refer to the level of detail at which ground truth information can be accessed. Controllability allows system performance to be evaluated in a wide range of scenarios and increases the applicability of the results of a performance evaluation. Last but not least, computational efficiency allows quick access to simulation data and it therefore aids in reducing

the time spent testing the algorithms.

This chapter begins by describing the approach to signal generation used in the models addressed in this thesis. Then, a brief review of models to generate neuronal firing patterns is given. Finally, the most common approaches for obtaining test signals are divided into categories according to signal origin and the different models are discussed in terms of realism, controllability and computational efficiency.

3.1 Signal Construction

In the signal models implemented in Papers II and VI, the signal is assembled as the sum of spike trains from all neurons contributing to the recording. For each neuron, a spike waveform is determined and random spike times are generated using the neuron's firing model (see following section). The neuron's spike train $v_n(t)$ is then generated according to

$$v_n(t) = \sum_{m_n=1}^{M_n} s_n(t - \tau_{n,m}) \tag{3.1}$$

where $\tau_{n,m}$ is the mth spike time of neuron n, M_n is the total number of spikes that the nth neuron fires during the recording and $s_n(t)$ is the pre-calculated spike waveform of the nth neuron. The total signal is then given by the sum of spike trains from all N neurons, i.e.

$$v(t) = \sum_{n=1}^{N} v_n(t). \tag{3.2}$$

In the case multiple electrode sites are simulated (Paper VI), the signal is constructed in this way for each electrode site. Other signal components, such as the LFP, thermal noise and powerline interference are generated separately and then added to the signal. Apart from simultaneous intra- and extracellular recordings, the models discussed in the following sections can all be used in combination with this signal construction procedure.

3.2 Firing Models

Firing models are statistical models that are used to generate spike trains or firing patterns for the neurons that contribute to the recording. Although they may not be suitable for studying the dynamics of neuronal networks [35], such statistical models are very practical from the point of view of providing

realistic test data for characterizing system performance. Three such models are briefly discussed here, all of which have been implemented and used in the work reported in the papers.

3.2.1 Spontaneous Firing

The simplest firing model assumes no functional relations between the neurons in the recording and all neurons are assumed to fire spikes at random times. The generation of spike times can be modeled, for instance, by a Poisson process or a renewal process [36]. The distributions of *inter-spike intervals* (ISIs) for the Poisson and renewal processes can then be modeled as exponential and gamma distributions, respectively. Due to its larger number of degrees of freedom, the renewal process has the advantage of being able to model both the absolute and relative refractory periods of the neuron (see Section 2.1.3). This can be solved to some extent for the Poisson process by using a truncated exponential function for the ISI generation. However, this only captures the absolute refractory period.

In Papers II and VI, models for generating spike times with gamma distributed ISIs were implemented and used. The gamma distribution is characterized by it's shape and scale factors, k and θ, respectively. The shape factor relates to how narrow the distribution is – a small shape factor indicating a narrow distribution. The scale factor relates to the location of the distribution and is thus related to the mean value. The mean of the distribution is given by the product of the shape and scale factors. In the papers, we used a model description based on a given shape factor k and the mean ISI which is given by the inverse of the mean firing rate $\bar{f}$, i.e.

$$ISI \sim \Gamma\left(k, \frac{1}{\bar{f}k}\right). \tag{3.3}$$

Generating random spike times for a given neuron with a given ISI distribution was performed in three steps. First, the total number of spike times to be generated was assumed to be roughly given by the duration of the recording times the mean firing rate of the neuron. Second, the corresponding number of random ISIs was drawn from the ISI distribution. Finally, the ISIs were integrated to obtain the spike times.

3.2.2 Bursting Neurons

The phenomenon of bursting refers to when the spikes from a neuron arrive in clusters of e.g. 2 to 20 spikes. The bursts are separated by *inter-burst intervals* (IBIs) of up to 30 seconds and during the time of a burst, the spikes are fired

at regular or sometimes unusually short inter-spike intervals. The ISI usually increases during the burst and the spikes successively decrease in amplitude, which is likely to be caused by decreased Na^+ conductance resulting from the sustained depolarization [37, 38].

In the simulator described in Paper VI, a model for simulating bursting neurons was implemented. IBIs were assumed to be gamma distributed and the distribution was characterized by a mean burst rate and a shape parameter, and the number of spikes per burst was assumed to follow a Poisson distribution, characterized by a mean value [36]. Spike times were then generated in three steps. First, burst times were generated in the exact same way as spike times for the spontaneous firing models described in the previous section. Second, a random number of spikes was assigned to each burst. Third, spike times were generated within each burst by adding up a given fixed ISI plus a small normally distributed random shift. The third step was implemented in this way for the sake of simplification. Also, for the sake of simplification, spike amplitude was kept fixed within bursts.

3.2.3 Correlated Neuronal Activities

Statistical models can be useful for generating test data that simulates network activity to some extent. Such test data extend the challenge for the system under test from only having to extract firing patterns of individual neurons to that of also unraveling the connections between them. In Paper VI, we included the implementation of the method described in [39] provided by the authors of that paper, to generate spike times for neurons whose activities were assumed to be correlated to a given extent, specified by a covariance matrix.

3.3 Simultaneous Intra- and Extracellular Recordings

Although this class of test signals would perhaps fit better into the context of test signal *acquisition* rather than *generation*, we include it here since it is used within the same context and represents an important class of test signals [40, 29]. It was previously mentioned that the major problem with using real recordings as test signals is the absence or incompleteness of a priori knowledge about the true information content in the signal. However, by simultaneously performing high SNR intracellular measurements on the neurons contributing to the recorded signal, this problem is overcome to some extent.

In terms of realism, this class of test signals is highly advantageous, the signals being real in nature. However, they lack in terms of controllability

since the possibilities of modifying the recording properties are very limited. Also, despite the presence of the high SNR intracellular recordings, they may not capture all of the true neuronal activity represented in the extracellular recording and thus, the reliability of the ground truth may be limited. The concept of computational complexity does not apply here. Due to their realism and despite the limitations mentioned here, simultaneous intra- and extracellular recordings may be suitable as benchmark signals to use at later stages of algorithm testing.

3.4 Mathematical Models

This class of models is based on mathematical descriptions of the origin of the extracellular signal. The models belonging to this class can be assigned to three different subcategories, depending on the level of detail they capture and the methods used for modeling the spike waveforms. These categories are discussed in the following sections.

3.4.1 Compartment Models

Compartment models are based on mathematical descriptions of how the action potential propagates along the neuron [41, 42, 12]. To facilitate the simulation of complex neuronal geometries, the neuron is divided into structural elements, or compartments, whose electrical properties are well defined. By modeling the membrane properties of each compartment, as well as the interactions between adjacent compartments, compartment models can be used to study the neuronal response to action potential generation at a high level of realism.

Having obtained the time-varying membrane currents in each compartment, they can be used to calculate the resulting voltage changes in the extracellular space using the *line source approximation* (LSA) [43]. The fundamental assumption of the LSA model is that each compartment, represented by a cable (compartment), can be replaced by a linear arrangement of point current sources. The contribution from a given point source to the electric potential in a given measurement point outside the neuron is then given by

$$\phi = \frac{i}{4\pi\sigma_e r} \tag{3.4}$$

where i is the current, σ_e is the conductivity of the extracellular space and r is the distance between the point source and the measurement point. The extracellular potential is then computed as the sum of the potentials from all point sources in all compartments. This combination of compartment modeling and the LSA thus facilitates realistic simulation of the spatial dependency

of the measured spike waveform as a function of the morphology and ionic conductance properties of the neuron [11, 12].

Since the possibility of varying the geometrical and electrical properties of neuronal compartment models is practically unlimited, they are highly flexible and offer a high degree of controllability. However, due to the level of detail and the relatively high computational complexity, simulating extracellular recordings with direct compartment model and the LSA is time consuming and thus unpractical when simulating a wide range of recording scenarios. It is possible, however, to pre-calculate the LSA on a grid of measurement points and then to use interpolation to determine the spike waveforms in off-grid measurement points [44]. Although this may speed up the test signal generation, it does not scale well with increasing the number of neuron-electrode pairs.

3.4.2 Template Based Models

Template based signal models rely on libraries of fixed spike waveforms that are selected, usually at random, and assigned to the neurons contributing to the recording [45, 46]. Apart from the distance between the electrode and the neuron, no geometrical factors influencing the measured spike waveform are accounted for. The spike attenuation as a function of distance r can be modeled, for instance, as a power law

$$A(r) = \frac{1}{(1 + ar)^b} \tag{3.5}$$

where the coefficients a and b determine the shape of the amplitude decay function as the distance r is increased.

Template based models have been shown to provide a high degree of realism in terms of general signal properties (power spectral density, amplitude distribution and autocorrelation), but in their simplest form, this realism only holds when simulating recordings with single non-moving electrodes. Multi-electrode recordings can be simulated with template based models, but again, only the difference in spike amplitudes between the recording channels is accounted for and other variations in the shapes of spike waveforms are discarded.

The absence of mathematical descriptions of how the spike waveform is generated makes the controllability of template based models limited. For instance, studying the effects of electrode movements on the performance in spike sorting is not feasible since no information is available describing how the spike waveforms would be influenced by the movements (except for amplitude). The discrete nature of experimentally obtained spike libraries also limits their usability and controllability.

Although the simplicity of template based models limits their usability to some extent, they should not be underestimated. Due to their low complexity it is possible to generate large sets of test signals with relatively little effort. This facilitates statistical comparison of system performance, which is attractive from a generalizability point of view. This type of analysis was carried out in Paper IV.

In the simulator described in Paper II we implemented a template based model for the synthesis of test signals. The spike library was generated by detecting and sorting spikes from a large number of in-vivo recordings from the cat cerebellum [47]. Spike detection and spike sorting were performed with absolute value threshold detection and *Chronux* [3, 48], respectively. The library contained 85 mean spike waveforms. The simulated recording volume was divided into a near field and a far field, containing target neurons and noise neurons, respectively, and all neurons were assigned a randomly selected spike waveform from the library. Neurons in the near field were initially assigned a constant spike amplitude of one (maximum absolute value). This was changed in a later implementation (Paper IV) to have the near field amplitudes vary randomly. Neurons in the far field were assigned a random position and the spike amplitude was calculated according to the power law of Equation 3.5. Synthetic recordings were validated in terms of power spectral density using Welch's method, autocorrelation and usability for quantifying spike detection accuracy.

In Paper III, we presented a method to extend the spike library provided in Paper II by statistical modeling. Principal component analysis (PCA) was used to find orthonormal basis waveforms describing the spike templates and the distributions of the weights of the first N basis waveforms were then fitted to an N-dimensional K-modal Gaussian mixture model. The number of dimensions ($N = 6$) was taken as the number of dimensions that accounted for 99% of the variance in the spike waveforms. The modeling procedure was carried out for various numbers of modes and the model with the lowest Bayesian information criterion was selected ($K = 2$). The statistical model was shown to be usable for generating random spike waveforms whose properties were similar to those of the original template waveforms.

3.4.3 Compressed Compartment Models

In Paper VI, we introduced a new class of models that combine the realism and controllability of compartment models with the computational simplicity of template based models. An action potential was simulated in four compartment models of a CA1 pyramidal neuron [11] using the simulation environment *NEURON* [49] and the LSA was then used to calculate the extracellular spike

waveforms on a dense grid of measurement points surrounding the neurons. Singular value decomposition was used to find a set of orthonormal basis waveforms for the calculated spikes within a volume bounded by a model specific ellipsoid concentric with the center of the neuron's soma. The weights of the first six basis waveforms were then individually fitted to trivariate polynomial functions of the measurement point coordinates. The spatial dependency of the weight $w_n(x, y, z)$ of the nth basis waveform $b_n(t)$ was thus assumed to be given by

$$w_n(x, y, z) = \sum_i c_{i_n} x^{\mathbf{e}_{i,1}} y^{\mathbf{e}_{i,2}} z^{\mathbf{e}_{i,3}} \tag{3.6}$$

where c_{i_n} is the ith coefficient of the polynomial and $\mathbf{e}$ is a matrix whose ith row contains the ith combination of x, y and z exponents included in the model. Having solved this fitting problem for all six dimensions, the spike waveforms within the model ellipsoid could be calculated as a linear combination of the basis waveforms with the weights calculated according to the polynomials, i.e.

$$s(t, x, y, z) = \sum_{n=1}^{6} w_n(x, y, z) b_n(t). \tag{3.7}$$

For measurement points outside the model ellipsoid, the spike waveform was calculated in the point of intersection between the ellipsoid and a line of sight from the measurement point to the origin and then scaled with a power-law function of the point's distance from the ellipsoid. The procedure of calculating spike waveforms was implemented using matrix multiplications and thus a very large number of measurement points could be treated simultaneously.

The model was validated by comparing modeled and true (*NEURON* generated) spike waveforms in terms of shape and amplitude and by showing that noise properties could be controlled by the appropriate selection of recording geometry and firing properties of distant neurons. The models were implemented into a computationally efficient object oriented simulation tool, written in *Matlab* and the HDF5 file format [50] was used to store simulation data on-the-fly in a memory-efficient way. The model was shown to greatly increase the computational efficiency in simulating realistic multi-channel test recordings compared to previous methods.

In Paper V, the model of Paper VI was used to explore the effects of electrode movements on the appearance of detected spikes in feature space and show that a given movement path in the physical space translated to a similar movement path in the feature space. A path measure was introduced that reflected the distance of points along the path to the mean point on the path. This insight was then used to perform estimation of the electrode position

based on the feature space representation of detected spikes, assuming that spike features were related to electrode position through the linear model

$$\mathbf{B}^T\mathbf{S} = \mathbf{A}\mathbf{X} + \eta \tag{3.8}$$

where the product $\mathbf{S}$ is the spike matrix, $\mathbf{A}$ is a transformation matrix that transforms physical electrode coordinates in $\mathbf{X}$ to points in the spike feature space spanned by $\mathbf{B}$ and η represents noise or variations not captured by the transformation matrix.

3.5 Hybrid Models

Hybrid signal models are based on superimposing synthetic spike trains, generated by any of the models in the "mathematical models" class, onto a real recording of physiological noise [51]. Hybrid models thus offer a high degree of realism, but lack in controllability to some extent, depending on which model is used to generate the synthetic spike trains. Computational complexity also depends on the model used to generate the synthetic spike trains. Since no explicit mathematical description of the background noise is used, hybrid models are not suitable for the simulation of multichannel recordings where electrode sites are closely spaced.

Chapter 4

Wireless Brain-Machine Interfaces and Data Reduction

In its traditional form, the measurement system presented in Chapter 2 relies on wired connections between the system parts. When performing acute experiments on anesthetized subjects, this is not a problem, since the mobility of the subject and risks for surgical/post-surgical complications are usually insignificant factors. However, in experiments on awake and freely moving subjects or in clinical applications where the acquisition device is chronically implanted, the wires and the bulk of the equipment, as well as the potential risks for post-surgical complications become problematic. Wireless brain-machine interfaces have the potential to overcome these limitations.

By *wireless BMIs*, we refer to a measurement system where the electrode, the amplifier, the acquisition part and some of the processing part is implanted into the subject along with a transmitter. The implanted part of the BMI sends the acquired data to an external unit that is used for controlling the measurement device and handling and analyzing the incoming measurement data.

Despite the potential of wireless BMIs, designing them involves a great challenge that mainly stems from two fundamental characteristics of wireless BMIs. First, due to physical and implementational limitations, the information carrying capacity of wireless communication systems is limited. Thus, the transmission of raw sampled data from the neural recording system immediately becomes problematic when increasing the number of measurement channels.

Second, the requirement that the subject-/patient part of the wireless BMI is fully or partially implanted, leads to great restrictions regarding power and area consumption. The second point is especially important from the point of view of clinical applications, where reliability and patient safety are of major significance and can be jeopardized by batteries running low, requiring either inductive charging or surgical operation for replacement.

In this chapter, the challenges involved in designing wireless brain-machine interfaces are discussed from the perspectives of the wireless link as well as the neural data processing part of the system. First, the limitations of the wireless link are presented and the feasibility of influencing its properties in order to increase its capacity are discussed. Second, two steps in obtaining data reduction are discussed as means of efficiently using the available computational and wireless resources at hand. The task of increasing the channel capacity to facilitate high data throughput has not been addressed in the papers, but is still briefly discussed here due to the importance of wireless link capacity within the field of wireless communications.

4.1 Wireless Link Capacity

In 1948, Claude Shannon invented modern information theory. His results characterized the limits for error-free communication within a communication system and provided a theoretical framework for analyzing link capacity [52]. Although the actual achievable data throughput of a communication link depends on implementational aspects such as modulation and coding, Shannon's results can serve as guidelines when approximating maximum performance.

A special case of Shannon's theory applies to the lossless, frequency flat Additive White Gaussian Noise (AWGN) channel, for which the input-output signal relationship is given by

$$y(n) = x(n) + w(n) \tag{4.1}$$

where $y(n)$ is the received signal, $x(n)$ is the transmitted signal and $w(n)$ is the noise added by the channel. The capacity of such a channel (in bits per second) is given by

$$C_{awgn} = B \log_2 \left(1 + \frac{P}{N_0 B}\right) \tag{4.2}$$

where B is the bandwidth, P is the transmitted power and N_0 is the noise spectral density. Under the AWGN assumption, the capacity of the wireless link is thus governed by the signal-to-noise ratio (SNR) P/N_0 and the bandwidth of the channel, B.

As mentioned earlier, the theoretical capacity limit presented above can be used for indicative purposes when relating available channel resources to required data throughput. Assuming that each channel of the neural recording system samples the neural data at a sampling rate of 25 kHz and a wordlength of 10 bits, each channel will provide a raw data rate of 250 kbps to the wireless link. For a bandwidth of 1 MHz and an SNR of 0 dB, the maximum theoretical channel capacity is 1 Mbps, which ideally facilitates the transmission of raw data from four neural recording channels. The data rate vs. capacity ratio thus obviously scales badly with the number of neural recording channels. For instance, a 96 channel neural recording [53] would result in a raw data rate of 24 Mbps, which is way beyond the maximum capacity for the AWGN channel in the above example.

The data rate vs. capacity problem can be addressed by either reducing the data rate or by increasing the channel capacity. According to Equation 4.2, the only ways of directly influencing the channel capacity are to increase the SNR by increasing the transmitted power, or to increase bandwidth. Due to regulatory restrictions set by, e.g. the Federal Communications Commission (FCC) and risks for tissue damage caused by heat, increasing the transmitted power cannot be done indefinitely. Beside those external limitations, increasing the transmitted power is not beneficial in terms of the energy constraints on the implant, as discussed earlier. Due to the high exploitation of the radio spectrum, increasing bandwidth is also limited by regulations and external circumstances.

Moving away from the assumption of the single-input-single-output (SISO) channel in Equation 4.1, it is possible to consider other types of systems to increase the capacity. Multiple-input-multiple-output (MIMO) systems use multiple transmitter and receiver antennas to exploit the spatial dimension in the path between the transmitter and receiver antennas and thereby offer an increase in capacity that is related to the number of subchannels introduced [54, 55]. However, the number of antenna elements and the spatial relationship between the antenna elements are important factors in determining the increase in capacity. Increasing the number of and spacing between antenna elements increases the spatial diversity in the MIMO channel. Assuming a large spacing between the antenna elements and an equal number of transmitter and receiver antennas (N), the maximum capacity is given by

$$C_{MIMO} = NB\log_2(1 + \frac{P}{N_0 B}), \tag{4.3}$$

which is N times the capacity of the SISO channel considered earlier. This is assuming that the receiver has full knowledge of the channel's state at each

time[1]. Achieving this capacity is not realistic in real scenarios, since it is based on the assumption that all subchannels are completely independent and do not interfere with each other [54]. However, using this as a reference for the required capacity by the data rate produced by the 96 neural recording channels in the previous example, we would need a 24×24 MIMO system. Thus, theoretically, MIMO does offer a way of increasing capacity to meet the requirements, but at the cost of drastically increased computational complexity, power consumption and area consumption, all of which are non-feasible properties from the point of view of the implanted part of a wireless BMI.

4.2 Data Reduction

Although, theoretically, there exist means of increasing the channel capacity to match the data rate requirements, they are not practical as discussed in the previous section. Addressing the problem from the other end, i.e. from the standpoint of the input data to the transmitter and utilizing the available channel capacity wisely, is more feasible with respect to both throughput and energy consumption. In this section, two steps in achieving minimal data rate into the transmitter are discussed. Due to the energy constraint on the implant, only low-complexity techniques for data reduction are addressed.

4.2.1 Minimization of Input Data

The simplest way of performing data reduction is that of ensuring minimal acquisition of data, i.e. minimization of sampling rate and resolution in the A/D converter block of the measurement system. Such minimization can be performed either by estimating the bandwidth of the signal and thereby the Nyquist rate [13, 3], but since the spectrum of the extracellular signal is not clearly confined within a certain frequency range, this estimation suffers from subjective assumptions.

Another approach is to minimize sampling rate and resolution while ensuring sustained performance in the analysis of the transmitted data. As already discussed in Section 2.2.3, the work reported in Paper IV addressed the minimization of sampling rate and resolution with sustained performance in spike detection and spike sorting, and thereby the minimization of bitrate into the recording system. We conclude that absolute threshold spike detection required a sampling rate of 16 to 31 kHz and an effective sampling resolution of 9 bits

[1] The channel state information can be established by having the transmitter send a training signal that is used to estimate the channel's transfer function.

for the accuracy to be maintained within 5% and 1% of the maximally achievable accuracy, respectively. For spike sorting we concluded that given that the spikes were well aligned, sampling rate and resolution could, upon spike detection, be lowered to 5 kHz and 5 bits, respectively, with an error tolerance of 1%. In the paper, we concluded by presenting a resource allocation scheme that provides guidelines for minimizing the data rate into the wireless link for various system configurations.

4.2.2 Compression of Spike Data

The compression of spike data involves extracting the spiking component from the recording already at the implant and thus significantly reducing the data rate into the transmitter. Depending on the level of detail of the subsequent analysis that is to be carried out, more or less detailed representations of the extracted spiking component are transmitted. Among the most straightforward approaches described so far include (in order of increasing amount of processing on the implant) transmitting timestamps of the detected spikes [56], the extracted spike waveforms along with timestamps [57, 58] and finally the neuron labels of spikes along with timestamps [59]. In the first approach, no spike sorting is carried out and only the gathered activity of the spiking neurons – the multi-unit activity – can be characterized. In the second approach, the entire spike sorting procedure is carried out in the external unit and in the third approach, the entire spike sorting procedure is carried out in the implant.

Assuming that the single unit activity is to be characterized, the external unit needs access to sorted spike trains. From that point of view, the first approach mentioned above is ruled out, since all information about spike waveforms is lost. If successfully implemented, the third approach would be attractive, being the one conveying information about the activities of individual neurons at the lowest data rate. However, performing spike sorting in the implant is generally both computationally complex and it makes it difficult for the external unit to validate the results since the spike waveforms are never seen by it. The second approach offers a feasible compromise, considering that it retains the information about the spike waveform and leaves the computationally complex task of spike sorting to the less constrained external unit. This approach can be combined with compression algorithms to provide essentially the same information to the external unit, but at a lower data rate than when transmitting the entire spike waveforms.

A *compression basis* consists of a set of waveforms that span a space in which detected spike waveforms have a *sparse* representation. *Sparseness* implies that once a spike waveform has been projected onto the full set of basis waveforms, the majority of the transform (compression) coefficients are small

in amplitude compared to the minority of high-amplitude coefficients. Thus, the spike waveform can be approximated by a linear combination of a small subset of the compression basis waveforms. Compression is thus achieved by discarding the insignificant coefficients and only transmitting the significant ones. This procedure of discarding transformation coefficients can be referred to as *dimensionality reduction*.

For compression of this kind to be beneficial both in terms of data reduction and computational complexity, the compression basis needs to be selected to introduce sparseness, while facilitating a straightforward procedure of selecting compression coefficients to be transmitted. Thus, ideally – from this point of view – one would select a fixed compression basis that, despite never being updated, introduces sparseness and allows the same set of compression coefficients to be selected in all cases. Due to the stereotypical shape of spike waveforms encountered in neural recordings, these criteria can be fulfilled without significant loss in performance compared to more complex methods involving adaptive compression basis and coefficient selection. This was indicated by the results presented in Paper III, that showed that the spike library of Paper II was by 99% described by the first six principal components, indicating that spike waveforms could be compressed with a fixed generic compression basis derived from a large set of experimentally obtained spike waveforms and always using the first (six) compression coefficients.

In Paper VII, we studied the compression of spike waveforms with various combinations of system architectures, spike detectors and fixed compression bases and compared it with the cases when no compression was performed and when an optimal compression basis was found. The spike detectors included were absolute value threshold detection and nonlinear energy operator detection and they were chosen due to their simplicity and performance [28, 32, 23]. System architectures differed in terms of processing task allocation.

Compression was performed by projecting detected spike waveforms onto a compression basis and then removing redundant coefficients, according to

$$\mathbf{W}_d = \mathbf{B}_d \mathbf{B}_c^T \mathbf{S} \tag{4.4}$$

where the matrix $\mathbf{W}_d$ contains the compression coefficients, $\mathbf{B}_d$ is a dimensionality reduction matrix that removes redundant coefficients, $\mathbf{B}_c$ contains the compression basis waveforms in its columns and $\mathbf{S}$ contains the detected spikes in its columns. Since fixed compression bases were assumed, the external unit had full knowledge of the basis being used and could thus reconstruct the detected spikes from the received coefficients $\mathbf{W}_d$ according to

$$\hat{\mathbf{S}} = \mathbf{B}_c \mathbf{B}_d^T \mathbf{W}_d. \tag{4.5}$$

First, we compared the accuracy in spike reconstruction and spike sorting for all combinations of architectures, spike detectors and compression bases. Then, we focused on the basis obtained by singular value decomposition of the spike library from Paper II, since that basis represented the class of generic bases specially derived from spike data. The results indicated that the fixed generic compression basis allowed significant compression with insignificant decrease in spike reconstruction and sorting accuracy, compared to the optimal and uncompressed case.

Chapter 5

Contributions and Discussion

This chapter summarizes the results and contributions of my work within the research field. The papers are summarized individually and their main findings are put into the context of the research field as presented in the previous chapters. The chapter is concluded with a general discussion and some thoughts about possible future work.

My personal contributions to the papers in all cases constitute the main participation in producing the papers, i.e. designs and implementations of the studies as well as processing, analyzing and accounting for the results and conclusions. My co-authors and supervisors have provided me with valuable input throughout all parts of these processes, i.e. during the planning and executions of the studies and writing of the papers.

5.1 Summary of the Papers

5.1.1 Paper I: Implementation of a Telemetry System for Neurophysiological Signals

In this paper, the design, implementation and testing of a four channel wearable telemetry system for neurophysiological measurements is presented. The main purpose of the work reported in this paper is to investigate the practicality of implementing wireless BMIs with generic, commercially available components. Such a pilot investigation is considered to be an important step in identifying the bottlenecks that should be considered in custom designs of wireless BMIs.

The wearable part of the system measures $30 \times 37 \times 3$ mm and consists of a custom designed amplifier and a commercially available integrated circuit containing an A/D converter, a microcontroller and a wireless transceiver. The

system is built using discrete surface-mounted components. The four-channel amplifier has variable gain (58 to 73 dB) and the system can be used to sample and transmit raw data from one of the four channels, at a rate of 3.7 kHz, to an external PC containing a graphical user interface for system control and visualization of data. To save space and to simplify the amplifier design, the amplifier is designed to have the four measurement channels share the greater part of the amplifier chain. However, due to the long settling time when switching between input channels, the design is deemed unpractical for cyclic sampling schemes and from that point of view, it is concluded that complete channel-dedicated amplifier chains are likely to be more practical in a multi-channel telemetric system.

Despite the flexibility of the presented telemetry system, we conclude that a more refined set of design parameters is needed for future system designs and finding those requires further studying of the properties of the extracellular neural signal. This insight serves as a motivation for the work reported in the remaining papers.

5.1.2 Paper II: Spike Library Based Simulator for Extracellular Single Unit Neuronal Signals

This paper deals with the generation of test data with a-priori known characteristics for the quantitative assessment of accuracy in analyzing spike data. The paper presents a simulation tool for synthesizing single channel test recordings utilizing a template based extracellular recording model (see Section 3.4.2). A library of eighty-five experimentally obtained spike templates is used in combination with a renewal process model for spike time generation to generate spike trains for the individual neurons contributing to the recording, which are then added up to comprise the synthetic signal. Spike times and signal components (spiking, biological noise and thermal noise) are stored separately, allowing easy scaling of signal to noise ratio. The spike templates represent mean spike waveforms extracted from a large number of extracellular recordings in the cat cerebellum, and cover a wide range of waveform morphologies. The simulator provides test recordings with realistic background noise and is useful for quick generation of test data with varying properties in terms of firing characteristics, number of target neurons and background noise level. At the time of writing the paper, the availability of simulation tools for the purpose of quantifying system performance was limited, and therefore an important contribution of this work was the releasing of the code for general use within the research community.

5.1.3 Paper III: Statistical Modelling of Spike Libraries for Simulation of Extracellular Recordings in the Cerebellum

This paper provides a realistic extension to the discrete spike library provided in Paper II. Principal component analysis is performed on the original spike library to find a set of orthonormal basis waveforms. The distribution of the first six principal component weights across the assembly of eighty five original waveforms is then fitted to a six-dimensional bi-modal Gaussian mixture model, that can be used to generate arbitrary spike waveforms which are likely to occur, given the original spike library.

The model is shown to be able to generate new spike libraries with the same overall characteristics as the original spike library. In combination with the simulator reported in Paper II, the synthetic spike library is shown to provide synthetic recordings with realistic properties. A double-blind test is performed in which trained neuroscientists, with extensive experience in analyzing spike data, are asked to point out synthetic spikes in a matrix containing a random mixture of real and synthetic spikes. None of the participants performs better than chance when discriminating between real and synthetic spikes.

The number of principal component dimensions used to model the spike library (six) is in agreement with previous studies where large assemblies of varying spike waveforms have been analyzed in a similar manner. This result indicates that a compression basis derived from a generic spike library can be used for computationally simple compression of spike data (see Paper VII).

5.1.4 Paper IV: Minimizing data transfer with sustained performance in wireless brain-machine interfaces

This paper deals with the minimization of data rate into the wireless link by studying in detail the influence of sampling rate and sampling resolution on the accuracy in spike detection and spike sorting at various noise levels and numbers of target neurons. The simulator reported in Paper II, in combination with a library of 2000 synthetic spike waveforms generated with the statistical model derived in Paper II, is used to synthesize 150 test recordings with known spike times and spike identities; 50 of each number of target neurons, 1, 2 and 4. The recordings are processed with various spike detectors and spike sorters and the accuracies are plotted against the simulation variables. For spike detection with the absolute value threshold detector (ABS) and spike sorting with principal component analysis and fuzzy c-means (PCA and FCM), thresholds of 1% to 5% below maximum accuracy in each case are applied to the performance curves to identify sampling rate and resolution breakpoints.

The distributions of these breakpoints across the test recordings are studied as functions of noise level and number of target units and a set of guidelines for the selection of sampling parameters in a given recording situation is established. Procedures for estimating performance in spike detection and spike sorting are described in detail and the PCA feature space representation of spikes as a candidate predictor of spike sorting performance is examined. The influence of spike detection jitter, arising from asynchronous sampling of spikes, on the performance in spike sorting is quantified and a sampling rate breakpoint at which the effect of spike detection jitter becomes insignificant, is identified.

Noise level is identified as the main design factor to concern for two reasons. First, noise level is the primary factor that influences maximum achievable performance in spike detection and spike sorting. Second, as noise level is increased, the distinctiveness of sampling rate breakpoints for spike detection is decreased. Spike alignment is found to be an important factor in lowering the sampling rate breakpoints for spike sorting and the feature space representation of spikes is found to be significantly correlated with performance in spike sorting. However, our results indicate that the clustering algorithm employed is robust to a significant change in the feature space representation, and thus a prediction of spike sorting performance solely based on the spike features is likely to overestimate performance breakpoints.

The sampling rate and resolution breakpoints for spike detection with ABS at practical noise levels are found to be 16 to 31 kHz (5% and 1% error tolerance) and 9 bits, respectively. For spike sorting with PCA and FCM, the breakpoints are found to be 5 kHz and 5 bits, respectively, assuming that spike alignment has been achieved prior to feature extraction. This indicates that significant data reduction can be achieved directly after spike detection by simply discarding samples and bits.

Our comparison of algorithms for spike detection and spike sorting indicates that the breakpoints found for ABS and PCA/FCM can be used as guidelines for predicting the breakpoint locations for the other algorithms.

5.1.5 Paper V: Spike-Feature Based Estimation of Electrode Position in Extracellular Neural Recordings

In this paper, we use synthetic recordings based on compressed compartment models (see Paper VI) to explore how physical movements of the recording electrode translate to movements of detected spikes in the PCA feature space. We show that there is a characteristic relationship between movements in the two domains. This characteristic relationship is demonstrated by comparing a path measure for three different electrode movements (linear, elliptic and P-shaped) in both spaces. We show that the relationship between electrode movements

and spike movements can be modeled by a linear transformation between the two coordinate systems and we finally use the model to estimate the position of the electrode based on PCA features of detected spikes. The estimation involves introducing a training phase in which the transformation matrix is estimated, and then transforming the spike features directly to physical electrode positions using this matrix.

The insights provided by this preliminary study are valuable in the context of spike-tracking, i.e. the pairing of spikes representing the same neuron in recordings separated in time, where electrode movements may have occurred in the meantime. The electrode positioning procedure introduced here could be used to quantify post-implantational movements of electrodes by learning the transformation matrix during linear electrode insertion and then estimating the relative position of the electrode in future recordings based on the acquired spike waveforms. This work serves as a demonstration of the usability of compressed compartment models in modeling physical properties of the recording setup.

5.1.6 Paper VI: Computationally efficient simulation of extracellular recordings with multielectrode arrays

This paper introduces an extracellular signal model that establishes the category of compressed compartment models (see Section 3.4.3). Four compartment models of CA1 pyramidal neurons are used to calculate extracellular spike waveforms on a dense grid of measurement points surrounding the models, using the line source approximation method (LSA). The measurement points are then divided into two groups, according to their corresponding spike amplitude. This division results in two regions – the neuron's near- and far-field, respectively – separated by an ellipsoid, concentric with the cell soma. We apply singular value decomposition (SVD) to the matrix containing all spike waveforms in the near-field to find a set of orthonormal basis waveforms and their respective weights as functions of position relative to the neuron. The weights of the first six dimensions in this decomposition are then individually fitted to trivariate polynomial functions of the measurement point coordinates (x, y, z), allowing the basis waveform weights to be calculated in any off-grid point in space, ensuring continuous change in the waveform when moving along a path of measurement points. For measurement points in the far-field, we model the spike waveforms by finding the point of intersection between the model ellipsoid and the line of sight to the origin, calculating the waveform in that point according to the near-field model, and then attenuating it by a power-law function of the distance. The power-law function is estimated by fitting the amplitudes of spikes in the original measurement points in the far-field to a

power-law of the measurement points' distances to the model ellipsoid. The model is validated by comparing true and modeled spike waveforms in terms of shape and amplitude.

An object oriented simulation tool employing the simulation of multielectrode recordings with arbitrary neuronal population and electrode array geometries is presented and its usability is demonstrated by quantifying spike sorting performance as a function of electrode position. The results demonstrate very clearly the relationship between electrode position, shapes and feature space representations of the detected spike waveforms and the corresponding spike sorting accuracy. The simulator is used to show how the geometries and firing characteristics of neuronal populations can be tweaked to gain control over physiological noise properties, in terms of amplitude, power spectral density and sample histograms.

The modeling method introduced shows a significant improvement in computational and memory efficiency compared to previous methods, and adds a realistic way of simulating multielectrode recordings where the spike waveform from the same neuron differs in shape between electrode sites, not only amplitude. To demonstrate the increased time efficiency, direct LSA calculation of the 14.136 extracellular spike waveforms, calculated when estimating spike sorting accuracy as a function of electrode position, takes approximately 24 minutes. In contrast, our method calculates the entire set of waveforms in approximately 2.4 seconds, or a factor of 600 times faster. This increased computational efficiency increases the practicality of performing simulations in which the spatial dependency of measured spike waveforms is captured.

5.1.7 Paper VII: Compression of neural spikes with fixed generic bases for wireless brain-machine interfaces

This paper deals with the compression of detected spike waveforms with fixed generic compression bases and minimal processing on the implanted part of a wireless BMI. Selecting fixed bases and minimizing the computational complexity on the implant is done in order to maximize the utilization of the wireless link capacity while maintaining low power consumption. Three different system architectures are compared, involving different combinations and arrangements of spike detection, alignment, compression, reconstruction and sorting. Two different spike detectors (absolute value threshold detection and nonlinear energy operator detection) and five different compression bases are included. The first compression basis represents the case where the compression basis is optimized for each recording and is obtained by performing SVD on the matrix containing the spike waveforms that are to be compressed. The second compression basis is a fixed basis obtained by performing SVD on a matrix

containing a large number of spike waveforms calculated using the compressed compartment models derived in Paper VI. The third compression basis is a fixed basis obtained by performing SVD on the matrix of experimentally obtained spike templates provided in Paper II. The fourth and fifth compression bases are the downsampling basis and Haar wavelet basis, respectively. The test signals used are generated with the simulator reported in Paper VI and they represent a wide range of recording SNRs. For reference, an architecture involving no spike compression at all, is included.

All combinations of a system architecture, spike detector and compression basis are initially compared in terms of spike reconstruction accuracy and spike sorting accuracy. Second, based on the results from the first part, we select two combinations of architectures and spike detectors and compare their accuracies in spike reconstruction and sorting when compressing with the third compression basis. That basis is of special interest since it represents the class of generic fixed bases that have no mathematical relation to the data that is to be compressed, but is derived from a large assembly of real spike waveforms with various shapes. Thus, if successful in accurately representing the data, it shows that we can remove the computationally intensive tasks of finding and maintaining an optimal compression basis from the implant.

We conclude that given the appropriate system architecture, fixed generic compression bases can indeed be used for compressing spike data without significantly reducing accuracy in spike reconstruction and sorting, compared to using an optimal basis or not performing compression at all. We suggest a compression architecture consisting of the absolute value threshold spike detector, maximum value spike alignment on the implant side and compression with at least a four dimensional fixed generic compression basis, derived from a widely varying set of experimentally obtained spike waveforms. Assuming a wireless link capacity of 1 Mbps, four spiking neurons per channel, each firing on average 10 spikes per second, this would theoretically allow the transmission of over 600 neural recording channels, in contrast to four channels when transmitting raw neural data.

5.2 Discussion and Future Work

Wireless brain-machine interfaces offer solutions to many of the problems that are inherently present in wired BMIs. However, when increasing the number of neural recording channels beyond just a couple, the capacity of the wireless link soon becomes a limitation and efforts have to be made in order to either increase the link capacity or utilize the available capacity efficiently. Out of these two actions, efficient utilization of the wireless link is the most efficient

and practical one from a cost-benefit point of view, given the raw data rate of neural recordings and the regulatory and physical restrictions that limit the freedom to modify the channel capacity. Ensuring efficient use of resources demands access to realistic signal models that allow the detailed study of system performance as a function of input to the wireless link. This thesis addresses both the required signal modeling, and the use of the signal models for studying data reduction methods that ensure efficient use of resources.

As shown in Chapter 3, there are several ways of modeling the extracellular recording and those differ in terms of realism, controllability and complexity. When selecting a model to work with, one should consider the level of detail and realism that needs to be captured from the point of view of the application that is to be validated. For example, template based models probably offer an adequate representation of the neural signal when studying single channel recordings where electrode movements cannot arise, but they fail as soon as the spatial dimension is introduced in the form of movements or specific geometries.

Geometrically realistic models such as compartment models or compressed compartment models capture the spatial dependency of the measured signal and should therefore be considered for simulations with well-defined geometrical configurations and/or multi-channel recordings. Although the highest degree of realism is provided by direct simulations with compartment models, their complexity makes them unpractical to implement in simulation platforms where flexibility and ease of use are of great importance to the developer of algorithms for signal analysis. Developers need to be able to easily set up arbitrary recording scenarios and test the output of their analysis tools against the immediately available ground truth. Compressed compartment models provide a solution to this, combining the realism of compartment models with the computational efficiency of simpler models.

Working further on the user-friendliness of our compressed compartment models and making them available to the research community would be of great benefit from the perspective of developing "standardized" toolboxes for the analysis of neural data and the development of algorithms. The simulation output could be connected to an analog output, allowing it to be used to test the performance of complex hardware from the output of the electrode to the system's output. Due to the flexibility and ease of use of the simulation environment, its use could be incorporated into the training of neurophysiologists to provide a direct visual feedback representing the origin of the extracellular signal.

Building up a database of compressed compartment models of neurons from specific brain regions would facilitate the simulation of those regions and allow the combined simulation of complex electrode designs adapted to the geometry of the tissue. Not only would this be beneficial from the point of view of

designing algorithms for signal processing, but also from the point of view of designing electrodes to target specific brain regions. The realism of the model could be further increased by considering finite electrode designs and electrode encapsulation. A finite electrode site could be replaced with a cluster of point-electrodes and the signals from the point-electrodes could then be spatially integrated in order to obtain the signal measured by the finite electrode site. Electrode encapsulation could be simulated by studying the frequency response of the encapsulation tissue in real scenarios and adding it as a filter to the signal generation chain in the simulator. These extensions could potentially save a great amount of time spent in testing electrode designs and it would allow for a tighter integration between the procedures of designing electrodes and analysis methods.

Our results regarding the relationship between physical electrode movements and the feature space representation of detected spikes could potentially be valuable in clinical applications such as deep brain stimulation (DBS). By monitoring the feature space representation of detected spikes while inserting the DBS electrode, the relative position of the electrode could potentially be determined and its mechanical stability could be observed after implantation. The first step in this direction is to validate our preliminary model-based results experimentally.

Our results suggest that significant data reduction can be achieved by simply discarding samples after spike detection and/or by compression with fixed generic compression bases. The high compression ratios obtained by introducing our methods and the insignificant difference in performance in signal analysis compared to the uncompressed case, suggest that the methods considered here are indeed feasible alternatives to more sophisticated solutions where compression basis optimization is employed. The next step is to implement our data reduction techniques into a physical platform, such as FPGA, to provide added insight into the practical aspects of their implementation and use. Having implemented an analog interface to the compressed compartment model based simulator, as mentioned above, the real-life performance of the hardware implementation could be verified using the simulator.

References

[1] M. a. Lebedev and M. a. L. Nicolelis, "Brain-machine interfaces: past, present and future.," *Trends in neurosciences*, vol. 29, pp. 536–46, Sept. 2006.

[2] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, *Principles of Neural Science, Fourth Edition*. McGraw-Hill Medical, 4 ed., 2000.

[3] P. Mitra and H. Bokil, *Observed Brain Dynamics*. Oxford University Press, USA, 2008.

[4] A. Hodgkin and A. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *Bulletin of mathematical biology*, vol. 52, no. 1, pp. 25–71, 1990.

[5] J. I. Vandenberg and S. G. Waxman, "Hodgkin and Huxley and the basis for electrical signalling: a remarkable legacy still going strong," *The Journal of Physiology*, vol. 590, pp. 2569–2570, June 2012.

[6] J. P. Donoghue, "Connecting cortex to machines: recent advances in brain interfaces.," *Nature neuroscience*, vol. 5 Suppl, pp. 1085–8, Dec. 2002.

[7] M. Nicolelis, "Brain-machine interfaces to restore motor function and probe neural circuits," *Nature Reviews Neuroscience*, vol. 4, no. May, 2003.

[8] G. Buzsáki, "Large-scale recording of neuronal ensembles.," *Nature neuroscience*, vol. 7, pp. 446–51, May 2004.

[9] A. B. Schwartz, "Cortical neural prosthetics.," *Annual review of neuroscience*, vol. 27, pp. 487–507, Jan. 2004.

[10] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding.," *Nature*, vol. 453, pp. 1098–101, June 2008.

[11] C. Gold, D. a. Henze, C. Koch, and G. Buzsáki, "On the origin of the extracellular action potential waveform: A modeling study.," *Journal of neurophysiology*, vol. 95, pp. 3113–28, May 2006.

[12] K. H. Pettersen and G. T. Einevoll, "Amplitude variability and extracellular low-pass filtering of neuronal spikes.," *Biophysical journal*, vol. 94, pp. 784–802, Feb. 2008.

[13] M. Abeles and M. Goldstein, "Multispike train analysis," *Proceedings of the IEEE*, vol. 65, no. 5, pp. 762–773, 1977.

[14] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials.," *Network (Bristol, England)*, vol. 9, pp. R53–78, Nov. 1998.

[15] U. Mitzdorf, "Current source-density method and application in cat cerebral cortex: investigation of evoked potentials and EEG phenomena.," *Physiological reviews*, vol. 65, pp. 37–100, Jan. 1985.

[16] E. Schmidt, "Electrodes for many single neuron recordings," in *Methods for Neural Ensemble Recordings* (M. Nicolelis, ed.), ch. 1, pp. 1–23, Boca Raton: CRC, 1999.

[17] F. Ejserholm and P. Kohler, "A polymer based electrode array for recordings in the cerebellum," *... (NER), 2011 5th ...*, pp. 376–379, 2011.

[18] S. F. Lempka, M. D. Johnson, M. a. Moffitt, K. J. Otto, D. R. Kipke, and C. C. McIntyre, "Theoretical analysis of intracortical microelectrode recordings.," *Journal of neural engineering*, vol. 8, p. 045006, Aug. 2011.

[19] R. Rieger and J. T. Taylor, "Design strategies for multi-channel low-noise recording systems," *Analog Integrated Circuits and Signal Processing*, vol. 58, pp. 123–133, Nov. 2008.

[20] K. Koch and M. Stieglitz, "Considerations on Noise of Amplifiers and Electrodes for Bioelectrical Signal Recording," *Engineering In Medicine And Biology*, pp. 0–2, 2002.

[21] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-Gaussian variability.," *Journal of neuroscience methods*, vol. 69, pp. 175–88, Nov. 1996.

[22] Y. Ghanbari, P. Papamichalis, and L. Spence, "Robustness of neural spike sorting to sampling rate and quantization bit depth," in *2009 16th International Conference on Digital Signal Processing*, pp. 1–6, IEEE, July 2009.

[23] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Minimizing data transfer with sustained performance in wireless brain-machine interfaces.," *Journal of neural engineering*, vol. 9, p. 036005, June 2012.

[24] S. Gibson, R. Chandler, V. Karkare, D. Markovic, and J. W. Judy, "An efficiency comparison of analog and digital spike detection," *2009 4th International IEEE/EMBS Conference on Neural Engineering*, pp. 423–428, Apr. 2009.

[25] R. Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering.," *Neural computation*, vol. 16, pp. 1661–87, Aug. 2004.

[26] S. Mukhopadhyay and G. C. Ray, "A new interpretation of nonlinear energy operator and its efficacy in spike detection.," *IEEE transactions on bio-medical engineering*, vol. 45, pp. 180–7, Feb. 1998.

[27] K. H. Kim and S. J. Kim, "A wavelet-based method for action potential detection from extracellular neural signal recording with low signal-to-noise ratio.," *IEEE transactions on bio-medical engineering*, vol. 50, pp. 999–1011, Aug. 2003.

[28] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application.," *IEEE transactions on bio-medical engineering*, vol. 51, pp. 905–11, June 2004.

[29] F. Franke, M. Natora, C. Boucsein, M. H. J. Munk, and K. Obermayer, "An online spike detection and spike classification algorithm capable of instantaneous resolution of overlapping spikes.," *Journal of computational neuroscience*, vol. 29, pp. 127–48, Aug. 2010.

[30] A. S. Dickey, A. Suminski, Y. Amit, and N. G. Hatsopoulos, "Single-unit stability using chronically implanted multielectrode arrays.," *Journal of neurophysiology*, vol. 102, pp. 1331–9, Aug. 2009.

[31] A. Pavlov, V. a. Makarov, I. Makarova, and F. Panetsos, "Sorting of neural spikes: When wavelet based methods outperform principal component analysis," *Natural Computing*, vol. 6, pp. 269–281, Aug. 2007.

[32] S. Gibson, J. W. Judy, and D. Marković, "Technology-aware algorithm design for neural spike detection, feature extraction, and dimensionality reduction.," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 18, pp. 469–78, Oct. 2010.

[33] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. New York, NY: Wiley, 2nd ed., 2001.

[34] G. T. Einevoll, F. Franke, E. Hagen, C. Pouzat, and K. D. Harris, "Towards reliable spike-train recordings from thousands of neurons with multielectrodes.," *Current opinion in neurobiology*, vol. 22, pp. 11–7, Feb. 2012.

[35] V. Moca, D. Nikolic, and R. Muresan, "Real and Modeled Spike Trains: Where Do They Meet?," *Artificial Neural Networks-ICANN 2008*, pp. 488–497, 2008.

[36] D. Heeger, "Poisson model of spike generation," *Handout, University of Standford*, pp. 1–13, 2000.

[37] B. Connors and M. Gutnick, "Intrinsic firing patterns of diverse neocortical neurons," *Trends in neurosciences*, 1990.

[38] J. M. Bower and D. Beeman, "The Book of Genesis - Exploring Realistic Neural Models with the GEneral NEural SImulation System," *Genesis*, p. 2003, 2003.

[39] J. H. Macke, P. Berens, A. S. Ecker, A. S. Tolias, and M. Bethge, "Generating spike trains with specified correlation coefficients.," *Neural computation*, vol. 21, pp. 397–423, Feb. 2009.

[40] K. D. Harris, D. a. Henze, J. Csicsvari, H. Hirase, and G. Buzsáki, "Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements.," *Journal of neurophysiology*, vol. 84, pp. 401–14, July 2000.

[41] L. S. Smith and N. Mtetwa, "A tool for synthesizing spike trains with realistic interference.," *Journal of neuroscience methods*, vol. 159, pp. 170–80, Jan. 2007.

[42] C. Gold, D. a. Henze, and C. Koch, "Using extracellular action potential recordings to constrain compartmental models.," *Journal of computational neuroscience*, vol. 23, pp. 39–58, Aug. 2007.

[43] G. R. Holt and C. Koch, "Electrical interactions via the extracellular potential near cell bodies.," *Journal of computational neuroscience*, vol. 6, no. 2, pp. 169–84, 1999.

[44] F. Franke, M. Natora, P. Meier, E. Hagen, K. H. Pettersen, H. Linden, G. T. Einevoll, and K. Obermayer, "An automated online positioning system and simulation environment for multi-electrodes in extracellular recordings.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2010, pp. 593–7, Jan. 2010.

[45] P. T. Thorbergsson, H. Jorntell, F. Bengtsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Spike library based simulator for extracellular single unit neuronal signals.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2009, pp. 6998–7001, Jan. 2009.

[46] J. Martinez, C. Pedreira, M. J. Ison, and R. Quian Quiroga, "Realistic simulation of extracellular recordings.," *Journal of neuroscience methods*, vol. 184, pp. 285–93, Nov. 2009.

[47] H. Jörntell and C.-F. Ekerot, "Reciprocal bidirectional plasticity of parallel fiber receptive fields in cerebellar Purkinje cells and their afferent interneurons.," *Neuron*, vol. 34, pp. 797–806, May 2002.

[48] "Chronux Analysis Software - http://www.chronux.org."

[49] M. L. Hines and N. T. Carnevale, "The NEURON Simulation Environment," *Neural Computation*, vol. 9, pp. 1179–1209, Aug. 1997.

[50] "The HDF Group website - http://www.hdfgroup.org."

[51] C. Pouzat, O. Mazor, and G. Laurent, "Using noise signature to optimize spike-sorting and to assess neuronal classification quality.," *Journal of neuroscience methods*, vol. 122, pp. 43–57, Dec. 2002.

[52] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge University Press, 2005.

[53] J. Csicsvari, "Massively Parallel Recording of Unit and Local Field Potentials With Silicon-Based Electrodes," *Journal of Neurophysiology*, vol. 90, pp. 1314–1323, Apr. 2003.

[54] G. Foschini and M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless personal communications*, pp. 311–335, 1998.

[55] A. F. Molisch, *Wireless Communications*. John Wiley & Sons, 2nd ed., 2012.

[56] R. R. Harrison, P. T. Watkins, R. J. Kier, R. O. Lovejoy, D. J. Black, B. Greger, and F. Solzbacher, "A Low-Power Integrated Circuit for a Wireless 100-Electrode Neural Recording System," *IEEE Journal of Solid-State Circuits*, vol. 42, pp. 123–133, Jan. 2007.

[57] "Alpha Omega - http://www.alphaomega-eng.com."

[58] M. Rizk, I. Obeid, S. H. Callender, and P. D. Wolf, "A single-chip signal processing and telemetry engine for an implantable 96-channel neural data acquisition system.," *Journal of neural engineering*, vol. 4, pp. 309–21, Sept. 2007.

[59] F. Zhang and M. Aghagolzadeh, "An implantable neuroprocessor for multichannel compressive neural recording and on-the-fly spike sorting with wireless telemetry," *Biomedical Circuits and*, pp. 3–6, 2010.

Part II

Included Papers

Paper I

Implementation of a Telemetry System for Neurophysiological Signals

Abstract

With an ever increasing need for assessment of neurophysiological activity in connection with injury and basic research, the demand for an efficient and reliable data acquisition system rises. Brain-machine interfaces is one class of such systems that targets the central nervous system. A necessary step in the development of a brain-machine interface is to design and implement a reliable and efficient measurement system for neurophysiological signals. The use of telemetric devices increases the flexibility of the devices in terms of subject mobility and unobtrusiveness of the equipment. In this paper, we present a complete system architecture for a wearable telemetry system for the acquisition of neurophysiological data. The system has been miniaturized and implemented using surface-mount technology. System performance has been successfully verified and bottlenecks in the architecture have been identified.

Based on: P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Implementation of a Telemetry System for Neurophysiological Signals", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 1254 – 1257, 2008.

1 Introduction

The spinal cord is the main pathway for signals traveling between the brain and the body. Spinal cord injury may cause a break in this pathway. Depending on the location and severity of the lesion, different parts of the body can lose their function partially or totally [1]. In addition to processing motor and sensory information, signals in the central nervous system (CNS) carry valuable information about perception, memory, pain, learning mechanisms etc. [2]. Access to the signals conducted by the CNS could hence provide the possibility to restore function after injury and study the underlying mechanisms of the features mentioned above.

A brain-machine interface (BMI) is an artificial interface between the nervous system and the outside world. With appropriate ways of handling the information delivered by the interface, it becomes possible to use it to provide control signals for prosthetic devices. With the ever increasing interest in BMIs, the need for efficient ways of handling the data increases [3]. Some important parameters in this aspect are pre-amplifiers, filtering requirements, sampling, data transmission, power consumption, size and flexibility in operation. Together, these factors determine the usefulness of implementing BMIs with implanted telemetric devices.

The properties of the measured signals determine the design criteria for the physical implementation of the BMI. Extracellular recordings have been shown to provide information about the intracellular activity of neurons [4][5]. However, factors such as encapsulation of electrodes and electrode position influence the properties of the acquired waveform [5][6]. Due to the variations in signal properties depending on these parameters, design criteria will vary between situations. Based on experience of recordings with wire electrodes on different structures in the CNS, we have made some general assumptions regarding important features such as amplitude and bandwidth. The amplifier should have sufficient gain for the A/D converter to detect signal amplitudes ranging from tens to hundreds of μV with bandwidth up to 5 kHz [7][8].

Various ways of implementing multi-channel neural amplifiers and acquisition systems have been proposed. In some designs, each channel has been implemented with a partial or full amplifier chain [9][10][11]. Different approaches have also been made to design telemetric devices [12][13][14].

This paper suggests a complete architecture for a miniature wearable four-channel neural data acquisition system for animal experiments. The work is intended to provide a proof of concept for the suggested architecture. The architecture can be expanded to work with an arbitrary number of channels, though within some restrictions related to lowered performance with increased amount of channels. The system has been implemented and the function of its

parts has been successfully verified.

2 System Description

The core building blocks of the telemetry system are a front-end analog amplifier with variable gain and input channel and an nRF24E1 transceiver module from *Nordic Semiconductor*. The nRF24E1 features an on-board 100 ksps, 8 channel, 12 bit A/D converter, a serial RS-232 interface, a 2.4GHz transceiver and an 8051 compatible CPU. The amplifier gain, reference level and bandwidth are set to interface the A/D converter. The CPU is used to control two 4:1 multiplexers (ADG804) to select an input channel and a gain. A 2.4 GHz ceramic chip antenna (ANT-2.45-CHP) is connected to the nRF24E1 via a balun matching network. The nRF24E1 firmware is stored on an in-circuit programmable EEPROM (25LC320A). The whole wearable device is powered by a single 3V Li battery (CR2032). Figure 1 shows the top side of the first prototype of the telemetry system.

2.1 Analog Front-End

A circuit schematic of the analog amplifier is shown in figure 3. The inputs are highpass filtered with a first order filter (f_c = 34 Hz) to suppress a potential DC-offset present at the electrode tips. Large resistors (RIN, 4.7 MΩ) increase the input impedance and provide biasing at the inputs. One of the four input channels is selected with MUX1 (Analog Devices ADG804), via I/O pins on the nRF24E1. The amplifier consists of three operational amplifiers (OPA348). A virtual ground is provided with OP3. The resistors Rr1 and Rr2 are chosen so that the virtual ground potential is in the centre of the A/D converter's input signal range. The battery level is monitored with one of the A/D converter channels of the nRF24E1. The first gain stage provides a fixed gain of 40 dB. The second stage provides a variable gain of 18 to 34 dB. The gain of the second stage can be chosen from four different values. The gain is set with MUX2 (Analog Devices ADG804) via I/O pins on the nRF24E1. Both amplifier stages provide further highpass filtering through the capacitors Cf1 and Cf2. A filter bank on the amplifier output provides variable lowpass filtering for sampling at various sampling rates. The outputs of the filters are connected to different channels of the A/D converter. Filter selection is done by selecting the appropriate A/D converter input.

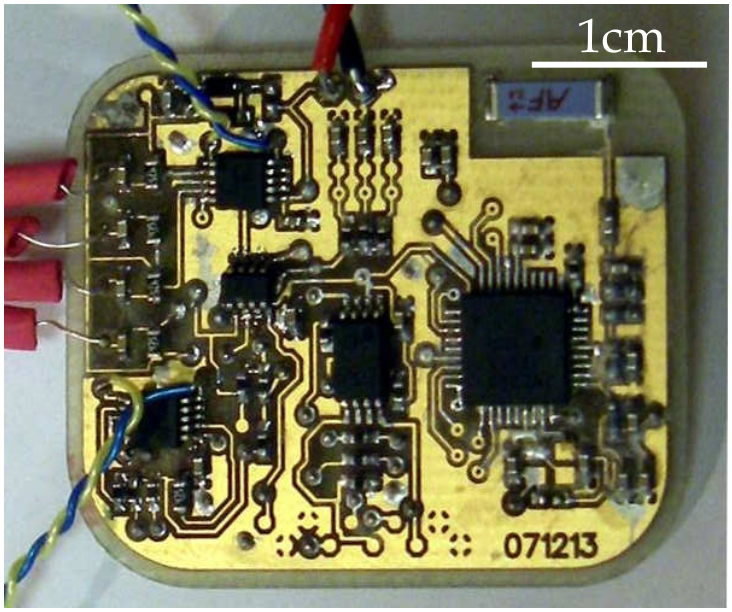


Figure 1: The dimensions of the first prototype of the wearable telemetry device are $30 \times 37 \times 3$ mm^3. All components are mounted on the top side of the PCB. The analog amplifier is on the left and the nRF24E1 with it's peripheral components is on the right.

2.2 Data-Acquisition and -Transmission

The wearable device listens for control words from the basestation at regular intervals. The length of these intervals can be set by the user during initial programming of the device. This reduces the current consumption of the device severely. The control words contain information about session mode, session length, input channel, amplifier gain, and sampling rate. The output signal of the amplifier is sampled with 8 bit resolution and a sampling rate according to the user. The A/D converter uses an internal bandgap voltage as a reference (nominally 1.22 V).

Wireless communication is carried out via the on-board transceiver on the nRF24E1. After clocking in receiver address and payload, data is sent in a short burst at 1 Mbps, i.e. samples are saved in a buffer and then transmitted. When data is not being sent or received, the transceiver can be kept in standby mode in order to reduce its current consumption.

Data-acquisition and -transmission can be carried out in various manners,

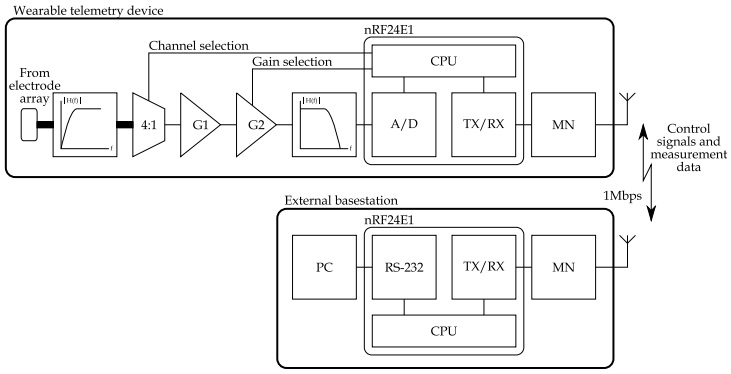


Figure 2: A block diagram of the telemetry system. The analog front-end and nRF24E1 module are implemented on the same PCB. The external base-station is based on an nRF24E1 evaluation board connected to a PC via a serial link.

depending on the task. Signals can either be sampled at high sampling rates, stored in an internal memory and sent in short bursts, or they can be sent one sample at a time, allowing for on-line monitoring of the registered activity. The latter option lowers the sampling rate because of increased processing time per sample.

2.3 External Basestation

An external basestation is used to communicate with the wearable telemetry device. Control words are sent by the user from a PC to an nRF24E1 on the basestation via a serial interface (RS-232). The nRF24E1 on the basestation transmits the control words to the wearable device and starts waiting for incoming measurement data. Once the measurement data is received, it is forwarded via the serial port to the PC for further analysis and processing.

2.4 User Interface

The operator of the system can interact with it via a graphical user interface (GUI), implemented in *MATLAB*. Through the interface, parameters such as sampling rate, input channel(s), amplifier gain, session length and mode of operation can be set. Figure 4 shows a screenshot of the GUI. Old measurement data can be accessed through the same graphical interface and frequency analysis, spectrogram, digital filtering and audio playback of selected segments of the measurement data can be activated. The user can enter a comment apply-

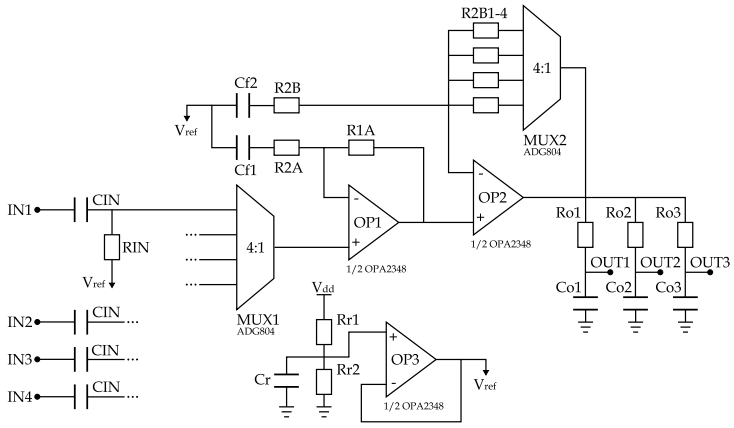


Figure 3: The analog front-end of the wearable telemetry device consists of a single amplifier chain. Channels are time-division multiplexed at the amplifier inputs after highpass filtering of the channels.

ing to the capture session that is saved with the measurement data for later reference. Sampling rate calibration can be initiated from the GUI.

3 Verification of Performance

The performance of the telemetry system was evaluated by measuring some important parameters and comparing with results from simulations and expected values according to component specifications. Table 1 shows a summary of the results. Spice simulations were done in *NI Multisim 10*.

3.1 Analog Front End

Input Impedance

The input impedance of amplifier was measured to be 2.2 MΩ at 10 kHz. Simulations gave an input impedance of 1.5 MΩ, also at 10 kHz.

Amplifier Gain and Bandwidth

The amplifier gain was measured at all gain settings to range between 58 and 73 dB. Simulations gave gain values corresponding to those given by the design criteria, 55 to 70 dB. With the 1.22 V internal bandgap voltage as A/D

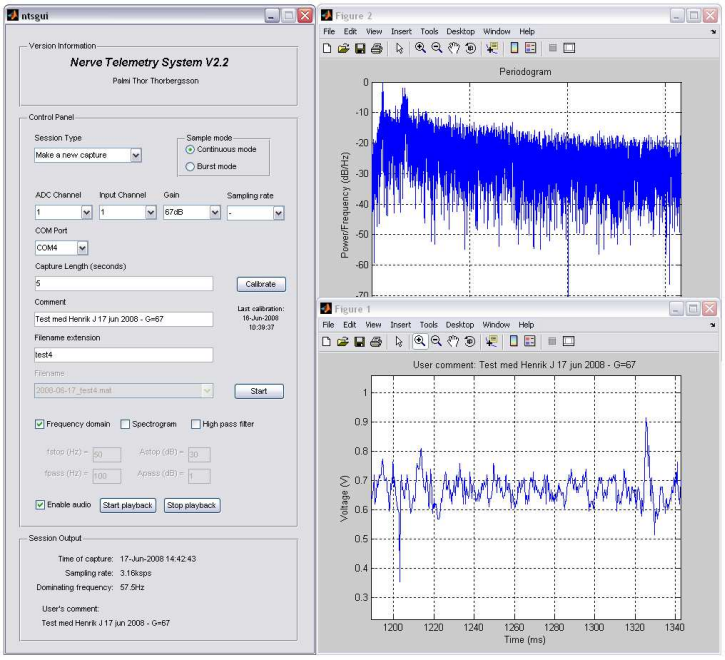


Figure 4: The wearable device is controlled via a graphical user interface implemented in *MATLAB*.

converter reference, the amplifier is capable of full-scale amplification of signal amplitudes from 270 μV to 1.5 mV. Measured amplifier bandwidth was somewhat lower than the simulations indicated. This discrepancy is assumed to be a result of stray effects and inaccurate modeling of some of the components.

Amplifier Noise

The input referred amplifier noise was measured at all gain settings and the average was found to be 5.6μV (rms). The simulations gave an input referred noise of 6.7 μV (rms). The difference may be due to approximations in calculations.

3.2 Sampling Rate

Sampling rate was measured in diffent modes of operation. When running in burst-mode, the total sampling rate was measured to be 25 ksps. When

Table 1: A summary of performance evaluation

Property	Expected / Simulated	Measured / Observed
Input impedance @ $f = 1kHz$ [MΩ]	1.5	2.2
Amplifier gain [dB]	55 to 70	58 to 73
Input referred amplifer noise (rms) [μV]	6.7	5.6
Amplifier current consumption [μA]	160	150
Current consumption of nRF24 and peripherals [mA]	≥ 3	3.5 to 9
Physical dimensions of wearable device [mm^3]	-	$30 \times 37 \times 3$

approaching the nRF24E1 limitations, the sampling rate was found to rely heavily on efficient programming of the device. Assuming continuous mode (sampling and transmitting one sample at a time), a sampling rate of 3.7 ksps was achieved.

3.3 Physical Dimensions

The physical dimensions of the first prototype of the wearable device were measured to be $30 \times 37 \times 3\,\mathrm{mm}^3$ (not including the battery). These dimensions are assumed to be acceptable for a "backpack" version of the wearable system [15].

3.4 Current Consumption and Lifetime

The current consumption of the amplifier was measured as the total current drawn from the battery with the nRF24E1 disconnected. It was measured to be 150 μA.

The current consumption of the nRF24E1 transceiver module was measured when running in different modes. With the measurement system running but the radio transceiver turned off, the current consumption was 3.5 mA. When the system was waiting for start signals from the basestation, the current consumption was 5 mA and when active (continuous mode), it was 9 mA. Current consumption depends strongly on efficient programming of the device firmware.

The system was powered by a GP CR2032 Li 3 V battery. This battery has a capacity of 210 mAh. Assuming a total current consumption of 9 mA in active continuous mode, this results in a lifetime of approximately 23 hours of continuous usage.

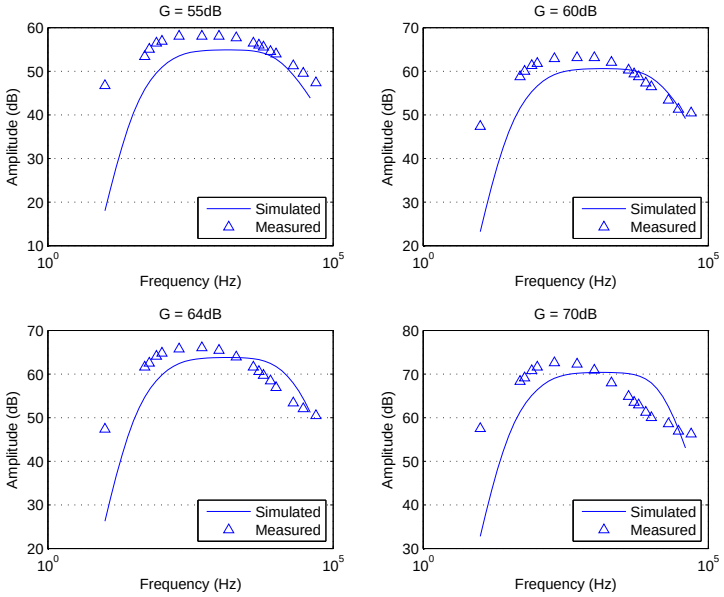


Figure 5: The amplitude of the amplifer transfer function at the four different gain settings.

3.5 Channel Selection

The time it takes to switch from one input channel to another was measured to be 60 ms. This is not due to multiplexer switching speed, but due to long settling time of the input amplifier chain.

4 Conclusions and Future Work

A wearable system for telemetric acquisition of neurophysiological signals has been designed, implemented and tested. Design criteria were set with signal properties such as amplitude and bandwidth in mind. The overall system performance has been evaluated and most parameters have been shown to meet the design criteria and behave according to expectations.

The choice to carry out channel selection early in the amplifier chain was done in order to simplify the amplifier design and save space on the PCB. However, the long settling time when switching channels makes cyclic sampling schemes between channels impractical. This suggests that the channel selection

be carried out later in the amplifier chain. An important factor to keep in mind in this regard is the the increased area consumption when increasing the size of channel-dedicated amplifier chains.

For an explicit set of design criteria to be set for the telemetry device, further studies will have to be made on the properties of the acquired signals, that is extracellular action potentials from different parts of the CNS. Some work based on simulations has been carried out [4][5][16]. However, flexibility and the presented system's ability to adapt to different situations makes it usable even when the exact characteristics of the signals are knot known. The system has been designed with flexibility in mind and will be further tested under various conditions in order to verify its degree of flexibility.

The wearable device will be developed further in terms of physical layout, system architecture and programming. The physical structure of the wearable telemetry device can be improved and reduced in size by using a module-based approach. The amplifier and the transceiver parts can be implemented in separate modules and stacked on top of each other. The size reduction will make it possible to implant the device in a small animal.

5 Acknowledgements

The authors would like to thank their collegues at the Neuronano Research Center at Lund University for their valuable input during this work.

References

[1] F. M. Maynard, M. B. Bracken, G. Creasey, J. F. Ditunno, W. H. Donovan, T. B. Ducker, S. L. Garber, R. J. Marino, S. L. Stover, C. H. Tator, R. L. Waters, J. E. Wilberger, and W. Young, "International Standards for Neurological and Functional Classification of Spinal Cord Injury. American Spinal Injury Association.," *Spinal cord*, vol. 35, pp. 266–74, May 1997.

[2] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, *Principles of Neural Science, Fourth Edition*. McGraw-Hill Medical, 4 ed., 2000.

[3] F. a. Mussa-Ivaldi and L. E. Miller, "Brain-machine interfaces: computational demands and clinical needs meet basic neuroscience.," *Trends in neurosciences*, vol. 26, pp. 329–34, June 2003.

[4] D. a. Henze, Z. Borhegyi, J. Csicsvari, a. Mamiya, K. D. Harris, and G. Buzsáki, "Intracellular features predicted by extracellular recordings in

the hippocampus in vivo.," *Journal of neurophysiology*, vol. 84, pp. 390–400, July 2000.

[5] C. Gold, D. a. Henze, C. Koch, and G. Buzsáki, "On the origin of the extracellular action potential waveform: A modeling study.," *Journal of neurophysiology*, vol. 95, pp. 3113–28, May 2006.

[6] W. M. Grill and J. T. Mortimer, "Electrical properties of implant encapsulation tissue.," *Annals of biomedical engineering*, vol. 22, no. 1, pp. 23–33, 1994.

[7] P. T. Thorbergsson, *Nerve Telemetry System*. Msc thesis, Lund University, 2007.

[8] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. a. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn, and J. P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia.," *Nature*, vol. 442, pp. 164–71, July 2006.

[9] I. Obeid, "A low power multichannel analog front end for portable neural signal recordings," *Journal of Neuroscience Methods*, vol. 133, pp. 27–32, Feb. 2004.

[10] B. Gosselin and M. Sawan, "A low power portable multichannel neural data acquisition system," *The 10th Annual Conference of the International . . .*, no. July, pp. 2–4, 2005.

[11] R. Rieger, M. Schuettler, D. Pal, C. Clarke, P. Langlois, J. Taylor, and N. Donaldson, "Very low-noise ENG amplifier system using CMOS technology.," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 14, pp. 427–37, Dec. 2006.

[12] I. Obeid, "A multichannel telemetry system for single unit neural recordings," *Journal of Neuroscience Methods*, vol. 133, pp. 33–38, Feb. 2004.

[13] Y.-K. Song, W. R. Patterson, C. W. Bull, D. a. Borton, Y. Li, A. V. Nurmikko, J. D. Simeral, and J. P. Donoghue, "A brain implantable microsystem with hybrid RF/IR telemetry for advanced neuroengineering applications.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2007, pp. 445–8, Jan. 2007.

[14] S. Farshchi, D. Markovic, S. Pamarti, B. Razavi, and J. W. Judy, "Towards neuromote: a single-chip, 100-channel, neural-signal acquisition, processing, and telemetry device.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2007, pp. 437–40, Jan. 2007.

[15] S. Xu, S. K. Talwar, E. S. Hawley, L. Li, and J. K. Chapin, "A multichannel telemetry system for brain microstimulation in freely roaming animals," *Journal of Neuroscience Methods*, vol. 133, pp. 57–63, Feb. 2004.

[16] C. Moulin, A. Glière, D. Barbier, S. Joucla, B. Yvert, P. Mailley, and R. Guillemaud, "A new 3-D finite-element model based on thin-film approximation for microelectrode array recording of extracellular action potential.," *IEEE transactions on bio-medical engineering*, vol. 55, pp. 683–92, Mar. 2008.

Paper II

Spike Library Based Simulator for Extracellular Single Unit Neuronal Signals

Abstract

A well defined set of design criteria is of great importance in the process of designing brain machine interfaces (BMI) based on extracellular recordings with chronically implanted micro-electrode arrays in the central nervous system (CNS). In order to compare algorithms and evaluate their performance under various circumstances, ground truth about their input needs to be present. Obtaining ground truth from real data would require optimal algorithms to be used, given that those exist. This is not possible since it relies on the very algorithms that are to be evaluated. Using realistic models of the recording situation facilitates the simulation of extracellular recordings. The simulation gives access to a priori known signal characteristics such as spike times and identities. In this paper, we describe a simulator based on a library of spikes obtained from recordings in the cat cerebellum and observed statistics of neuronal behavior during spontaneous activity. The simulator has proved to be useful in the task of generating extracellular recordings with realistic background noise and known ground truth to use in the evaluation of algorithms for spike detection and sorting.

Based on: P. T. Thorbergsson, H. Jorntell, F. Bengtsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Spike Library Based Simulator for Extracellular Single Unit Neuronal Signals", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 6998 – 7001, 2009.

1 Introduction

One of the current promising trends in the field of brain-machine interfaces (BMI) is development toward long term extracellular recordings with chronically implanted multi-electrode arrays (MEA) in the central nervous system (CNS). Detection and classification of spikes are of major importance to successful implementation of a BMI based on extracellular recordings.

The algorithms chosen for spike detection and classification will determine the design criteria for signal acquisition hardware. However, the task of choosing an algorithm is not a trivial one making qualitative evaluation of their performance necessary.

In order to evaluate the performance of algorithms for spike detection and classification under various hardware implementations, we have chosen to implement a simulator to generate extracellular recordings. Simulation gives access to ground truth about spiking activity in the recording and thereby facilitates a quantitative assessment of algorithm performance since the characteristics of the signals are known a priori.

Similar approaches have been taken by others to perform the task of algorithm assessment. However, not many simulators have been fully published, making it necessary for researchers to implement their own versions. Previous works include [1], [2] and [3], where simulators based on the same ideas as ours were used. A fully documented simulator based on analytical models has been published in [4]. In contrast, we have developed a simulator based on large amounts of extracellular recordings that is more readily applicable to our experimental setups.

The simulator described in this paper is fully documented and will be published for general use under a *Creative Commons* license [5] as a library of *MATLAB* functions along with an extensive and expandable spike library. The performance of the simulator has been successfully verified by comparing features such as firing statistics, power spectral density and autocorrelation of simulated and real signals.

2 Models

2.1 Neuronal Distribution and Density

Neuronal density varies among structures in the CNS. We have chosen to use a modified density estimate provided by [6]. In this work, the number of hippocampal CA1 pyramidal cells contained within a cylinder of a given radius was estimated.

Our modification involves replacing the cylinder with a sphere of equal radius but assuming the same number of neurons. The motivation behind this modification is that we want to simulate activity in CNS structures that do not necessarily have the prominent laminar organization that is encountered in the hippocampus [7]. We further assume an isotropic neuronal distribution. Figure 1 shows how the recording environment is modeled.

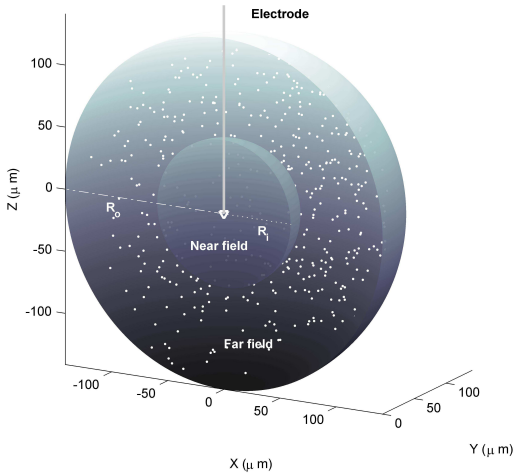


Figure 1: A model of the recording environment. The white dots in the far field represent noise units. Target units are placed in the near field.

2.2 Unit Isolation

In the default setup of the simulator we assume the volume surrounding the electrode to be divided into two parts; "near field" and "far field". The surfaces of the inner and outer spheres shown in Figure 1 bound the two volumes. Spikes coming from neurons within the near field and the far field are referred to as target units and noise units respectively. We assume a small amount of active target units to be present in the near field and we assume those to be separable from the noisy background activity contributed by the noise units in the far field.

2.3 Extracellular Spike Amplitude

The variation in spike shape and amplitude has been studied by [6], [8] and [9]. Our amplitude model for the noise units is based on the result in [9] that at large distances (in the electrode's far field), the amplitude decays as $1/r^n$ where n is between 2 and 3. We have not included the spatially dependent lowpass filtering also described in [9] since the spike prototypes in our library are obtained from actual recordings and are therefore assumed to have undergone this filtering already.

Based on observations of simulated and real signals in combination with the work mentioned above, we model the normalized spike amplitude decay as

$$A = \begin{cases} \frac{1}{(Kr+1)^2} & \text{for noise units} \\ 1 & \text{for target units} \end{cases} \tag{1}$$

where K is a scaling factor that specifies the rate of decay. Within the near field of the electrode we currently assume a constant amplitude of one.

2.4 Inter Spike Interval and Refractory Period

To generate spike times for our target and noise units, we assume a renewal process with gamma distributed inter spike intervals (ISI). An advantage of this assumption is that both the absolute and relative refractory periods are directly implemented in the model [10]. The spike times $\tau_p(n)$ for unit p are thus given by

$$\tau_p(n) = \sum_{j=1}^{n} ISI_j \quad , \quad ISI \sim \Gamma(k, \theta) \tag{2}$$

where k and θ are the shape and scale factors of the gamma distribution respectively. The value of the shape factor varies among units with different mean firing rates $\overline{f}$, but an appropriate value can be obtained by estimating parameters in a real ISI distribution (see Figure 4). By definition of the gamma distribution, the scale factor is determined by the mean ISI, $\overline{ISI}$, and shape factor k

$$\theta = \frac{\overline{ISI}}{k} = \frac{1}{\overline{f}k}. \tag{3}$$

2.5 Noise

We assume that the background noise mainly consists of the sum of scaled spike trains generated by noise neurons in the far field of the electrode. The scaling factor is the same as the amplitude decay in Equation 1. Apart from

the amplitude decay, the noise contributing spike trains are generated in the same way as the target unit spike trains. Instead of assuming a common mean firing rate for all noise units, the firing rate for each noise unit is drawn from a uniform distribution bounded by values given by the user.

We assume thermal noise to be present at the input of the recording amplifier. The root-mean-square (RMS) of the thermal noise can be expressed as

$$\sqrt{\bar{e_n^2}} = \sqrt{4kTRB} \tag{4}$$

where k is Boltzmann's constant, T is the temperature, R is the input resistance of the recording system (electrode and amplifier) and B is the system's bandwidth [11]. The values of those parameters can be adjusted to match an actual setup, but typical values for an implanted system ($T = 310K$, $R = 1M\Omega$, $B = 10kHz$) will give RMS values around $13\mu V$ at the amplifier input.

2.6 Model limitations

The models have limitations in the assumptions of the dynamics and stationarity of the underlying processes. Correlation between different spike trains and bursting activity is not accounted for and we assume constant spike morphologies throughout the duration of the simulation. Further, non-spiking activity (passive signaling [7]) is not accounted for and the assumptions of isotropic neuronal distribution and absence of amplitude decay in the electrode's near field are simplifications worth bearing in mind.

3 Spike Library

Spike waveforms were detected in and extracted from recordings performed in various regions in the cat cerebellum [12] and sorted using the open-source software package *Chronux* [13][14]. Thresholds for spike detection were set automatically using the method described in [1]. The average waveforms were upsampled to 100ksps and stored. Executing this process on an ensemble of recordings containing well isolated single unit activity resulted in a library consisting of 85 different waveforms.

To obtain a qualitative measure of the characteristics of the spike library, we looked at features such as spike duration, frequency contents and general morphology of the stored spikes. The results of the frequency analysis are not shown here since they correlate strongly with spike duration. This examination showed us that the library is sufficient as a basis for modeling the recordings needed for our future algorithm assessment.

We define spike duration as the time period where the absolute amplitude of the largest phase of the spike is above half its peak value. The spike duration histogram in Figure 2 shows that the vast majority of spikes have durations that classifies them as fast spikes [8]. This provides us with an upper bound for testing the algorithms since fast spikes are assumed to pose the biggest challenge to them and is therefore regarded as a desirable feature. Figure 2 also shows five representative spikes from the library.

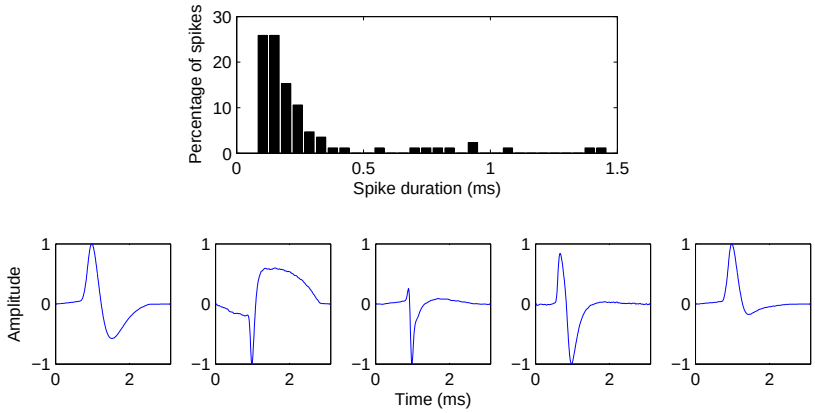


Figure 2: Spike duration histogram (upper) and five spike from the spike library that demonstrate various spike morphologies present in the library (lower).

4 Algorithm

The basic ideas behind the simulation algorithm are inspired by the simulator described in [1]. The algorithm is summarized in Algorithm 1.

The user provides the algorithm with input such as duration of the recording (D), sampling rate (f_s), number of target neurons (N_u), standard deviation of physiological background noise (σ_n), parameters of thermal noise (T, R, B), mean firing rate of target units (f_u), a range of firing rates for noise units (f_n) and rate of amplitude decay in far field (K). For each noise neuron, a firing rate is drawn from a uniform distribution bounded by the given values. In the case of multiple target units, the mean firing rates of the individual units can be set separately.

For each of the target units, spike times are generated (Equation 2) and a random spike waveform is chosen from the spike library. The waveform is then

added to the recording with unchanged amplitude at the obtained spike times.

To generate the background noise, each noise unit is assigned a random position in the far field of the recording electrode (see Figure 1) and a random firing rate is chosen. The amplitude of the unit is then derived from it's distance from the electrode tip (Equation 1). The noise units' spike times are generated in the same way as the target units' and they are added to the recording trace in the same manner as well. White noise is generated according to Equation 4 and added to the recording.

The output of the simulation is the spike times and labels of all (target) spikes in the recording, the simulated recording and background noise as well as the actual waveforms of the target units as taken from the spike library.

Input: Duration of recording, sampling rate, number of target units, standard deviation of physiological noise, thermal noise parameters, mean firing rates, rate of amplitude decay in far field.

Output: Target unit spike times, entire recording, noise component of recording, target unit waveforms.

foreach *Noise/target unit* P **do**

Generate a spike train $s_p(t)$ of N spikes w_k with amplitude A_p (Equation 1) occurring at $\tau_p(n)$ (Equation 2):

$$s_p(t) = A_p \sum_{n=1}^{N} w_k(t - \tau_p(n)) \quad , \quad k \sim U(1, L)$$

where k is the index of the selected spike waveform and L is the number of spikes in the library.

end

Add the spike trains and thermal noise $e(t)$ to obtain the final signal $v(t)$:

$$v(t) = \sum_{p=1}^{P} s_p(t) + e(t)$$

Algorithm 1: The extracellular recording simulator.

5 Verification of Performance

5.1 Methods

To evaluate the performance of the simulator, we selected a set of segments from our recordings and roughly estimated features such as number of separable units, mean firing rates and level of background noise. In order to try to mimic the real recordings, these estimates were used as input parameters to the simulator. Since the modeling of the background noise has proved to be the most challenging task in the implementation, we focused our attention toward segments with low target unit activity and low signal-to-noise ratio (SNR). We then compared the autocorrleation [3] and its Fourier transform, the power spectral density (PSD), for real and simulated signals to get a qualitative assessment of the similarities. The PSD was estimated with Welch's method. Results from both analyses (averages over four segments of data) are shown in Figure 3 to facilitate comparison with results from earlier studies. To evaluate the validity of the assumption of gamma distributed inter spike intervals, we fitted a gamma distribution to inter spike intervals obtained from in-vivo recordings.

To demonstrate the usefulness of the simulator in the task of calculating the probability of detection and false positives in spike detection, we ran a batch of simulated signals through a spike detection algorithm and calculated the resulting probabilities.

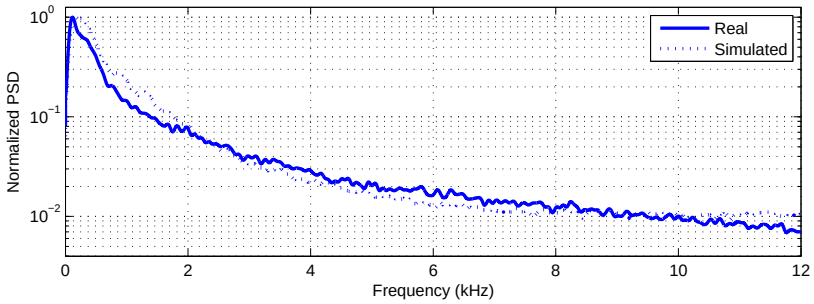
5.2 Results

The comparison between the power spectral densities of the real and simulated signals revealed strong similarities (see Figure 3(a)). The densities resemble those obtained by [8] when studying frequency contents of background activity in extracellular recordings.

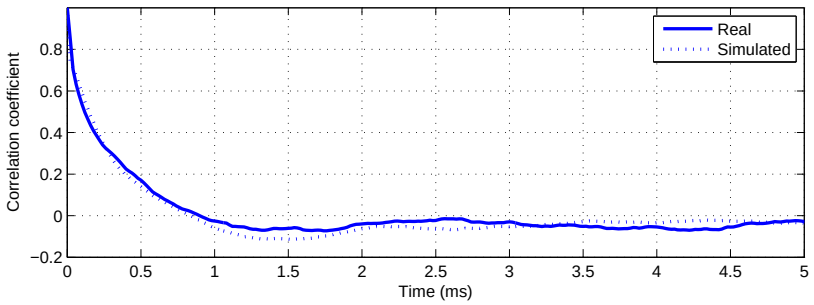
The properties of the PSD are influenced by the modeling of the background noise. Assuming varying activity among neurons and assigning random mean firing rates to the noise neurons gave a good match.

The autocorrelation of the simulated and real signals (Figure 3(b)) showed strong similarities. Reference [3] reported significant autocorrelation at delays up to around 1.2 ms. The shorter interval in our results is mainly caused by the fact that our spike library is dominated by fast spikes. We ran simulations with synthetic spikes of various durations as well and saw a clear connection between the duration of significant autocorrelation and "dominating" spike duration in the library.

Figure 4 shows a histogram of measured ISI during typical spontaneous



(a) Power spectral density of real and simulated recordings. The simulation parameters were $N_u = 5$, $f_u = 10$, $f_n \sim U(1, 50)$, $\sigma_n = 0.2$, $K = 0.05$.



(b) Autocorrelation of real and simulated recordings.

Figure 3: A comparison of real and simulated recordings

activity of a single neuron in the cat cerebellum. The histogram and the fitted gamma distribution show close resemblance and support the assumption of gamma distributed ISI [10].

Figure 5 shows a short segment of a simulated signal and demonstrates the usability when testing spike detection with a threshold crossing criterion. In this case, the probability of detection and false positives was $P_D = 95.35\%$ and $P_{FP} = 4.13\%$ respectively.

6 Conclusions and Future Work

A simulator based on extracellular spikes and observed statistics of neuronal firing has been implemented and tested. The simulator has proved to be useful for providing simulated extracellular recordings to use in the evaluation of

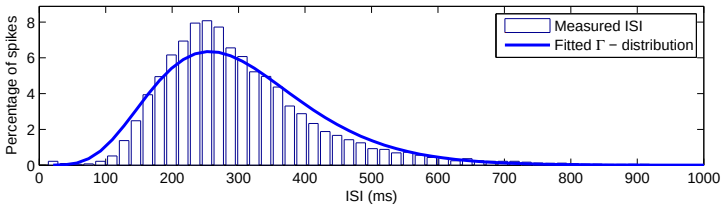


Figure 4: Measured ISI during spontaneous activity of a single neuron. The parameters of the fitted gamma distribution are $k = 6.4$ and $\theta = 0.047$.

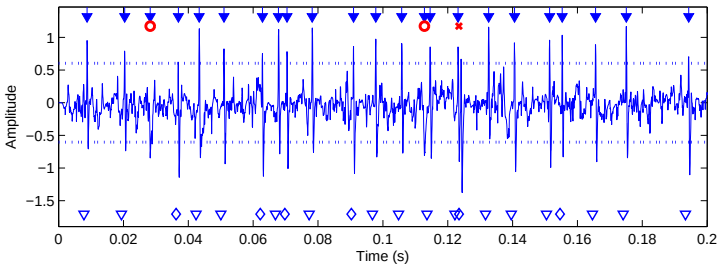


Figure 5: A short segment of a simulated recording. The diamonds ($\diamond$) and triangles (∇) at the bottom indicate the beginning of spikes belonging to two target units present in the recording (ground truth). The arrows at the top indicate detected spike times obtained with a double amplitude threshold (dotted lines). The circles and crosses at the top indicate false positives and missed spikes respectively.

algorithms for spike detection and sorting.

The simulator will be fully published along with an expandable spike library. In [14], some problems behind diverse conventions in methodology are mentioned. We believe that a joint effort would make the resulting research more straight forward and applicable. Our aim is to establish an open venue for researchers to submit their spike libraries and additions to the algorithm. Increased size of the library and more detailed information on specific regions in the CNS will facilitate the simulation of activity in specific areas of the CNS.

Reducing the limitations of the simulator is a work in progress. We plan to investigate appropriate and biologically valid ways of modeling the correlation between target units and implement time-varying firing statistics. These features will be added to the simulator as they come along.

References

[1] R. Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering.," *Neural computation*, vol. 16, pp. 1661–87, Aug. 2004.

[2] S. Gibson, J. W. Judy, and D. Markovic, "Comparison of spike-sorting algorithms for future hardware implementation.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2008, pp. 5015–20, Jan. 2008.

[3] U. Rutishauser, E. M. Schuman, and A. N. Mamelak, "Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo.," *Journal of neuroscience methods*, vol. 154, pp. 204–24, June 2006.

[4] L. S. Smith and N. Mtetwa, "A tool for synthesizing spike trains with realistic interference.," *Journal of neuroscience methods*, vol. 159, pp. 170–80, Jan. 2007.

[5] "Creative Commons - http://creativecommons.org/."

[6] D. a. Henze, Z. Borhegyi, J. Csicsvari, a. Mamiya, K. D. Harris, and G. Buzsáki, "Intracellular features predicted by extracellular recordings in the hippocampus in vivo.," *Journal of neurophysiology*, vol. 84, pp. 390–400, July 2000.

[7] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, *Principles of Neural Science, Fourth Edition*. McGraw-Hill Medical, 4 ed., 2000.

[8] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Variability of extracellular spike waveforms of cortical neurons.," *Journal of neurophysiology*, vol. 76, pp. 3823–33, Dec. 1996.

[9] K. H. Pettersen and G. T. Einevoll, "Amplitude variability and extracellular low-pass filtering of neuronal spikes.," *Biophysical journal*, vol. 94, pp. 784–802, Feb. 2008.

[10] D. Heeger, "Poisson model of spike generation," *Handout, University of Standford*, pp. 1–13, 2000.

[11] R. Pettai, *Noise In Receiving Systems*. New York, NY: Wiley, 1984.

[12] H. Jörntell and C.-F. Ekerot, "Reciprocal bidirectional plasticity of parallel fiber receptive fields in cerebellar Purkinje cells and their afferent interneurons.," *Neuron*, vol. 34, pp. 797–806, May 2002.

[13] "Chronux Analysis Software - http://www.chronux.org."

[14] P. Mitra and H. Bokil, *Observed Brain Dynamics*. Oxford University Press, USA, 2008.

Paper III

Statistical Modelling of Spike Libraries for Simulation of Extracellular Recordings in the Cerebellum

Abstract

Brain machine interfaces with chronically implanted microelectrode arrays for signal acquisition require algorithms for successful detection and classification of neural spikes. During the design of such algorithms, signals with a priori known characteristics need to be present. A common way to establish such signals is to model the recording environment, simulate the recordings and store ground truth about spiking activity for later comparison. In this paper, we present a statistical method to expand the spike libraries that are used in a previously presented simulation tool for the purpose described above. The method has been implemented and shown to successfully provide quick access to a large assembly of synthetic extracellular spikes with realistic characteristics. Simulations of extracellular recordings using synthesized spikes have shown to possess characteristics similar to those of in-vivo recordings in the cat cerebellum.

Based on: P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Statistical Modelling of Spike Libraries for Simulation of Extracellular Recordings in the Cerebellum", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 4250 – 4253, 2010.

1 Introduction

Brain Machine Interfaces (BMIs) are an emerging field within neuroscience. BMIs allow uni-/bidirectional communication with the central nervous system (CNS), facilitating studies of neuronal mechanisms as well as extraction of control signals for operating prosthetic devices. One class of BMIs uses extracellular recordings in the cerebral cortex as their input signals. These recordings are done with chronically implanted electrode arrays connected to external devices for data acquisition and signal processing. A major problem in this type of BMIs is the amount of data obtained from the recordings. This makes it necessary to implement efficient algorithms for extraction of relevant information and thereby reduction of data to be stored or transmitted through the system.

The extracellular recordings consist of two major components; a low frequency local field potential, representing mainly synaptic activity, and high frequency "spiking activity", representing activity of single neurons [1]. Extraction of information from single-unit spiking activity depends on successful detection and classification of spikes. During development of algorithms for these tasks, signals with a priori known characteristics (spike times and classes) are needed. We have previously implemented and reported on a simulator that is based on statistical models for spike times and basic assumptions about the recording environment [2], [3]. The simulator assigns a spike waveform to every contributing neuron and assumes that the waveform does not change during the recording. The waveforms are randomly selected from an assembly of experimentally obtained spikes. Such an assembly is referred to as a spike library. Spike shapes depend on several factors, including type and geometry of the neuron and spatial relationship between the neuron and the recording electrode [4], [5]. Although the original spike library spans a wide range of waveform morphologies, its discrete and sparse nature poses obvious restrictions in this regard.

In this paper we report on a method to expand a spike library to an arbitrary size in order to cover a wider range of waveform morphologies. We find the basis waveforms (principal components) that describe the original spike library and estimate parameters in a statistical model describing their weights. We then use the model to generate new weights that, when applied to the principal components, result in new spike waveforms that follow the statistics of the original data. The method has been shown to be successful in synthesizing an arbitrary number of spike waveforms to use in the simulation of extracellular recordings for testing of spike detection and sorting algorithms.

2 Background

2.1 The Original Spike Library

Spikes were detected in and extracted from several recordings in the cat cerebellum [6]. Spikes from each recording were sorted using *Chronux* [7], [8], ensemble averaging was used for noise reduction and average spike waveforms were stored. The original spike library consists of 85 spike waveforms. In the simulations presented in this paper, some of the spikes in the original library are considered to be outliers due to excessive deviations in location and shape of major waveform landmarks and are therefore discarded in the modelling procedure described here.

2.2 Principal Component Analysis

When performing principal component analysis (PCA) on an ensemble of spikes, we find an orthonormal basis to describe the spikes by applying singular value decomposition (SVD) on the original spike matrix with the mean waveform subtracted from each spike. The output of the analysis are the basis vectors (principal components), their relative contributions to the total variability in the dataset (eigenvalues of the covariance matrix of the data, "latent roots"), and component weights for every spike in the dataset [9]. By using all the principal components, the dataset can be entirely described by

$$\mathbf{S} = \mathbf{PW} \tag{1}$$

where the i-th original spike is in column i of the matrix $\mathbf{S}$, principal component j is in column j of the matrix $\mathbf{P}$ and the weight of principal component j for spike i in column i and row j of the matrix $\mathbf{W}$.

We estimate the number of principal components needed to describe the information contained in the data by looking at the relative contributions of the principal components to the variance in the data. The remaining components are assumed to describe background noise and are discarded. By setting a threshold for what percentage of variance should be considered to contain information, we can automatically find the number of principal components needed do describe the data. Since the number of principal components needed is usually smaller than the number of samples in each spike, this allows us to reduce the dimension of the problem from the original number of samples/spike to the number of principal compoents used to describe the data. This is a commonly used approach in spike sorting, where principal component weights are used as spike features. An approximation of the spike matrix in the first N principal components is

$$\hat{\mathbf{S}} = \mathbf{P}_N\mathbf{W}_N \tag{2}$$

where $\mathbf{P}_N$ contains the first N columns of $\mathbf{P}$ and $\mathbf{W}_N$ contains the first N rows of $\mathbf{W}$ in Equation (1).

2.3 Weight Distributions

The statistics of the spike waveforms can be examined by looking at the distributions of the weights of the first N principal components across the entire original dataset. The distributions of individual component weights can be visualized in a histogram over the rows of $\mathbf{W}_N$ in Equation (2). However, it is assumed that certain combinations of principal components are less likely than others. This motivates us to look at the joint distributions of all component weights and assume that the component weight distribution is described by a Gaussian mixture model in N dimensions and with K mixture components. I.e. the columns of $\mathbf{W}_N$ in Equation (2) are assumed to be stochastic variables coming from an $N-$dimensional $K-$modal Gaussian distribution.

A key assumption of this paper is that the original spike library is a sample drawn from a large population of spikes. This sample can be used to derive information about the statistical properties of the underlying population. By estimating model parameters, we get an idea of what the rest of the spikes in the population might look like and by generating principal component weights according to this model, we can synthesize an arbitrary number of spikes with similar characteristics as the original spikes, and with shapes within the spectrum of "possible" shapes.

3 Methods

3.1 Derivation and Utilization of Model Parameters

Principal component analysis is performed on the original spike library to obtain principal components, component weights and latent roots. The cumulative sum of latent roots is plotted and a variance threshold of 99% is applied to select the number of principal components to use, N. The component weights are fitted to an N-dimensional K-modal Gaussian mixture model using the function *gmdistribution.fit* in *MATLAB*. Since the number of modes, K, is unknown, the parameter estimation is carried out for one to six modes ($K \in [1, 6]$) and the model with the lowest Bayesian information criterion (BIC) is selected. The BIC is used as it favors models with low complexity. The estimated model is used to generate a matrix of random principal component weights, $\tilde{\mathbf{W}}$ and the new spikes are constructed by

$$\tilde{\mathbf{S}} = \mathbf{P}_N \tilde{\mathbf{W}}. \tag{3}$$

The entire procedure is illustrated in Figure (1).

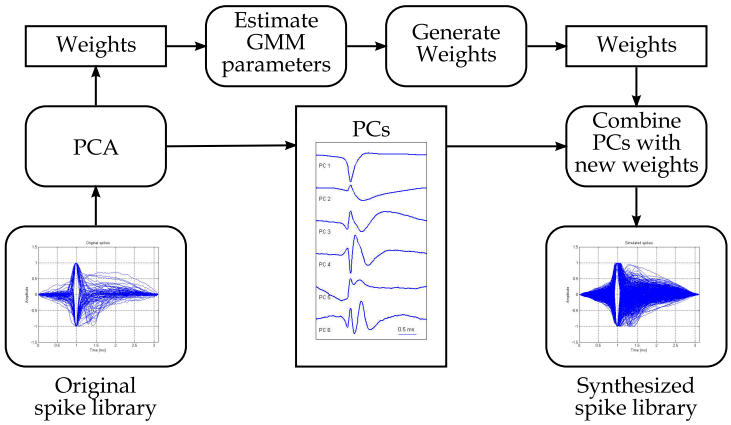


Figure 1: Principal component analysis (PCA) is performed on the original spike library and the parameters of a Gaussian mixture model describing the resulting weight distribution are estimated. The model is used to generate new weights which are applied to the first N principal components, resulting in a synthesized spike with similar characteristics as the original spikes.

3.2 Evaluation of Synthesized Spike Libraries

To evaluate the overall quality of the modelling, we carry out several comparisons between the original and synthesized spike libraries. The library features of interest are distribution of spike durations, distribution of Euclidean interspike distances and sample intensity. The features are examined in histograms across the spike libraries. Usability in simulation of extracellular recordings is evaluated by running simulations in EAPSim [2], [3] with a real and synthesized spike library and comparing the power spectral densities of the simulated recordings. General appearance of spikes is evaluated in a double blind test on neuroscientists with long experience in working with spike data.

Feature Comparison: Original vs. Synthesized Spike Libraries

We define spike duration as the time period during which the absolute amplitude of the largest phase of the spike is above half its peak value [2]. Spike duration is calculated for all spikes in the real and synthesized spike library and the distributions are compared in histograms.

Euclidean interspike distance between spikes s_i and s_j is defined as

$$d_{i,j} = \sqrt{\sum_{m=1}^{M} (s_i(m) - s_j(m))^2}. \quad (4)$$

where m is the sample index. For each spike library, the distance is calculated for every spike pair in that library.

We define sample intensity as the histogram across every row of the spike matrices $\mathbf{S}$ and $\tilde{\mathbf{S}}$. Sample intensity provides a qualitative measure of the range of spike morphologies spanned by a spike library. A similar measure has previously been used in [7] to visualize dominating spike waveforms in an assembly of spikes in spike sorting.

Evaluation of Simulated Recordings

Four sets of simulated extracellular recordings are generated with EAPSim [2], [3]. Each set consists of five recordings. In two sets, we use the original spike library and in two sets, we use a synthesized library with 2000 spike waveforms. The sets have zero and four target units present respectively. Power spectral density (PSD) is estimated for all recordings using Welch's method and the mean of the PSDs of all recordings at a given setting is compared between the datasets.

Double Blind Test

To evaluate the quality of synthesized spikes with respect to general appearance, we present two experienced neuroscientists with a double blind test. A 9×10 matrix of spike figures, each showing either an original or synthesized spike, is shown to the subjects and they are asked to identify synthesized spikes. For each of the spike figures, we first select (with equal probability) either the original or synthesized library. We then select (without replacement) a random spike from that library. The only information given to the subjects is that each figure either shows an original or synthesized spike. The results are evaluated with the *VassarStats* statistical tool [10].

4 Results

4.1 Model Parameters

Figure (2) shows the relative contribution of the first N principal components to the variance in the original spike library. A 99% variance threshold is ap-

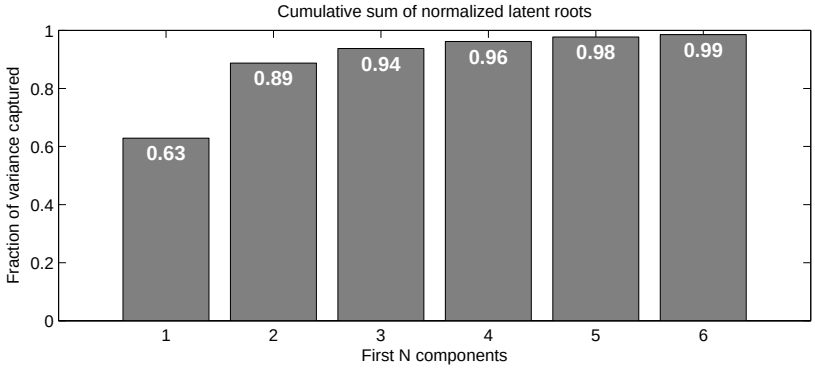


Figure 2: The relative contribution of the first N principal components to the variance in the original spike library. $N = 6$ principal components capture 99% of the variance in the data.

plied and we conclude that the first six principal components capture 99% of the variance in the data. According to the Bayesian information criterion, we model the weight distribution with two components ($K = 2$). As a result, the weight distribution is assumed to be described by a 6-dimensional 2-component Gaussian mixture model. Table (1) summarizes the results from the parameter estimation. Figure (3) shows the first six principal components and their individual weight distributions in the original spike library.

4.2 Feature Comparison: Original vs. Synthesized Spike Libraries

Feature Comparison

Figure (4) shows the comparison between features of the original and synthesized spike libraries. A qualitative analysis of the figures shows that we obtain close matches between original and synthesized spike libraries in all cases.

Evaluation of Simulated Recordings

Figure (5) shows means of power spectral densities for five simulated recordings with four target units, using original and synthesized spike libraries. The results for background noise only (zero target units) are very similar and are not shown here. In [2], we showed that a good match in power spectral densities of simulated and in vivo recordings could be obtained with our original

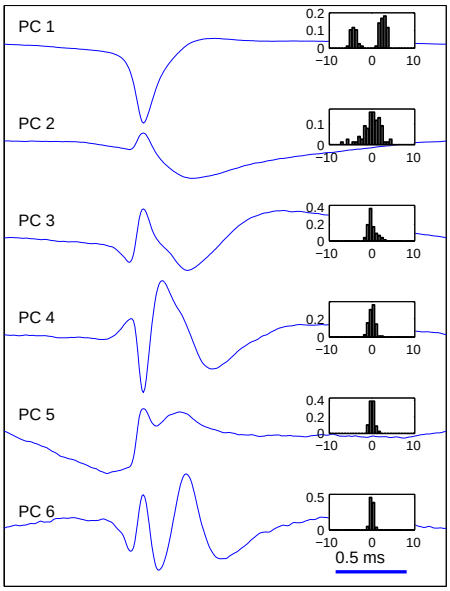


Figure 3: The first 6 components and their original weight distributions (inset histograms).

spike library. The close match between the curves in Figure (5) among with the previously mentioned observations shows that realistic spectral features in simulated extracellular recordings can be obtained even when using simulated spike libraries.

Double Blind Test

Analysis of the double blind tests show, within a 95% confidence interval, that none of the subjects performed significantly better than chance when discriminating between original and synthesized spikes.

5 Conclusions and Future Work

In this paper, we have described a method in which we use principal component analysis to obtain a statistical model to describe the waveforms in an experimentally obtained spike library. The statistical model, among with the originally obtained principal components, is used to synthesize a spike library of arbitrary size. Our results show that the modelling and synthesis result

Table 1: Gaussian MixtureModel Parameters

Parameter	Component 1
Mixing proportion	0.62
Mean	$\begin{bmatrix} 2.66 & -0.11 & -0.08 & 0.08 & -0.02 & -0.01 \end{bmatrix}$
Covariance matrix	$\begin{bmatrix} 0.46 & 0.67 & 0.27 & -0.23 & 0.15 & 0.14 \\ 0.67 & 5.29 & -0.64 & -0.49 & 0.00 & 0.13 \\ 0.27 & -0.64 & 0.57 & -0.05 & 0.16 & 0.11 \\ -0.23 & -0.49 & -0.05 & 0.19 & 0.00 & -0.05 \\ 0.15 & 0.00 & 0.16 & 0.00 & 0.29 & 0.00 \\ 0.14 & 0.13 & 0.11 & -0.05 & 0.00 & 0.08 \end{bmatrix}$
Parameter	Component 2
Mixing proportion	0.38
Mean	$\begin{bmatrix} -4.4 & 0.18 & 0.13 & -0.12 & 0.03 & 0.02 \end{bmatrix}$
Covariance matrix	$\begin{bmatrix} 0.53 & 0.18 & 0.46 & -0.49 & -0.03 & -0.05 \\ 0.18 & 4.49 & 1.02 & 0.84 & -0.01 & -0.22 \\ 0.46 & 1.02 & 1.62 & 0.11 & -0.27 & -0.19 \\ -0.49 & 0.84 & 0.11 & 0.9 & 0.01 & 0.09 \\ -0.03 & -0.01 & -0.27 & 0.01 & 0.31 & 0.00 \\ -0.05 & -0.22 & -0.19 & 0.09 & 0.00 & 0.29 \end{bmatrix}$

in spikes with realistic features, usable for realistic simulation of extracellular recordings in the cerebellum.

The model will be implemented into EAPSim [3] for common use. By allowing principal component weights to move within the modelled distribution, we can model variations in spike shapes over time within or between recordings. These variations would facilitate studies on algorithms for spike tracking and spike sorting under dynamic conditions.

Our results show that six principal components are sufficient to describe 99% of the variance in the original spike library of cerebellum recordings. This result is consistent with the independent results reported in [11] where the authors performed principal component analysis on a large ensamble of spikes coming from different neurons and concluded that 99% of the variance was described by the first six principal components. These results give us reason to suspect that spike sorting algorithms with correlation against a constant set of basis shapes (PCs) might be feasible.

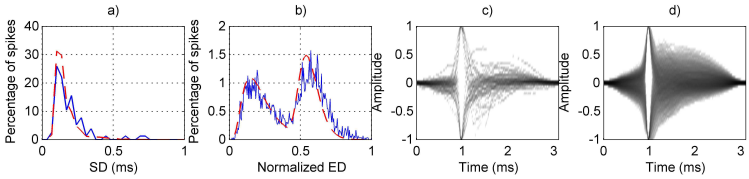


Figure 4: Feature comparison between the original (whole lines in a) and b)) and synthesized (dashed lines in a) and b)) spike libraries. Figures a) and b) show distributions of spike durations and interspike distances respectively. Figures c) and d) show sample intensity for the original and synthesized spike libraries respectively.

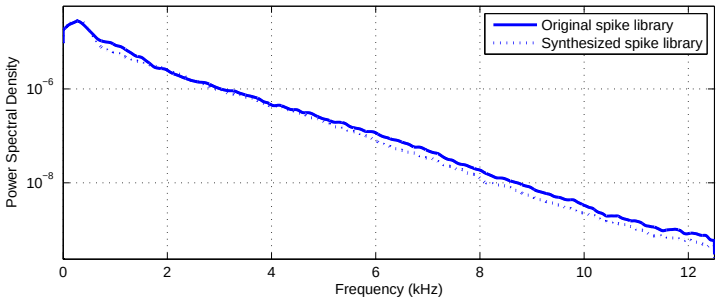


Figure 5: Power spectral density of simulated extracellular recordings with original and simulated spike libraries.

References

[1] H. S. Bokil, B. Pesaran, R. a. Andersen, and P. P. Mitra, "A method for detection and classification of events in neural activity.," *IEEE transactions on bio-medical engineering*, vol. 53, pp. 1678–87, Aug. 2006.

[2] P. T. Thorbergsson, H. Jorntell, F. Bengtsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Spike library based simulator for extracellular single unit neuronal signals.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2009, pp. 6998–7001, Jan. 2009.

[3] "EAPSim Website - http://www.eit.lth.se/staff/Palmi_Thor.Thorbergsson."

[4] D. a. Henze, Z. Borhegyi, J. Csicsvari, a. Mamiya, K. D. Harris, and G. Buzsáki, "Intracellular features predicted by extracellular recordings in the hippocampus in vivo.," *Journal of neurophysiology*, vol. 84, pp. 390–400, July 2000.

[5] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Variability of extracellular spike waveforms of cortical neurons.," *Journal of neurophysiology*, vol. 76, pp. 3823–33, Dec. 1996.

[6] H. Jörntell and C.-F. Ekerot, "Reciprocal bidirectional plasticity of parallel fiber receptive fields in cerebellar Purkinje cells and their afferent interneurons.," *Neuron*, vol. 34, pp. 797–806, May 2002.

[7] "Chronux Analysis Software - http://www.chronux.org."

[8] P. Mitra and H. Bokil, *Observed Brain Dynamics*. Oxford University Press, USA, 2008.

[9] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials.," *Network (Bristol, England)*, vol. 9, pp. R53–78, Nov. 1998.

[10] "Vassarstats Clinical Calculator 1 - http://vassarstats.net/clin1.html."

[11] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-Gaussian variability.," *Journal of neuroscience methods*, vol. 69, pp. 175–88, Nov. 1996.

Paper IV

Minimizing data transfer with sustained performance in wireless brain-machine interfaces

Abstract

Brain machine interfaces (BMIs) may be used to investigate neural mechanisms or to treat the symptoms of neurological disease and are hence powerful tools in research and clinical practice. Wireless BMIs add flexibility to both types of applications by reducing movement restrictions and risks associated with transcutaneous leads. However, since wireless implementations are typically limited in terms of transmission capacity and energy resources, the major challenge faced by their designers is to combine high performance with adaptations to limited resources. Here, we have identified three key steps in dealing with this challenge: 1) the purpose of the BMI should be clearly specified with regard to the type of information to be processed, 2) the amount of raw input data needed to fulfill the purpose should be determined, in order to avoid over- or under dimensioning of the design, 3) processing tasks should be allocated among the system parts such that all of them are utilized optimally with respect to computational power, wireless link capacity and raw input data requirements. We have focused on step 2) under the assumption that the purpose of the BMI (step 1) is to assess single- or multiunit neuronal activity in the central nervous system with single channel extracellular recordings. The reliability of this assessment depends on performance in detection and sorting of spikes. We have therefore performed absolute threshold spike detection and spike sorting with principal component analysis and fuzzy c-means on a set of synthetic extracellular recordings while varying sampling rate and resolution, noise level and number of target units, and used known ground truth to quantitatively estimate the performance. From the calculated performance curves, we have identified sampling rate and resolution breakpoints, beyond which performance is not expected to increase by more than 1-5%. We have then estimated the performance of alternative algorithms for spike detection and spike sorting in order to examine the generalizability of our results to other algorithms. Our findings indicate that the minimization of recording noise is the primary factor to consider in the design process. In most cases there are breakpoints for sampling rates and resolution that provide guidelines for BMI designers in terms of minimum amount raw input data that guarantees sustained performance. Such guidelines are

essential during system dimensioning. Based on these findings we conclude by presenting a quantitative task allocation scheme that can be followed to achieve optimal utilization of available resources.

Based on: P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Minimizing data transfer with sustained performance in wireless brain-machine interfaces", *Journal of neural engineering*, pp. 036005, 2012.

1 Introduction

The substantial progress made within the field of neuroscience in recent decades depends in part on the development of brain-machine interfaces (BMIs). By providing a connection between the central nervous system (CNS) and the external world, BMIs constitute powerful tools for both the investigation of neural mechanisms in awake, freely moving animals and for clinical applications [1, 2, 3]. Many of the approaches to BMI implementation to date rely on wired connections between the different parts of the system [4, 5], which restrict the subject from moving around and are associated with increased risk for infections and other complications due to transcutaneous leads. The implementation of wireless communication links within the BMI (Figure 1 (a)) solves many of these problems. However, with an ever increasing number of recording channels [6], designers of wireless BMIs are challenged with increased demands, which must be met by careful system dimensioning.

The primary building blocks of a BMI are the acquisition unit, the processing unit and the actuation unit (Figure 1(a)). The acquisition unit acquires a signal that carries information about ongoing neural activity within the CNS and forwards it to the processing unit. The role of the processing unit is generally to extract that information from the acquired signal. The output from the processing unit is passed on to the actuation unit that executes commands based on the measured neural activity. The actuation unit provides feedback to the subject, e.g. electrical stimulation of nerves or muscles, the driving of a prosthetic limb or wheelchair or the operation of a personal computer. In research applications, the researcher makes use of either the raw (unprocessed) acquired signal or the processed signal from any of the stages within the processing unit. In a wireless BMI, the acquisition unit is implanted along with some or all parts of the processing unit and the wireless link.

Typically, the signal is acquired through implanted microelectrodes [7]. The recorded signal then consists of background noise reflecting the activity of distant neurons, low frequency local field potentials representing synaptic activity and a higher frequency spiking activity, representing APs from close-by neurons [8], here referred to as target units. When acquisition is performed in this manner, the processing unit typically extracts information about the compound activity of a group of neurons (multi unit activity) by detecting spikes in the acquired signal [9], or the firing patterns of the individual neurons (single unit activity) by sorting the spikes upon their detection [10] (Figure 1(b)).

From this point on, we assume that the acquisition unit digitizes the signal at a certain sampling rate (samples/second) and sampling resolution (bits/sample) and that all signal processing is performed on the digitized signal in the processing unit. We also assume that the objective of the processing

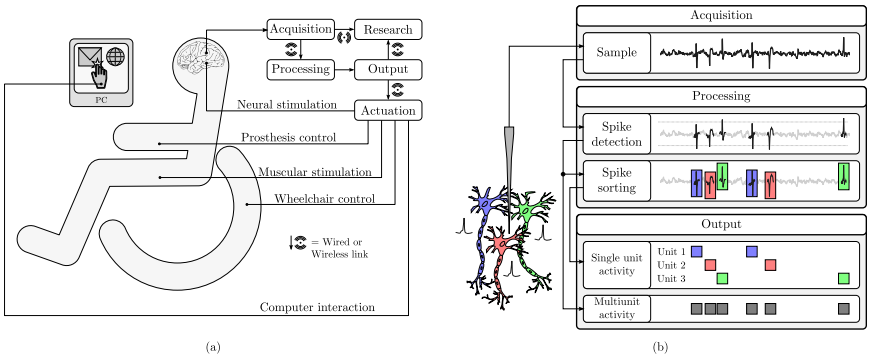


Figure 1: (a) A brain-machine interface (BMI), acquires signals from the central nervous system (CNS), processes them and forwards the output to an actuator or to a research application. The actuator provides feedback that is modulated by the acquired neural activity. The feedback can be in the form of neural stimulation, prosthetic limb control, muscle stimulation, wheelchair operation or the operation of a personal computer. Wireless links between the parts of a BMI are an attractive solution to problems related with decreased freedom of movement and increased risk for post surgical complications that are often associated with wired BMIs. The wireless link(s) can be implemented in different parts of the system, depending on how processing tasks are allocated among the system parts. (b) Implanting microelectrodes into the extracellular space of the CNS is a common way of acquiring signals in BMI applications. Extracellular recordings in the CNS can be used to assess the gathered activity of a group of neurons close to the electrode (multi unit activity) or that of the individual neurons (single unit activity). The processing steps involved are spike detection and spike sorting.

unit is to extract information about neuronal firing, i.e. single or multi unit activity. The amount of information needed, i.e. the accuracy in the detection and sorting of spikes, depends on the application. For instance, in applications involving the control of prosthetic limbs, BMIs can be trained to work with a fairly high error tolerance but in applications where the objective is to perform analysis of detailed network functions, high accuracy is of key importance. However, both applications profit from increased performance in terms of accuracy in detection and sorting of spikes.

During the design of a wireless BMI, several competing goals need to be carefully considered and balanced. Reliable assessment of neuronal firing patterns requires accurate detection and sorting of spikes. In turn, accuracy in

spike detection and sorting relies on high-quality recordings that capture an adequate amount of information. However, wireless links are highly power-consuming and limited in data transmission capacity compared to their wired counterparts. Hence, the balancing act of the wireless BMI designer involves minimizing the amount of raw data passed to the wireless link without jeopardizing information transfer in terms of accurate assessment of neuronal firing patterns, and without excessively burdening the resources available on the implanted part of the system.

This leads us to identifying three crucial steps in the process of designing BMIs:

1. **Specify the purpose of the BMI**
   What type of information should the processing unit extract?

2. **Determine the amount of data needed**
   How much raw data needs to be acquired for the output of the processing unit to be as accurate as possible? I.e. what sampling rate and resolution need to be employed in the acquisition unit to ensure maximum achievable performance in a given recording situation?

3. **Allocate processing tasks**
   How should processing tasks be allocated among the system parts in order to obtain a balance between performance and resource utilization and to avoid over- or underdimensioning?

For wireless BMIs, these steps aid in obtaining a balanced design as described above and for wired BMIs, they help in minimizing the amount of data to be stored for off-line analysis. Such a minimization is beneficial both in terms of lowering the cost of data storage and reducing post-processing time. Step 3 is of particular importance in the wireless case due to the distributed system configuration that obviously characterizes wireless BMIs. Figure 2 illustrates the demands posed by the processing task allocation onto the individual parts of the BMI. Increasing processing task allocation on the implant decreases the demand on the wireless link, but increases the demands on computational resources, and vice versa.

There has been great interest in the development of wireless BMIs in recent years and the implementations suggested so far differ in terms of all three key steps mentioned above. These include the transmission of the raw signals from a large number of channels, either in analog [11, 12, 13] or digital form [14], the transmission of timestamps representing the occurrence of spikes in the signal [15], the transmission of digitized spike waveforms [16, 17] and neuronal assignment labels of sorted spikes [18].

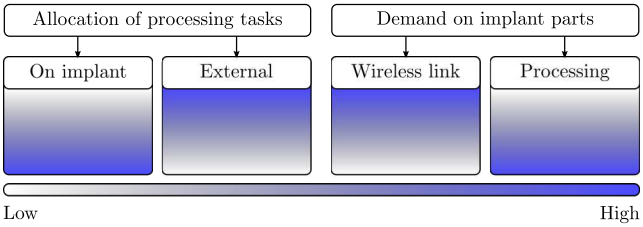


Figure 2: The allocation of processing tasks among the parts of a wireless BMI and its influence on the demands placed on the implanted parts of the system need to be considered during BMI design. High/low color intensities indicate high/low task allocation and high/low demand on the implant's parts (wireless link and processing part). Given a purpose for the BMI, the minimum amount of data needed to fulfill the purpose and the resources available, processing tasks should be allocated among the parts of the BMI in a way that ensures high output reliability and optimum utilization of the resources at hand.

Despite the diversity among wireless BMI implementations, arising mostly from differences in BMI purposes (step 1) and task allocation (step 3), current knowledge of the actual need for raw input data maximum achievable performance (step 2) is sparse. Efforts have been made to quantify these requirements, which are reported in [8, 19, 20, 21, 22]. All of these focused on the spike sorting step, using various estimations of spike waveform bandwidth and performance measures to conclude that spike sorting required a sampling rate between 16 and 30 kHz and a sampling resolution above 4 bits. Sampling resolution was only considered in one of the studies [21]. Two of the studies [21, 22] considered the influence of waveform misalignments introduced by spike detection jitter, i.e. when spikes from the same neuron are sampled at different times and concluded that it gave rise to spurious clusters in the spike feature space. In [22], this effect was assumed to be significant at sampling rates below 10kHz, but in [21] the limit was found to be at 30kHz.

Although the investigations summarized here are important steps in the direction of balanced BMI designs, they differ somewhat in their conclusions and are incomplete in the sense that they do not fully address the entire processing chain, i.e. both spike detection and spike sorting. There is still a need for systematic, quantitative and objective approaches to exploring the influence of the amount of raw input data on the reliability of the processing unit's output and the minimum amount of data needed to ensure maximum achievable performance in a given situation. This need becomes even greater when the number of recording channels increases.

The contributions of this work can be divided into four parts. First, we have performed a detailed estimation of the performance of spike detection with an absolute threshold and spike sorting with principal component analysis (feature extraction) and fuzzy c-means (clustering) as a function of sampling rate and resolution, noise level and number of target units. These estimations provide a description of the relationship between the amount of raw input data and the accuracy of the processing unit's output under various circumstances. Second, we have quantitatively identified sampling rate and resolution "breakpoints" at which performance falls below 1% to 5% below the maximum achievable performance in each case. The breakpoints provide an estimation of the minimum amount of raw input data needed to ensure minimal loss of accuracy in the assessment of single and multi unit activity. Third, in order to see how the results from these analyses generalize to other algorithms, and to explore the effects on performance introduced by increasing the amount of computational resources, we have performed the same type of performance analysis for other algorithms (detection with nonlinear energy operator, stationary wavelet transform and matched-filter based detection, and spike sorting with discrete wavelet transform), but at a lower level of detail. The results from these preliminary analyses show that the algorithms in many cases have similar dependencies on sampling rate and resolution. Our results for absolute value threshold detection and principal component analysis with fuzzy c-means thus allow us to draw qualitative conclusions about the amount of raw data required by the other algorithms for accurate processing unit output. Last, we have summarized our findings in a diagram that relates the allocation of processing tasks to the demands on the wireless link and the computational resources at hand. Our results provide useful guidelines for the selection of sampling rate and resolution when dimensioning wireless BMIs.

2 Methods

2.1 Synthesized Recordings

The simulator described in [23] was used to synthesize 150 extracellular neural recordings, each with a duration of 30 seconds. A library of 2000 synthetic spikes was used for the synthesis [24]. The recordings were divided into 3 sets of 50 recordings each, containing 1, 2 and 4 target units respectively. The target unit spike waveforms were chosen randomly from the spike library. Ground truth (true spike times, identities and waveforms) was stored with all recordings. Target unit spike amplitudes were drawn from a uniform (rectangular) distribution between 0.5 and 1.0 ($U(0.5, 1.0)$) and were then normalized with

respect to the largest one. The strongest unit in each recording thereby had a amplitude of 1 (unipolar), assumed to correspond to 100 μV. In all recordings, target units had a mean firing rate of 20 spikes/second and gamma distributed inter-spike intervals.

White noise was added to represent thermal noise generated in the recording electronics. The thermal noise amplitude was set according to

$$v_{t_{rms}} = \sqrt{4kTBR} \tag{1}$$

where k is Boltzmann's constant, T is the temperature and B and R are the bandwidth and input resistance of the analog front end, respectively [25]. With parameters set to $T = 310K$, $B = 50$ kHz and $R = 200$ kΩ, this gave an input referred noise of 13 μV_{rms}, or 0.13 when normalized to the largest spike amplitude. Physiological background noise was synthesized by simulating the spiking activity of distant neurons with mean firing rates chosen randomly from a uniform distribution between 1 and 50 spikes/second. Signal components (target unit activity, physiological background noise and thermal noise) were stored separately, allowing us to scale the physiological background noise to any desired level and study its influence on processing unit output. Physiological background noise was referred to in terms of its standard deviation, σ_N, normalized to the maximum spike amplitude. Figure (3) shows an example of a synthesized recording, sampled at 25 kHz, with 2 target units and physiological background noise scaled to 0.05, 0.1 and 0.2 respectively prior to downsampling.

Dynamic recording properties such as varying noise level, spike amplitude, number of target units etc. were left out of the simulations in order to ensure objectivity and full control over simulation variables. Real recordings with dynamic properties can be modeled as a continuous series of stationary recording segments. Thus, the results presented here can be applied to the dynamic case by considering either the "worst segment" or the "mean segment" in the recording.

The recordings were synthesized at a sampling rate of 100 kHz and a sampling resolution determined by machine precision. When exploring the effects of sampling rate and resolution on the performance in spike detection and spike sorting (see following sections), the recordings were downsampled using *MATLAB*'s function `resample`, that includes antialiasing filtering. Sampling resolution was defined in terms of the least significant bit (LSB) amplitude relative to the maximum spike amplitude of 1, and was varied by discretizing the sample values as multiples of the LSB amplitude with *MATLAB*'s function `quant`. The number of effective bits for each sampling resolution was then calculated as

$$N_b = \log_2(V_{max}/V_{LSB}) \tag{2}$$

Table 1: 150 recordings with 1, 2 and 4 (50 each) target units were synthesized and stored along with ground truth about neuronal activity. Ground truth was used to quantify the performance in spike detection and spike sorting while altering simulation variables according to the table values.

Simulation Variables	**Values**
Number of target units (N_u)	1, 2, 4
Target unit amplitude	0.5 - 1.0[a]
Std. deviation of background noise (σ_N)	0.01 - 0.3[b]
Std. deviation of thermal noise ($v_{t_{rms}}$)	0.13[c]
Sampling rate (f_s)	0.5 - 100 kHz
Sampling resolution (Q_s)	1 - 14 bits[d]

[a] The largest unit has an amplitude of 1, corresponding to 100 μV

[b] Relative to a spike amplitude of 100μV.

[c] Relative to a spike amplitude of 100 μV at a bandwidth of 50kHz

[d] Effective bits, assuming full utilization of the ADCs dynamic range of $\pm$ 1 relative to a spike amplitude of 100 μV

where N_b is the number of effective bits, V_{max} is the analog-to-digital converter's (ADC's) dynamic range and V_{LSB} is the LSB amplitude. For the sake of generalization, we assumed the dynamic range of the analog-to-digital converter to be fully utilized and hence set V_{max} to 2 ($\pm$1), i.e. the maximum peak-to-peak spike amplitude. The breakpoints obtained and presented in our results can hence be adapted to a real situation in which the ADC's dynamic range is larger than the spike amplitude by adding one extra bit of sampling resolution for every doubling of the dynamic range relative to the maximum spike amplitude. Table (1) provides an overview of the simulation variables and their covered ranges.

2.2 Spike Detection

Several spike detection algorithms have been described in the literature, varying in both complexity and performance. Some of these that have been described within the context of low-power, low-complexity BMI applications, are *ABSolute value threshold detection* (ABS) [26], *Nonlinear Energy Operator detection* (NEO) [27], *Stationary Wavelet Transform Product detection* (SWTP) [28] and *Matched-Filter based detection* (MF) [29].

In [30] the authors compared ABS, NEO and SWTP and concluded that NEO provided the highest accuracy, ABS the second highest (only slightly below NEO) and SWTP the lowest. In terms of chip area and computational

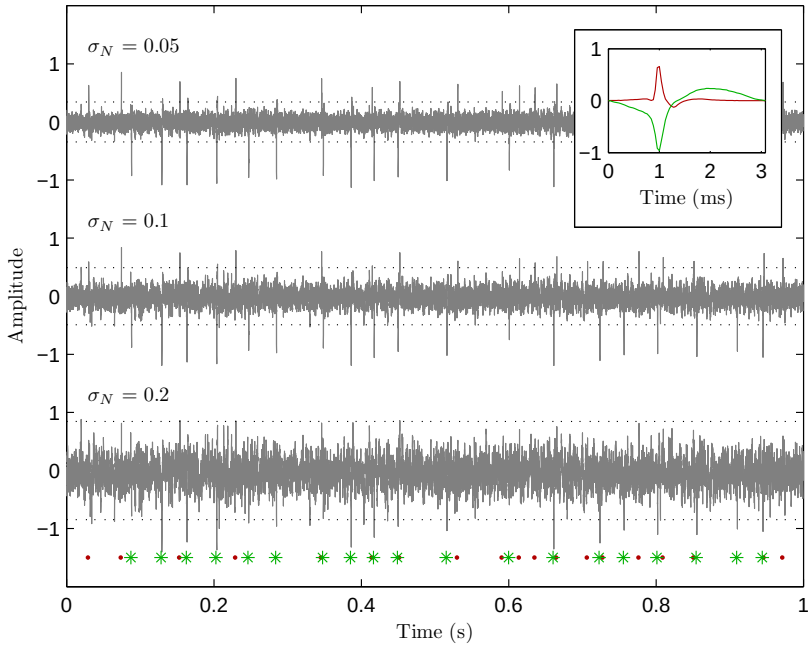


Figure 3: A synthesized recording with two target units and physiological background noise scaled to three different levels (σ_N = standard deviation of physiological background noise). Ground truth (true spike waveforms, target unit identities and spike times) stored with the synthesized recordings allowed for quantitative assessment of performance in spike detection and spike sorting. True spike times of the two units are shown along the lower edge of the figure ("•" and "⋆") and the true spike waveforms are shown in the inset. The dotted lines represent spike detection thresholds for absolute value threshold detection, obtained with Equation (3).

complexity, ABS had the lowest cost, NEO the second and SWTP the highest. In [29] the authors compared ABS, NEO and MF (along with others) and concluded that ABS was as effective as NEO and MF. In terms of computational complexity, ABS had the lowest cost, NEO the second and MF the highest. Their results also indicated that in order to improve spike detection, maximizing signal-to-noise ratio (SNR) was superior to employing advanced preprocessing.

We stress the fact that the main focus of this work is on wireless autonomous implants with limited resources, making the choice of fully automatic low-

complexity algorithms for signal processing essential. With this in mind, and considering the small differences in reported performance but significant differences in complexity between the proposed algorithms, we chose to perform the identification of sampling rate and resolution breakpoints on absolute value threshold detection; that being a feasible choice in terms of both performance and complexity. We also carried out a preliminary analysis of the performance of the other algorithms. These are briefly described in Section 2.4.

In ABS, spike detection is performed by applying a threshold of

$$T = 4\hat{\sigma}_N = 4 \cdot \text{median}\left\{\frac{|v|}{0.6745}\right\} \tag{3}$$

where $\hat{\sigma}_N$ is the estimated standard deviation of background noise and $|v|$ is the absolute value of the digitized signal's amplitude [26].

Estimation of Spike Detection Accuracy

Detected spike times (sample indices of threshold crossings) were compared to the ground truth (spike onset and duration) and the results were used to quantify performance in terms of true and false positive detection rates, as explained in Figure 4. A recovery time of 2 ms, during which a new spike detection could not occur, was applied after each detection. From the true spike onsets and durations, time frames of true spike occurrences were established. Every detected spike time was then compared with the true spike time frames and detection scores were assigned according to the number of true spike time frames, N, covering the detected spike time:

Case 1: No true spike time frame ($N = 0$)
Number of false positives, N_{FP} increased by 1.

Case 2: Exactly 1 true spike time frame ($N = 1$)
Detection score $d = 1$ assigned to the true spike whose true time frame covered the detection time.

Case 3: More than one true spike time frames ($N > 1$)
Detection score $d = 1/N$ assigned to the true spikes whose true time frames covered the detection time.

In case of a double detection, i.e. two detection times being covered by the same true spike time frame, the second detection was classified as a false positive detection (Case 2'). However, in case the second of these two detection times was covered by yet another true spike time frame, the first true spike did not influence the treatment of the second (Case 3').

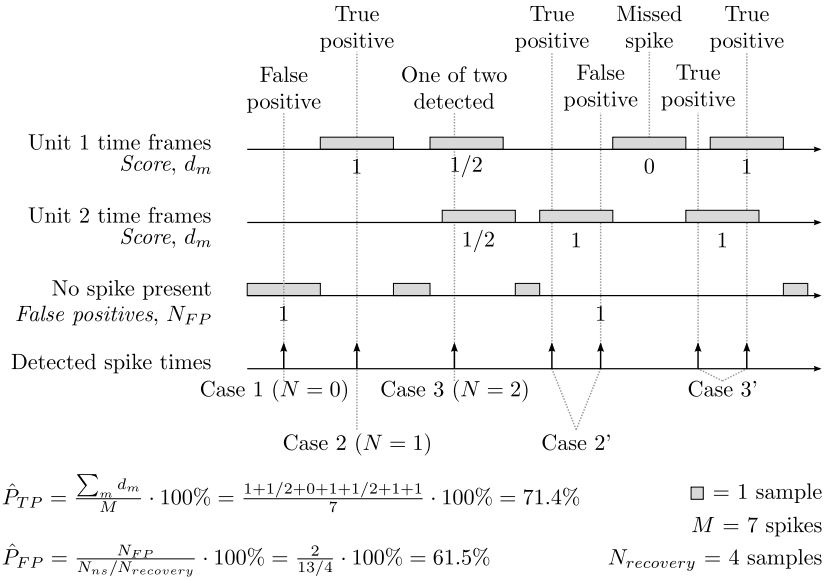


Figure 4: Spike detection performance was quantified by comparing spike detection times with the known presence of spikes according to the ground truth stored during recording synthesis (spike onset and duration). Three main cases were identified: 1) the detected spike time was not covered by any true spike, 2) the detected spike time was covered by exactly $N = 1$ true spike and 3) the detected spike time was covered by $N > 1$ true spikes. In case 1, the detection was classified as a false positive detection. In case 2, the detection was classified as a true positive detection and a detection score of $d_m = 1$ was assigned to the true spike. In case 3, the detection was classified as a true positive detection with weight $1/N$ and detection scores of $d_m = 1/N$ were assigned to the true spikes. In rare cases, 2 detections could occur during the same true spike interval. In such cases, only the first detection was classified as a true positive detection and the second was classified as a false positive detection (case 2'). An exception from this was made when the second detection was covered by yet another true spike, in which case the first detection did not influence the treatment of the second (case 3').

Probability of true positive detection, P_{TP}, was estimated by the rate of true positive detection, $\hat{P}_{TP}$ and was calculated according to

$$\hat{P}_{TP} = \frac{\sum_{m=1}^{M} d_m}{M} \cdot 100\% \tag{4}$$

where M is the number of true spikes in the recording and d_m is the detection score assigned to true spike m, obtained by the above procedure. A detection score of zero was assigned to missed spikes.

Probability of false positive detection, P_{FP}, was estimated by the rate of false positive detections, $\hat{P}_{FP}$, defined as and calculated according to

$$\hat{P}_{FP} = \frac{N_{FP}}{N_{ns}/N_{recovery}} \cdot 100\% \tag{5}$$

where N_{FP} is the total number of false positive detections obtained by the above procedure, N_{ns} is the number of samples in the recording not covered by any true spike time frame and $N_{recovery}$ is the recovery time of the spike detector. The ratio $N_{ns}/N_{recovery}$ is hence a measure of the maximum achievable number of false positive detections in the recording.

Sampling Rate and Resolution Breakpoints for Absolute Value Threshold Detection

When identifying sampling rate breakpoints, the true positive detection rate, $\hat{P}_{TP}$, was interpolated (cubic spline) and thresholds of 1% and 5% below $\hat{P}_{TP}$ at full sampling rate were applied. The sampling rate breakpoint for each of the thresholds was taken as the sampling rate at which the true positive detection rate curve crossed the threshold. Two different thresholds were applied since the breakpoint locations varied strongly with the error tolerance, especially at high noise levels. Since the detection threshold was adapted to the noise level (Equation 3) to avoid false positive detections, the false positive detection rate was consistently negligible compared to the true positive detection rate. Therefore, it was neglected when identifying sampling rate breakpoints. Sampling resolution was held fixed at a level determined by machine precision while sampling rate was varied.

For sampling resolution, we included both true and false positive detection rate in the breakpoint estimation since, with lowered resolution, both true and false positive detection rates became significantly unstable. The instability was caused by rounding of both the signal and the detection threshold (see Section 3.1). We applied thresholds of 1% below the true positive detection rate at full sampling resolution and above the false positive detection rate at full sampling resolution to identify the sampling resolution at which the onset of the instability occurred. Thus we obtained two sampling resolution breakpoints, one at the resolution where true positive detection rate dropped below its threshold and one at the resolution where false positive detection rate rose above its threshold. The final sampling resolution breakpoint was taken

as the higher resolution breakpoint of these two. Sampling rate was held fixed at 25 kHz while sampling resolution was varied.

2.3 Spike Sorting and Feature Extraction

Although there has been great interest in the task of spike sorting, no standard algorithm exists and the algorithms proposed differ in terms of feature extraction as well as clustering approaches. We considered *Principal Component Analysis* (PCA) [10], the *Discrete Wavelet Transform* (DWT) [26] and *Discrete Derivatives* (DD) [30, 31] for feature extraction and *Fuzzy C-means* (FCM) [32] for clustering.

In [26], the authors compared PCA and DWT and concluded that DWT was superior in performance. This conclusion was mainly based on the observation that although the first principal components capture the majority of the variance in the original data, they do not necessarily provide the best cluster separation. In [33], the authors also compared PCA and DWT and also concluded that DWT was superior, but only in cases where it was well tuned to the data. In addition, they concluded that the DWT suffered from difficulties in finding the best wavelet-parameters to be used in clustering. In [30], the authors compared PCA, DWT and DD, in terms of both performance and computational complexity. They concluded that there was little difference in performance, but a significant difference in computational complexity – DD being the simplest and PCA the most complex. In [31], the performance of PCA, DWT and DD was compared. The differences varied somewhat between the cases studied, and from the results, it seems difficult to derive a general conclusion regarding the ranking of the methods across all cases. From a computational complexity point of view, DD should be a feasible alternative. However, it suffers from the same problems as DWT regarding selection of features and the selection of time scales to calculate slopes is arbitrary.

From the above, we conclude that there is no obvious best choice in these comparisons. However, PCA is well defined, well established and a widely used algorithm for feature extraction [30]. It is not the simplest in most computational terms, but it is generic in the sense that it has no case specific parameters that need to be set. Therefore it is straightforward and suitable to use for quantitative analysis over a wide range of cases. The other algorithms that were considered for preliminary analysis are described briefly in Section 2.4.

In PCA, an ordered set of orthonormal basis waveforms (principal components), that describe the variation in the set of spike waveforms presented to the algorithm, is found. The spike waveforms can then be completely described as linear combinations of the basis waveforms and the weights applied to the basis waveforms can be used as spike features in spike sorting. Since the set of

basis waveforms is ordered – the first waveform describing the most variation – dimensionality reduction is achieved by simply selecting the first N weights as spike features. N was set to 3 in our simulations.

Estimation of Spike Sorting Accuracy

True spike times and durations, obtained from ground truth, were used to extract spikes from synthesized recordings with 2 and 4 target units at full sampling rate (100 kHz). This was done in order to exclude any errors possibly induced in the spike detection step.

Spikes were sorted using fuzzy c-means clustering (FCM) [32] and were assigned to the cluster with the strongest membership grade. To ensure consistency in the processing, we assumed the number of target units to be known and provided it to the clustering algorithm as input (number of clusters). We are aware that such an assumption does not reflect reality, but we assume it to be valid for the sake of comparing clustering results while varying the simulation parameters of interest. The probability of correct spike classification, P_{TP}, was estimated as the correct classification rate, $\hat{P}_{TP}$, by comparing the cluster assignments with the true spike identities according to ground truth, as explained by examples in Figure 5.

Correct classification rate, $\hat{P}_{TP}$, was calculated by hierarchical examination of the evidence for mappings between cluster i and target unit j. Such an approach was necessary since the mappings between cluster identities and target units is non-deterministic. An evidence matrix, E, of size $N_u \times N_u$, where N_u is the number of target units, was constructed. Entry $e_{i,j}$ in the evidence matrix was set to the number of spikes truly coming from target unit j, assigned to cluster i by FCM. The evidence matrix was then processed according to the following procedure:

1. Set hierarchical level counter, n, to 1.
2. Find the $n-$th largest value, e_{max_n} in the evidence matrix, representing the $n-th$ largest evidence for mapping between a target unit and a cluster, assuming that evidence on higher hierarchical levels is true.
3. Set all values in the row and column of e_{max_n} to zero (eliminate evidence assumed to be false).
4. Increase hierarchical level counter, n, by 1.
5. Repeat from step 2 until the number of non-zero entries in the processed evidence matrix is lower than or equal to N_u.

Example 1:

$$E = \begin{bmatrix} e^{\star}_{1,1} & e_{1,2} & e_{1,3} & e_{1,4} \\ e_{2,1} & e_{2,2} & e_{2,3} & e_{2,4} \\ e_{3,1} & e_{3,2} & e_{3,3} & e_{3,4} \\ e_{4,1} & e_{4,2} & e_{4,3} & e_{4,4} \end{bmatrix} = \begin{bmatrix} \boxed{25} & 0 & 0 & 0 \\ 0 & 0 & 0 & \boxed{25} \\ 0 & 0 & \boxed{25} & 0 \\ 0 & \boxed{25} & 0 & 0 \end{bmatrix}$$

$\hat{P}_{TP} = \frac{25+25+25+25}{100} \cdot 100\% = 100\%$

* $e_{i,j}$ = Number of spikes in cluster i with true identity j

---

Example 2:

$$E = \begin{bmatrix} 15 & 1 & 3 & 4 \\ 5 & 1 & 7 & 8 \\ 2 & 1 & 10 & 9 \\ 3 & \boxed{21} & 5 & 4 \end{bmatrix} \longrightarrow \begin{bmatrix} \boxed{15} & 0 & 3 & 4 \\ 5 & 0 & 7 & 8 \\ 2 & 0 & 10 & 9 \\ 0 & 21 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 0 & 7 & 8 \\ 0 & 0 & \boxed{10} & 9 \\ 0 & 21 & 0 & 0 \end{bmatrix} \longrightarrow \begin{bmatrix} 15 & 0 & 0 & 0 \\ 0 & 0 & 0 & \boxed{8} \\ 0 & 0 & 10 & 0 \\ 0 & 21 & 0 & 0 \end{bmatrix}$$

$\hat{P}_{TP} = \frac{21+15+10+8}{100} \cdot 100\% = 54\%$

Figure 5: Spike sorting performance was measured as the percentage of correctly classified spikes, $\hat{P}_{TP}$, as obtained by the procedure described by examples 1 and 2. In both examples, 100 spikes coming from 4 units (25 from each) have been sorted with fuzzy c-means with the number of units (4) given as input. After clustering, the assigned cluster identities are compared with the true unit identities according to ground truth and the evidence matrix E is constructed. The entries of the evidence matrix, $e_{i,j}$ represent the number of spikes in cluster i belonging to unit j according to ground truth. In Example 1, all spikes have been correctly classified and cluster identities 1, 2, 3 and 4 correspond to target unit identities 1, 4, 3 and 2 respectively. This demonstrates the non-deterministic mapping between cluster and target unit identities described in the text. In Example 2, some of the spikes have been misclassified. Comparing this with Example 1, this can be described as "leakage" between the entries of the evidence matrix. "Leakage" between rows represents spikes coming from the same unit being classified as coming from different units (cluster splitting) and "leakage" between columns represents spikes coming from different units being classified as the same unit (cluster merging). The evidence matrix is processed in a hierarchical manner (arrows) by successively finding the strongest mappings between units and clusters, assuming previously found evidence to be true.

After completing this procedure, the number of correctly classified spikes was taken as the sum of the elements in the processed evidence matrix and the rate of correct classification, $\hat{P}_{TP}$ was taken as the ratio between that number and the total number of spikes present in the recording. Since fuzzy c-means is a non-deterministic algorithm, we performed the clustering 10 times for each case and kept the result with the largest rate of correct classification only.

Sampling Rate and Resolution Breakpoints for Spike Sorting with PCA and FCM

We applied a threshold of 1% below the correct classification rate at full sampling rate or resolution and the breakpoints were taken as the sampling rate or resolution where the curves fell below the threshold. In the sampling rate case, the curves were interpolated (cubic spline) prior to applying the threshold. When varying sampling rate, sampling resolution was held fixed at the original value determined by machine precision. When varying sampling resolution, the recordings where first downsampled to a fixed sampling rate of 25 kHz.

PCA Feature Space Representation as a Quantitative Measure of Performance

Quantifying spike sorting performance relies on some subjective assumptions (see Section 2.3) regarding the number of clusters and the choice of method for assessing the sorting results. Therefore, we found that using an objective measure that neglects the spike sorting algorithm and only considers the information provided to it would be a sound approach. Such a measure would greatly simplify the work of characterizing system behavior in terms of spike sorting performance.

We therefore examined the development of the feature space representation (PCA) of spikes as an objective, indirect measure of spike sorting performance. The measure was taken as the mean squared error (MSE) between the normalized feature matrix (first 3 principal component weights) at a given sampling rate or resolution and the normalized feature matrix at full sampling rate or resolution. Our expectation was that a convergence in the feature space representation as sampling rate or resolution was increased would lead to a convergence in spike sorting results, independent of the clusterability of the spikes. Therefore, optimization of performance in terms of the feature matrix MSE instead of spike sorting accuracy should be a valid and feasible approach. However, since this measure is not easily interpreted in terms of quality of the system output, namely the accuracy in single unit activity assessment, it is still victim to subjectivity when setting performance thresholds for breakpoint esti-

mation. Therefore, we chose to only look at the correlation coefficient between the correct classification rate and the MSE at this stage as a first step in employing such a measure in future studies. The correlation coefficient indicates the strength of the linear relationship between the two measures.

Effects of Spike Detection Jitter on Spike Sorting

Spike sorting and feature extraction analyses for PCA described so far were carried out in both the presence and absence of simulated spike detection jitter (Section 1). Spike detection jitter was simulated by applying a random time shift to every spike waveform once the waveforms were extracted and downsampled. The magnitude of the time shift is limited by the sample period and was hence drawn from a uniform (rectangular) distribution between -1/2 and 1/2 a sample period. Therefore, the effects of spike detection jitter were expected to be reduced as sampling rate was increased. Additional spike detection jitter caused by noise was not taken into account.

The effect of spike detection jitter on spike sorting performance was quantified as the difference between correct classification rate with and without jitter. The breakpoint was taken as the sampling rate at which the difference dropped below a threshold of 1%. The influence of spike detection jitter was examined in the same manner while altering sampling resolution.

Joint Effects of Sampling Rate and Resolution on Spike Sorting Performance

To account for any possible effects on spike sorting performance caused by a combination of employing both a low sampling rate and resolution, we calculated correct classification rate while jointly varying sampling rate and resolution on a subset of the values spanned when varying the individual parameters as described in previous sections. We assumed spike detection and extraction to have been performed on the signal at a sampling rate of 25 kHz, which is high enough for the effects of spike detection jitter to be negligible (Figure 11). We therefore did not include spike detection jitter in this part. We then calculated the correct classification rate, $\hat{P}_{TP}$, as a function of sampling rate and resolution at all noise levels studied for 2 and 4 target units. A threshold of 1% below correct classification rate at full sampling rate and resolution was applied and the breakpoint was taken as the point, in the region where $\hat{P}_{TP}$ was above the threshold, that minimized bitrate (sampling rate $\times$ sampling resolution, bits/second).

2.4 Generalization to Other Algorithms

In order to investigate how the sampling rate and resolution breakpoints obtained by the above analysis would generalize to the other algorithms considered (NEO, SWTP and MF for spike detection and DWT for spike sorting), we performed preliminary analysis on the influence of varying sampling rate and resolution on their performance. This preliminary analysis also provided insight into how an increase in available computational resources, and thus the possibility of employing more demanding algorithms, would influence the requirement for raw input data to the processing unit. These analyses were performed in the same way as described for ABS and PCA in Sections 2.2 and 2.3. The alternative algorithms considered are briefly described here. We chose not to implement DD for spike sorting in our analysis due to the arbitrariness in the choice of time scales for slope calculation.

Spike Detection

Nonlinear Energy Operator:

A threshold T is applied to the nonlinear energy operator $\Psi(n)$ for the signal $v(n)$. $\Psi(n)$ is obtained by

$$\Psi(n) = v^2(n) - v(n+1) \cdot v(n-1). \tag{6}$$

and T is set as

$$T = C\overline{\Psi(n)} \tag{7}$$

where $\overline{\Psi(n)}$ is the mean of the $\Psi(n)$ and C is a scaling factor adjusted empirically and then used as a constant [27]. C was set to 20 in our simulations.

Stationary Wavelet Transform Product

The stationary wavelet transform (SWT) of the signal $v(n)$ is calculated at 5 consecutive dyadic scales ($W(2^j, n)$, $j \in [1, 5]$). The scale with the maximum sum of absolute values is found ($2^{j_{max}}$) and the point-wise product $P(n)$ of wavelet coefficients over three consecutive scales up to $2^{j_{max}}$ is calculated as

$$P(n) = \prod_{j=j_{max}-2}^{j_{max}} |W(2^j, n)|. \tag{8}$$

$P(n)$ is then smoothed by convolution with a Bartlett window $w(n)$ (half the spike length) and a threshold T is applied to the smoothed $P_s(n)$. T is set as

$$T = C\overline{P_s(n)} \tag{9}$$

where C is a scaling factor and $\overline{P_s(n)}$ is the mean of $P_s(n)$ [28]. C was set to 2 in our simulations.

Matched Filter

The signal is convolved with a spike waveform that is the mean of spikes in the spike library used to generate the synthetic recordings (Section 2.1). A threshold T is applied to the absolute value of the resulting signal. T is set to a percentage of the input range of the (convolved) signal. T was set to 20% in our simulations.

Spike Sorting

Discrete Wavelet Transform:

The wavelet transform of each spike is calculated using a four-level multiresolution decomposition with Haar wavelets. The wavelet coefficients thus describe the spike waveforms at various scales and times [26]. The number of wavelet coefficients is the same as the number of samples in the spike waveforms. In contrast to PCA, the coefficients are not ordered and the set of coefficients used in clustering needs to be identified. Thus, every coefficient distribution needs to be tested and coefficient selection is based on the obtained test statistics.

Modified Lilliefors Test:

The Lilliefors test for normality employed in [26] provides a measure of deviation from normality for the coefficient distributions. In order to further reward coefficients with multimodal distributions, we modified the test to include a measure of separation between multiple modes. The empirical cumulative distribution function of the coefficients $ECDF(c)$ is calculated and compared to the cumulative distribution function of a normal distribution $NCDF(c)$ with the same mean and variance as the coefficient distribution. The test statistic is then taken as

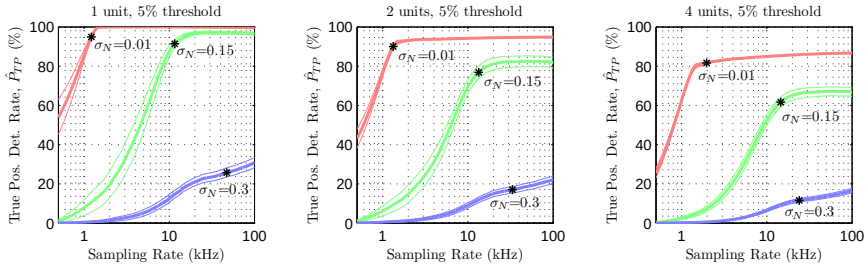
$$K = |\Delta_p| + ||d||/||d_{max}|| \tag{10}$$

where Δ_{lm} is the distance between the first and last peak of the function $d = |ECDF(c) - NCDF(c)|$ and $||.||$ denotes Euclidean norm. d_{max} is set to be $|0.5 - NCDF(c)|$. Thus, a large mode separation and large deviation from normality (first and second term of Equation 10 respectively) lead to a high test statistic. The coefficients with the three highest test statistics are selected for feature extraction.

Maximum-Difference Test:
The coefficients with the three largest variations are selected for feature extraction (see [30] for details on implementation).

3 Results

3.1 Spike Detection with Absolute Value Threshold



(a) (b) (c)

Figure 6: True and false positive detection rates, $\hat{P}_{TP}$ and $\hat{P}_{FP}$, were used to quantify spike detection performance while varying the number of target units present (1, 2 and 4), standard deviation of background noise ($\sigma_N = 0.01$ to 0.3), sampling rate (0.5 to 100 kHz) and sampling resolution (1 to 14 bits). (a) to (c) show the mean true positive detection rates (across all recordings) as functions of sampling rate at three of the noise levels studied for 1, 2 and 4 target units. Thresholds of 1 and 5% below true positive detection rate at full sampling rate were applied to find performace breakpoints. The stars indicate the location of the 5% breakpoints obtained for the mean curves. (d) to (f) and (g) to (i) show the mean true and false positive detection rates respectively as functions of sampling resolution. Thresholds of 1% below true positive detection rate at full sampling resolution and above 1% false positive detection rate at full resolution were applied and the overall sampling resolution breakpoint was taken as the higher one of the two resulting breakpoints. Breakpoints obtained for the mean curves are indicated with stars. 95% confidence intervals for the mean are indicated with thin lines. (Continued on page 132)

Figure 6 shows mean true and false positive detection rates (across all recordings) as functions of sampling rate and resolution at a subset of the noise levels studied (false positive detection rate as a function of sampling rate is not shown). The mean curves illustrate the general relationship between spike

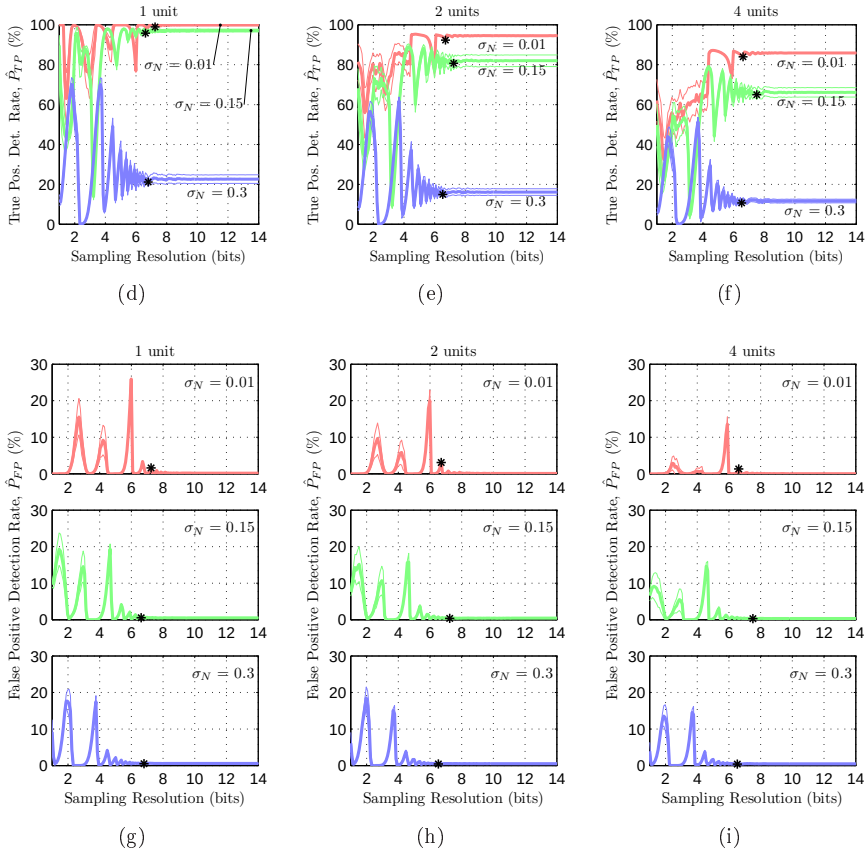


Figure 6: True and false positive detection rates for spike detection with absolute value threshold detection as functions of noise level, number of target units and sampling resolution (continued from page 131).

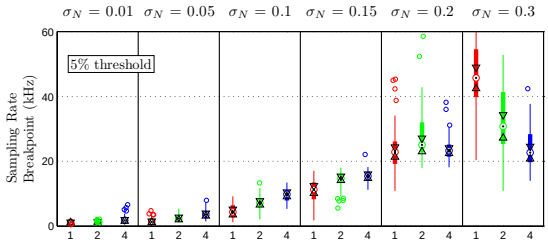
detection performance and sampling rate, sampling resolution, noise level and number of target units. Breakpoints obtained from the mean curves according to the procedure described in Section 2.2 are shown in the figure. Close to the sampling rate breakpoints, we generally observed a "knee" or plateauing in performance. As noise level increased, the knee in the true positive detection rate curves for sampling rate generally became less clear. Close to the sampling resolution breakpoints, we observed an increased instability in performance. This behavior is explained by the fact that as sampling resolution is lowered, the

signal-to-noise ratio switches between being enhanced and degraded as some resolutions lead to noise being suppressed (rounded downwards) and spikes being enhanced (rounded upwards), and some lead to both being either enhanced or suppressed. For absolute value threshold detection, this behavior is amplified since the detection threshold is locked to the same quantization levels as the signal (the threshold is obtained from signal median). As a result, performance was actually better at some low resolutions than at full resolution. However, since sampling resolution was varied in terms of least significant bit amplitude relative to the maximum spike amplitude, a change in either spike or noise amplitude would cause a change in the effective sampling resolution and thereby also a potentially dramatic change in performance when operating at these low resolutions.

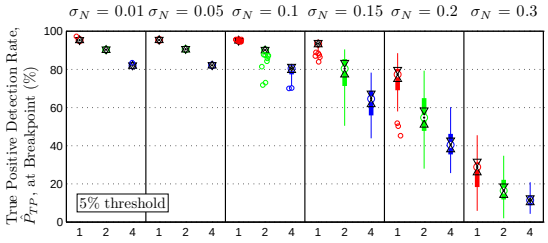
We recall that we assume a system architecture where signal digitization is performed in the acquisition unit and all processing is performed on the digitized signal. One way of excluding the threshold's contribution to the oscillating behavior is to consider an alternative architecture, in which the detection threshold would be either analog or set with a higher resolution than that employed for signal acquisition. In order to see the effects of such an alternative architecture, we tried using the threshold obtained from the signal at full resolution and observed that the onset of the oscillating behavior consistently occurred at a lower resolution. In other words, such an architecture would be more robust to lowered sampling resolution than the one studied here. With the intention to study a worst case scenario, we only present the results for the original thresholds (sampling resolution dependent), since it consistently overestimates the sampling resolution breakpoint compared to the full-resolution threshold.

Figure 7 shows the distributions of sampling rate and resolution breakpoints for all cases studied. The 1% sampling rate breakpoints are omitted in the figure. At the high-end noise levels, their locations were significantly higher than the 5% breakpoints due to the diminishing plateauing of performance curves. The 5% breakpoint was observed to provide a closer estimate of the curve knees. Maximum achievable spike detection performance decreased with increased noise level and number of target units. The maximum true positive detection rate deteriorated significantly at noise levels of $\sigma_N = 0.1$ to 0.15 (Figure 7(b)).

Sampling rate breakpoints were obtained at 1.4 to 90.7 kHz and 1.2 to 49.0 kHz (upper limit of 5% median comparison interval) for the 1% and 5% thresholds respectively, depending on noise level and number of target units present (Figure 7(a)). Lowering the threshold (error tolerance) from 5% to 1% hence caused a noticable increase in the demands with respect to sampling rate. However, in the cases where the highest of those breakpoints were encountered, the



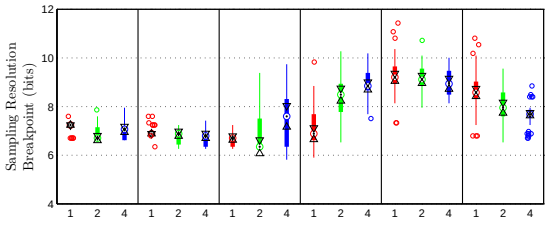
(a)



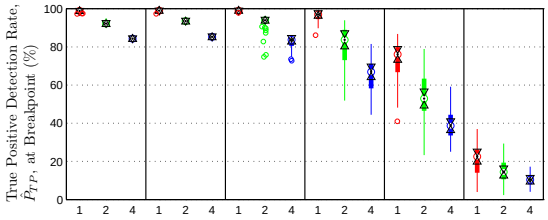
(b)

Figure 7: The distributions of sampling rate and resolution breakpoints for spike detection with an absolute threshold. (a) and (b) show sampling rate breakpoints and true positive detection rate at the breakpoints for the 5% threshold. (c) and (d) show sampling resolution breakpoints and true positive detection rate at the breakpoints. The number of target units present in each case is indicated with 1, 2 and 4. σ_N is the standard deviation of the physiological background noise in each case. Median comparison intervals ($p = 0.05$) are marked with triangles ($\triangle/\triangledown$). (Continued on page 135)

maximum achievable true positive detection rate suffered a significant decrease due to the increased noise level, as mentioned above. Therefore, the major consideration in these cases was not which sampling rate to employ, but whether the noise level could be lowered by any means. Assuming an upper limit of 0.15 on the noise level, we would obtain maximum sampling rate breakpoints of 30.6 kHz and 16.1 kHz for the 1% and 5% thresholds respectively ($\sigma_N = 0.15$, 4 target units). Sampling rate breakpoints were generally shifted upwards with increasing noise level and number of target units, except at high noise levels, where the breakpoints were lowered when increasing the number of target units. This is explained by the differences in the plateauing behavior of the performance curves at high noise levels (see Figure 6).



(c)



(d)

Figure 7: The distributions of sampling rate and resolution breakpoints for spike detection with an absolute threshold (continued from page 134).

Sampling resolution breakpoints were obtained at 6.8 bits to 9.3 bits (upper limit of 5% median comparison interval) (Figure 7(c)). The noise dependency of breakpoints was somewhat inconsistent, displaying a local maximum within the range of noise levels studied. However, this inconsistency was observed for all numbers of target units. Again, assuming a maximum noise level of $\sigma_N = 0.15$, a maximum sampling resolution breakpoint of 9.0 bits was obtained.

3.2 Spike Sorting with Principal Component Analysis

Spike Sorting

Figure 8 shows the correct classification rate, feature matrix mean squared error, feature space representation and cluster assignments as functions of sampling rate and resolution for a representative example recording with 4 target units at a noise level of $\sigma_N = 0.01$, both in the presence and absence of spike detection jitter.

Figure 9 shows sampling rate and resolution breakpoint distributions for spike sorting for all cases studied. Maximum correct classification rate was robust to an increase in noise level up to $\sigma_N = 0.3$ when 2 target units were

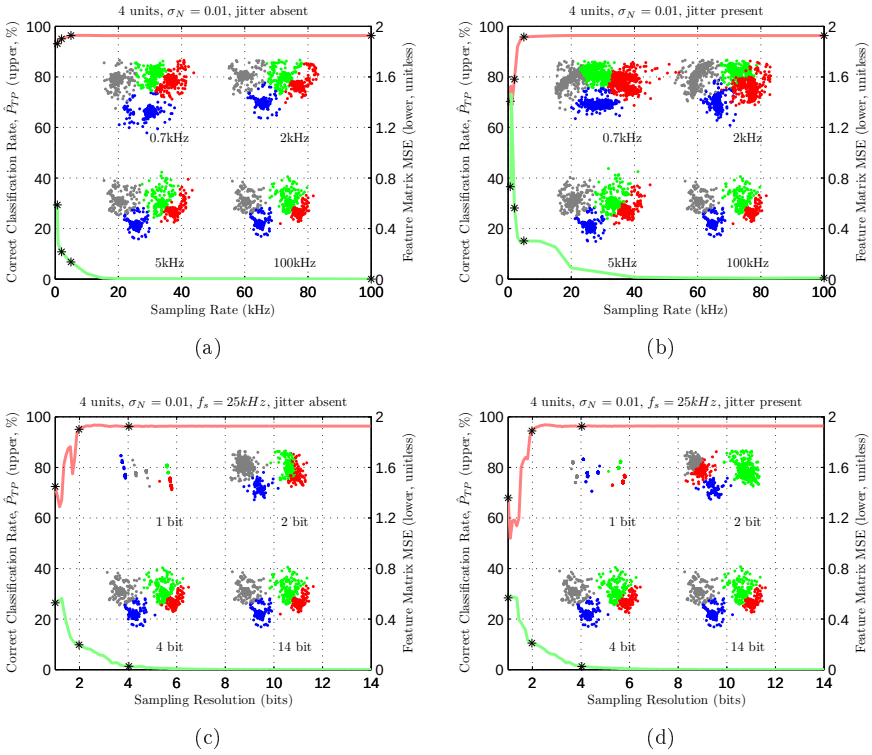


Figure 8: Performance in spike sorting and development of feature space representation were quantified as the correct classification rate ($\hat{P}_{TP}$, upper curve, red), and the feature matrix (principal component weights) mean squared error (MSE, lower curve, green) respectively. The feature matrix MSE at a given sampling rate or resolution was taken as the mean squared error between the feature matrix at that specific sampling rate or resolution and the feature matrix at full sampling rate or resolution. (a) and (b) show the $\hat{P}_{TP}$ and MSE as functions of sampling rate in the absence and presence of spike detection jitter respectively for one representative recording with 4 target units at noise level of $\sigma_N = 0.01$. (c) and (d) show the same, but as functions of sampling resolution. The insets show snapshots of the feature space representation of the spikes and cluster assignments after spike sorting with fuzzy c-means (red, green, blue, gray). The snapshot locations on the sampling rate and resolution axes are marked with stars on the $\hat{P}_{TP}$ and MSE curves.

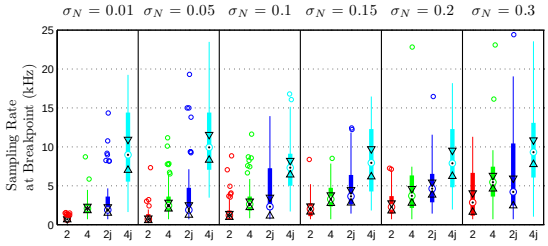
present, both in the presence and absence of spike detection jitter (Figure 9(b)). When 4 target units were present, a significant decrease in the maximum correct classification rate was seen already at a noise level of $\sigma_N = 0.05$. The presence of spike detection jitter caused no significant change in maximum correct classification rate. This is not surprising when considering that at the maximum sampling rate (100 kHz), spike detection jitter has no significant influence on spike sorting performance (Section 3.2, Figure 11).

Sampling rate breakpoints between 0.7 kHz to 11.6 kHz (upper limit of 5% median comparison interval) were obtained, depending on the number of target units and the presence of spike detection jitter (Figure 9(a)). Increasing noise level and the number of target units and introducing spike detection jitter generally caused an upward shift in the breakpoints. Differences in breakpoint locations per case across noise levels were generally insignificant. However, removing spike detection jitter generally caused a significant downward shift in breakpoint locations. Achieving spike alignment before the spike sorting step by any means (adequate sampling rate in the spike detection step or realignment in post-processing) therefore proved to be an important factor in lowering sampling rate breakpoints for spike sorting.

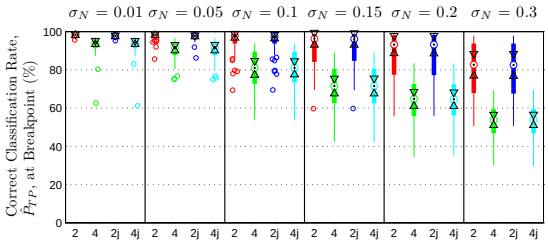
Sampling resolution breakpoints were obtained between 1.4 bits and 3.1 bits (upper limit of 5% median comparison interval), depending on noise level and the number of target units (Figure 9(c)). Spike detection jitter did not have significant influence on the location of sampling resolution breakpoints. Again, sampling resolution was varied while keeping sampling rate fixed at 25 kHz, high enough for the spike misalignments caused by spike detection jitter to have negligible effects on spike sorting performance (Section 3.2, Figure 11). As noise level was increased, the difference between the cases studied was generally reduced. Maximum correct classification rates (Figure 9(d)) were essentially the same as in the sampling rate part, again due to the employment of the sampling rate of 25 kHz (9(a)).

PCA Feature Space Representation as a Quantitative Measure of Spike Sorting Performance

Comparing the correct classification rate curves and feature matrix mean squared error curves as exemplified in Figure 8, we saw that in general, the correct classification rate seemed to converge faster than the feature space mean squared error. The correlation between the feature matrix MSE and the correct classification rate, $\hat{P}_{TP}$, was found to be significant ($p = 0.05$) in 97.2% and 92.3% of the recordings studied for sampling rate and resolution respectively. The median correlation over all recordings per case (noise level, number of target units and presence/absence of spike detection jitter) was significantly



(a)



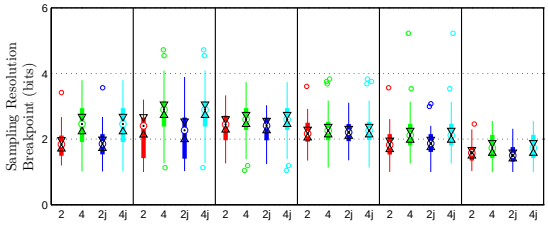
(b)

Figure 9: The distributions of sampling rate and resolution breakpoints for spike sorting with principal component analysis and fuzzy c-means. (a) and (b) show sampling rate breakpoints and correct classification rate at the breakpoints. (c) and (d) show sampling resolution breakpoints and correct classification rate at the breakpoints. The number of target units present in each case is indicated with 2 and 4 and the presence of spike detection jitter is indicated with an appended "j". σ_N is the standard deviation of physiological background noise in each case. Median comparison intervals ($p = 0.05$) are marked with triangles ($\triangle/\triangledown$). (Continued on page 139)

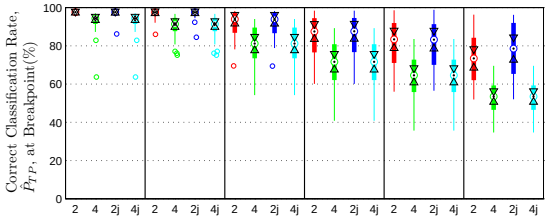
($p = 0.05$) different from 0 in all cases. Although clustering with fuzzy c-means seemed to be robust to some changes in feature space representation of spikes, as illustrated by the example in Figure 8, these results provide support to our expectation that optimization of performance in terms of the feature matrix MSE instead of $\hat{P}_{TP}$ is a feasible approach in future studies.

Effects of Spike Detection Jitter on Spike Sorting

Figure 10 shows the mean difference in correct classification rate between cases with and without spike detection jitter for 2 and 4 target units at some of the noise levels studied. In general, the influence of spike detection jitter decreased



(c)



(d)

Figure 9: The distributions of sampling rate and resolution breakpoints for spike sorting with principal component analysis and fuzzy c-means (continued from page 138).

with increased noise level and clearly seemed to converge to become insignificant as sampling rate was increased.

Figure 11 shows the distributions of sampling rate breakpoints for correct classification rate decrease caused by the introduction of spike detection jitter. Sampling rate breakpoints between 2.3 kHz and 11.5 kHz were obtained (upper limit of 5% median comparison interval). Noise level did not have significant influence on breakpoint locations per case, but a significant difference was seen between cases with 2 and 4 target units at all noise levels.

It may seem counterintuitive that the loss in spike sorting performance is the smallest at high noise levels. However, as noise level is increased, the correct classification curves are shifted downwards for both cases, i.e. with and without jitter, and the higher the noise level, the less becomes the difference between the two curves. Therefore, at high noise levels, there is less to be gained in spike sorting performance by achieving spike alignment.

This suggest that as long as spikes are detected in and extracted from the signal at a sampling rate of 11.5 kHz or higher, the spike detection jitter introduced will not decrease spike sorting performance by more than 1% in correct classification rate and hence no further measures need to be taken to

establish spike alignment in post-processing prior to the spike sorting step.

The breakpoints reported here are considerably lower than those reported in [21]. A possible explanation for this difference is the robustness towards changes in input data displayed by fuzzy c-means clustering (see Section 3.2). In [21] the authors identified the sampling rate at which the feature space representation converged. As qualitative examination of the example in Figure 8(b) indicates, the feature matrix MSE converges at a considerably higher sampling rate than the correct classification rate.

Spike detection jitter did not cause significant effects on spike sorting performance when sampling resolution was varied. Again, this is due to the fact that sampling resolution was altered while keeping the sampling rate fixed at 25 kHz, which is above sampling rate breakpoints in all cases (Figure 11).

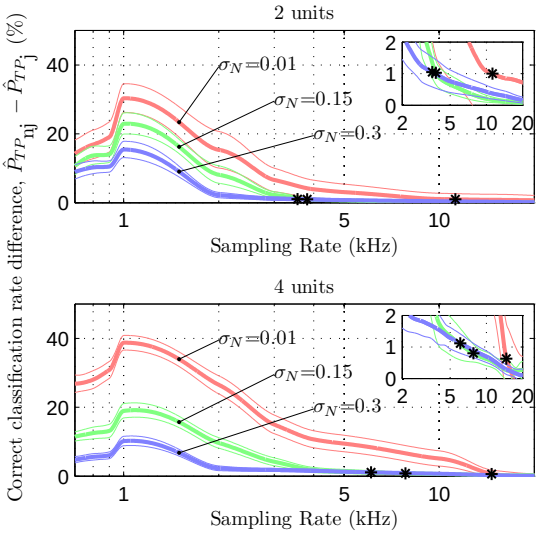


Figure 10: The influence of spike detection jitter on the performance of spike sorting was quantified as the difference in correct classification rate caused by the introduction of jitter, $\hat{P}_{TP_{\mathrm{nj}}} - \hat{P}_{TP_{\mathrm{j}}}$. The upper and lower panel show the mean difference over all recordings at three different noise levels (σ_N) for 2 and 4 target units respectively. The stars indicate the 1% breakpoints obtained for the mean curves. 95% confidence intervals for the mean are indicated with thin lines.

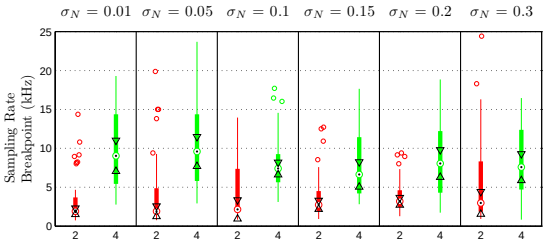


Figure 11: The distributions of sampling rate breakpoints for correct classification rate difference caused by spike detection jitter. The number of target units is indicated with 2 and 4. σ_N is the standard deviation of physiological background noise. Median comparison intervals ($p = 0.05$) are marked with triangles ($\triangle/\triangledown$).

Joint Effects of Sampling Rate and Resolution on Spike Sorting Performance

Figure 12 shows the mean correct classification rate (across all recordings) as a function of sampling rate and resolution for 4 target units at a subset of the noise levels studied. As noise level increased, the maximum achievable correct classification rate was decreased. Figure 13 shows the distributions of joint sampling rate and resolution breakpoints for spike sorting.

Maximum correct classification rate was robust to noise levels up to $\sigma_N = 0.2$ when 2 target units were present but deteriorated significantly already at a noise level of $\sigma_N = 0.05$ when 4 target units were present.

Sampling rate and resolution breakpoints were obtained at 1.0 to 13.5 kHz and 2.2 to 4.6 bits respectively. However, at the higher sampling rate limit, which occurred at a noise level of $\sigma_N = 0.3$, the maximum correct classification rate was just above 50%, meaning that the noise level was the primary factor to concern rather than sampling rate or resolution.

These breakpoints are similar to the breakpoints obtained when individually varying sampling rate and resolution (Section 3.2), although the individual treatment of sampling resolution seems to provide a somewhat lower breakpoint estimation. Therefore, when estimating performance breakpoints, a joint treatment of sampling rate and resolution appears to be a more reliable approach, although separate treatment provides a good approximation.

These results show that spike sorting performance does not suffer a loss in correct classification rate by more than 1% as long as spikes passed to the spike sorting algorithm are aligned, either by employing a sufficiently high sampling rate prior to spike detection (Section 3.2) or by post-processing, and sampled

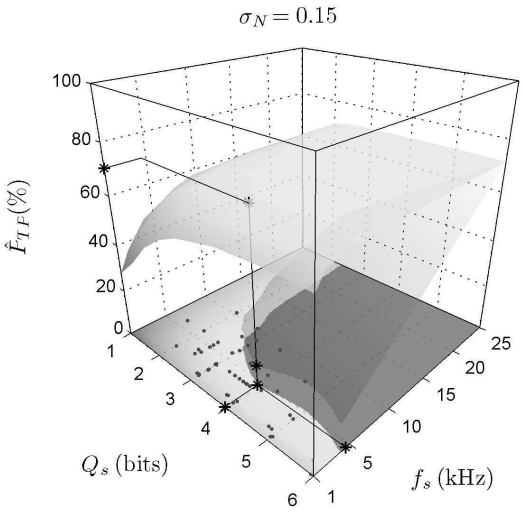
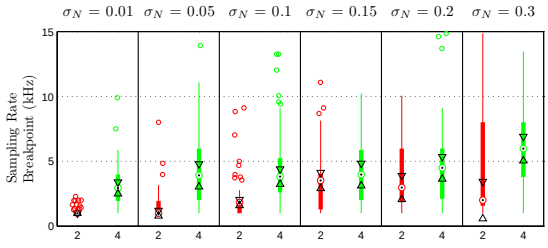
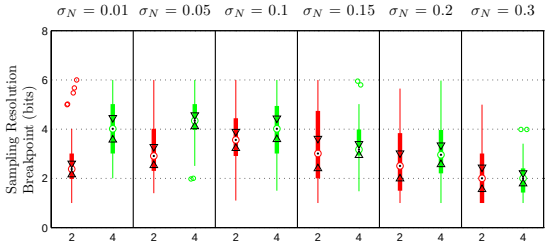


Figure 12: Joint effects of sampling rate, f_s, and resolution, Q_s, on spike sorting performance were quantified by calculating the correct classification rate, $\hat{P}_{TP}$, while jointly varying the two variables on a subset of the values spanned by the dedicated parts (Section 3.2). The figure shows the mean $\hat{P}_{TP}$ surface across all recordings for 4 target units at a noise level of $\sigma_N = 0.15$. A threshold of 1% below $\hat{P}_{TP}$ at full sampling rate and resolution was applied to the surface and the breakpoint was taken as the point in the area of the f_s-Q_s plane fulfilling the threshold criteria (dark area) that minimized bitrate (sampling rate $\times$ sampling resolution, bits/second). The bitrate is shown as the surface below the $\hat{P}_{TP}$ surface, normalized to the maximum $\hat{P}_{TP}$ value for the sake of visual clarity. The breakpoints obtained for the mean surfaces are indicated with stars and are projected onto the f_s-, Q_s- and $\hat{P}_{TP}$-axes and the bitrate surface. Breakpoints obtained for the individual recordings are shown with dots for comparison.

at a sampling rate and resolution of at least 13.5 kHz and 4.6 bits respectively. However, assuming a noise level of $\sigma_N = 0.2$ or lower, and thereby raising the maximum performance further above the 50% limit, the sampling rate breakpoint is lowered to 5.4 kHz.



(a)



(b)

Figure 13: The distributions of joint sampling rate (a) and resolution (b) breakpoints for spike sorting with principal component analysis and fuzzy c-means. The number of target units is indicated with 2 and 4 and σ_N is the standard deviation of physiological background noise. Median comparison intervals ($p = 0.05$) are marked with triangles ($\triangle/\triangledown$).

3.3 Generalization to Other Algorithms

Figure 14 shows the mean performance (across all recordings) of the algorithms considered for spike detection and spike sorting as a function of sampling rate and resolution for representative cases (noise level and number of target units).

Spike Detection

For all of the algorithms studied, the effect of sampling rate on false positive detection rate was insignificant in comparison with true positive detection rate. False positive detection rate did however increase slightly with increased noise level for all algorithms. Thus, we focus our discussion regarding sampling rate towards true positive detection rate. At low noise levels, ABS and NEO performed similarly and had a similar dependency on sampling rate. As noise level increased, NEO outperformed ABS and the curve knee for true positive

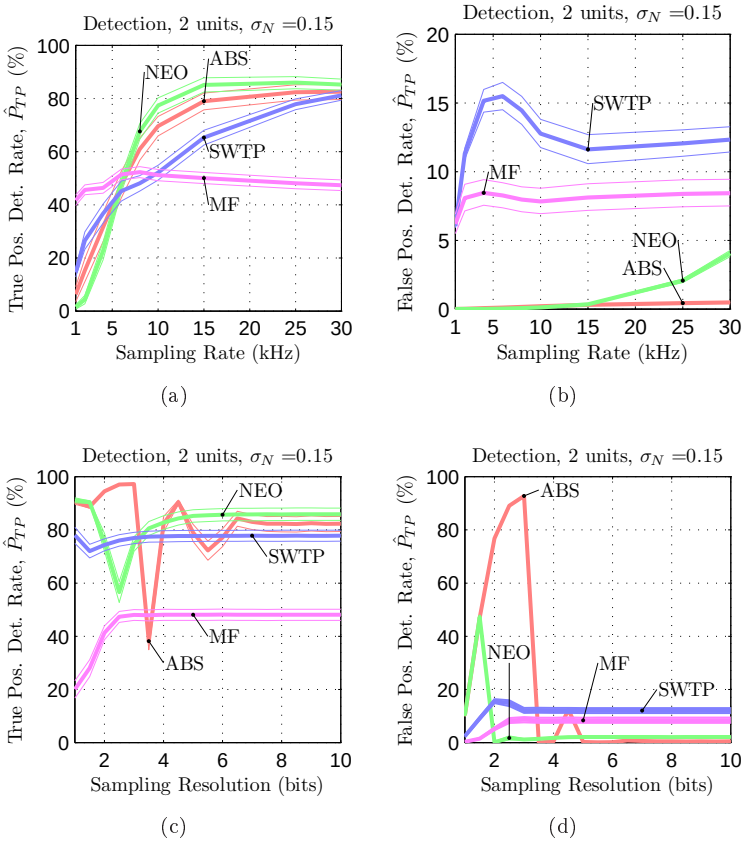
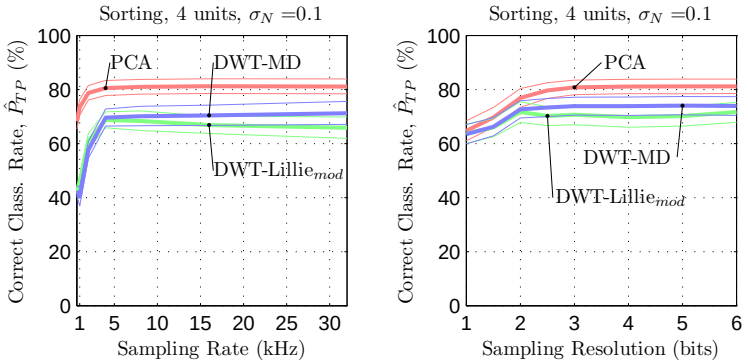


Figure 14: Comparison of algorithms for spike detection and spike sorting. (a) and (b) show mean true and false positive detection rates (across all recordings) respectively as functions of sampling rate. (c) and (d) show mean true and false positive detection rates (across all recordings) respectively as functions of sampling resolution. (a) and (b) show mean correct classification rates (across all recordings) as functions of sampling rate and resolution respectively. The cases shown here (noise levels and numbers of target units) are representative the cases studied. 95% confidence intervals for the mean are indicated with thin lines. (Continues on page 145)

detection rate became less clear. SWTP seemed to be the most robust to noise level, but had the least clear curve knee throughout. The unclear curve knee for



(a) (b)

Figure 14: Comparison of algorithms for spike detection and spike sorting (continued from page 144).

NEO at high noise levels and SWTP at all noise levels was similar to what was observed for ABS at high noise levels (see Section 3.1). The behavior of MF was rather inconsistent with the behavior of the other algorithms. At low noise levels it had the lowest achievable performance but in terms of true positive detection rate, it showed a relatively high robustness to increased noise level compared to ABS. Increasing the number of target units did not influence the relative performances between the algorithms.

Out of all of the algorithms, ABS showed the clearest dependency on sampling resolution (oscillations in performance curves as described in Section 3.1) and it's breakpoint estimates (onset of oscillations) consistently seemed to overestimate the breakpoints for the other algorithms. SWTP seemed to be the most robust algorithm to lowered sampling resolution, which is in line with the observation that it is the most robust to noise level, since lowering sampling resolution can be seen as an introduction of increased noise level.

Spike Sorting

PCA and DWT (with both feature selection methods) showed similar dependency on both sampling rate and resolution. In the mean cases, PCA consistently performed slightly better than DWT, but the curve knee seemed to occur at similar sampling rate and resolution for both algorithms. The difference in performance between PCA and DWT seemed to increase with increased noise level. The performance of DWT was not influenced significantly by the choice

of feature selection method.

4 Discussions and Conclusions

In this work we have explored the influences of varying sampling rate, sampling resolution, level of physiological background noise and number of target units on the performance in spike detection and spike sorting with absolute value threshold and principal component analysis with fuzzy c-means. From the results we have identified sampling rate and resolution breakpoints by applying thresholds of 1% to 5% below the maximum achievable performance level or the performance level at full sampling rate or resolution for each case. We have examined the development of the feature space representation of spikes as a means of quantifying the information provided to the spike sorting algorithm with the simplification of estimating performance breakpoints in mind. We have looked at how the presence of spike detection jitter influences spike sorting performance. Finally, with the intention to explore the generalizability of our results, we have performed a preliminary comparison of the performance of other algorithms for spike detection and spike sorting. Besides shedding light on generalizability, the results from that comparison also provide insight into the performance gain that increased computational resources would facilitate.

We conclude that performance curves for spike detection and spike sorting plateau in most cases and that sampling rate and resolution breakpoints can be estimated in order to find the minimum amount of raw input data that ensures maximum accuracy in assessment of single and multiunit activity. At high noise levels, breakpoints for spike detection are less clear since the plateauing in true positive detection rate is decreased (Figure 6) and maximum achievable performance for both spike detection and spike sorting is severely decreased (Figures 7 and 9). Therefore the minimization of noise should be the primary point of consideration in the design process. This is in agreement with the results reported in [29].

Maximum achievable performance decreased with increased number of target units in all cases (Figures 7 and 9). For spike detection, this is presumably due to an increased probability of overlapping spikes. We note that we considered the case where all target units had a mean firing rate of 20 spikes/second, which we consider to reflect a high overall degree of neuronal activity and thus a worst-case-scenario in this sense. For spike sorting, this is explained by the fact that the difficulty of the clustering problem increases with increased number of clusters and with increased similarity between clusters.

Our results show that the alignment of spike waveforms is an important factor in lowering the sampling rate breakpoints for spike sorting. The align-

ment can be achieved either by employing a sufficiently high sampling rate in the initial digitization step or by post-processing. The significant correlation between the PCA feature matrix MSE and correct classification rate in spike sorting indicates that optimization of performance in terms of feature matrix MSE instead of spike sorting accuracy is a feasible approach in future studies.

Coming back to the crucial steps in BMI design presented in Section 1, assuming that the assessment of single or multiunit activity is the purpose of the BMI (step 1) and that an adequate noise minimization ($\sigma_N \leq 0.15$) has been achieved, our results provide general guidelines for the choice of sampling rate and resolution (step 2) to be employed in the analog-to-digital converter of the acquisition unit.

For absolute threshold spike detection and the assessment of multi unit activity, a sampling rate of 16 to 31 kHz (5% and 1% performance loss tolerance respectively) and an effective sampling resolution of 9 bits (1% performance loss tolerance) should be employed (Section 3.1, Figure 7). For spike sorting with principal component analysis and fuzzy c-means, assuming that the spike detection guidelines are followed, the extracted spike waveforms do not need realignment and can be lowpassfiltered and downsampled to a sampling rate of 5 kHz and an effective sampling resolution of 5 bits (1 % performance loss, Section 3.2, Figure 13). For a recording containing 2 target units with a firing rate of 20 spikes/second each and assuming a spike duration of 3 ms, transmitting the extracted spike waveforms over a wireless link would imply an approximately eightfold reduction in bitrate as compared to transmitting the raw signal. Transmitting the waveforms after downsampling them would imply a total of approximately a fifty- to hundredfold bitrate reduction, depending on the initial sampling rate employed (16 to 31 kHz). We stress the precautions regarding sampling resolution and the dynamic range of the ADC mentioned in Section 2.1. The guidelines for spike sorting are applicable for noise levels up to $\sigma_N = 0.2$, although spike detection performance is significantly affected at such a high noise level.

Having observed that the performance of the algorithms considered in most cases have similar dependencies on sampling rate and resolution, we conclude that the breakpoints obtained for spike detection with absolute value thresholding and spike sorting with principal component analysis can be used to provide an indication of breakpoint locations for the other algorithms studied (Figure 14). Sampling rate breakpoints for NEO are expected to be similar to those for ABS, especially at low noise levels (at least up to $\sigma_N = 0.15$). At higher noise levels ($\sigma_N = 0.3$), the performance of NEO converges slower with increased sampling rate, but it's maximum performance is significantly higher than that of ABS. In general, SWTP converges slower than ABS, indicating that it requires a higher sampling rate to reach it's maximum performance. In

the high noise case however, it's maximum performance is higher than that of both ABS and NEO. MF's dependency on sampling rate was inconsistent with that of the other algorithms. This is possibly explained by the suboptimality in it's implementation, i.e. using the mean of all stored waveforms as a filter waveform. Out of all of the spike detection algorithms, ABS was influenced the most by reduced sampling resolution. However, it consistently seemed to overestimate the sampling resolution breakpoints of the other algorithms. From this we conclude that NEO and SWTP are feasible alternatives to ABS at high noise levels, but at the cost of increased sampling rate and computational complexity. We also conclude that the amount of raw input data (sampling rate and resolution) should be taken into account when comparing the performance of spike sorting algorithms. Both sampling rate and resolution breakpoints for spike sorting with DWT seemed to be similar to those for PCA. In contrast with the conclusions of [26], PCA performed better than DWT in the mean case. However, according to [33] PCA should be expected to perform better when the differences between target unit waveforms are mainly large-scale, which is likely to be the case in our synthesized recordings.

We emphasize that all of the algorithms considered involve some free parameters such as scaling factors for setting thresholds, length of smoothing windows, feature selection methods etc., and optimizing those for each specific case might boost their performance. In the fully unsupervised case however, such optimization might not be possible. We have thus kept all algorithm-specific parameters fixed throughout our simulations and in that sense we have tested a worst-case-scenario.

Recordings with non-stationary properties can be modeled as series of segments with stationary properties. Thus, the dimensioning of a BMI could be based on either the most challenging stationary segment in the recording or the "mean segment".

Given a set of resources in terms of computational and wireless link capacity, our guidelines give rise to a processing task allocation scheme (step 3), presented in Figure 15. By following the guidelines presented here and allocating processing tasks based on the resources available, the bitrate of data into the wireless link can be minimized without jeopardizing the reliability of the output information, namely that regarding single or multi unit activity. Thereby, a balanced BMI design is achieved – a design that neither suffers from over- nor underdimensioning. The guidelines presented here are based on spike detection with ABS and spike sorting with PCA.

Some sources of epistemic uncertainty related to this work should be mentioned. First, the simulation parameters were chosen to represent challenging scenarios in terms of high target unit activity, without taking into account the possibility of a low degree of activity. However, breakpoint estimations

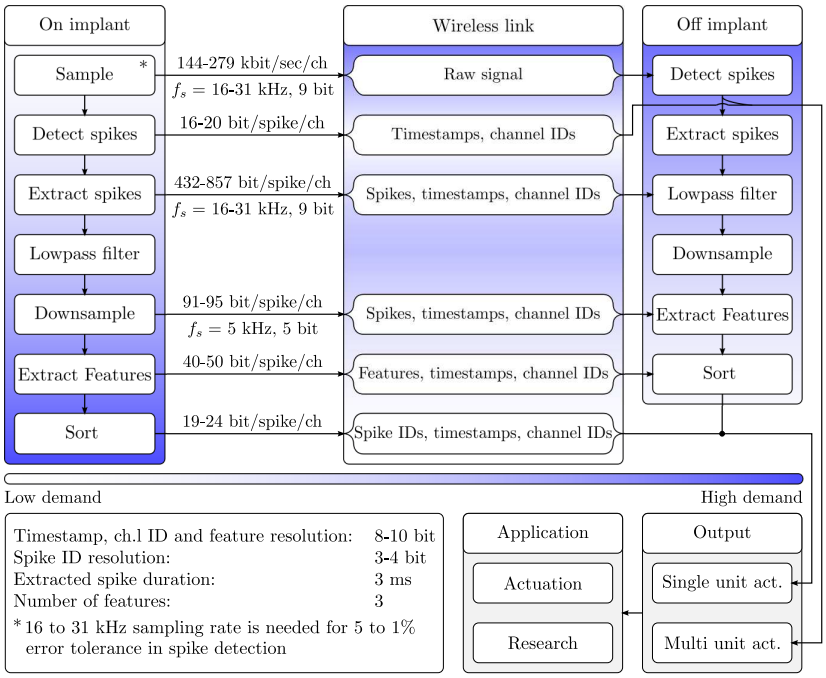


Figure 15: We conclude by presenting a task allocation scheme for wireless BMI designs, based on available resources (computational and wireless link capacity) and the purpose of the BMI in terms of processing unit output. The arrows indicate alternative processing task allocations. For each suggested task allocation, the bitrate and the type of data fed into the wireless link are provided for a typical case. The bitrate estimations are based on the sampling rate and resolution breakpoints obtained from our results and ensure minimum loss of performance in spike detection and spike sorting, and thereby sustained reliability in single and multiunit activity assessment, provided that minimization of noise has been performed ($\sigma_N \leq 0.15$ to 0.2).

based on such worst case scenarios should provide upper limits that can be adjusted (lowered) to match more specific situations. The workflow presented here can be used to estimate breakpoints for any given situation that deviates from the ones studied here in terms of e.g. number of target units, firing characteristics of contributing neurons and stationarity/non-stationarity of the recordings. Second, full breakpoint analysis was not performed for all of the algorithms considered. Instead, we showed that we can draw qualitative con-

clusions regarding their breakpoints, based on the breakpoints for ABS and PCA. At last, we assume single channel recordings to be performed. Again, this reflects a worst case scenario in terms of spike detection and spike sorting performance since multi channel recordings allow for improving performance by utilizing the correlations between near by channels. The introduction of additional recording channels would presumably lower the requirements for raw input data to each individual channel, provided that that there is some degree of correlation between the activity measured by the individual channels.

Acknowledgements

This work was supported by a Linnaeus Grant from the Swedish Research Council (no. 60012701), a grant from the Knut and Alice Wallenberg Foundation (no. 2004.0119) and the Medical Faculty at Lund University.

References

[1] J. P. Donoghue, "Connecting cortex to machines: recent advances in brain interfaces.," *Nature neuroscience*, vol. 5 Suppl, pp. 1085–8, Dec. 2002.

[2] A. B. Schwartz, "Cortical neural prosthetics.," *Annual review of neuroscience*, vol. 27, pp. 487–507, Jan. 2004.

[3] M. a. Lebedev and M. a. L. Nicolelis, "Brain-machine interfaces: past, present and future.," *Trends in neurosciences*, vol. 29, pp. 536–46, Sept. 2006.

[4] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. a. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn, and J. P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia.," *Nature*, vol. 442, pp. 164–71, July 2006.

[5] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding.," *Nature*, vol. 453, pp. 1098–101, June 2008.

[6] J. Csicsvari, "Massively Parallel Recording of Unit and Local Field Potentials With Silicon-Based Electrodes," *Journal of Neurophysiology*, vol. 90, pp. 1314–1323, Apr. 2003.

[7] G. Buzsáki, "Large-scale recording of neuronal ensembles.," *Nature neuroscience*, vol. 7, pp. 446–51, May 2004.

[8] P. Mitra and H. Bokil, *Observed Brain Dynamics*. Oxford University Press, USA, 2008.

[9] G. W. Fraser, S. M. Chase, A. Whitford, and A. B. Schwartz, "Control of a brain-computer interface without spike sorting.," *Journal of neural engineering*, vol. 6, p. 055004, Oct. 2009.

[10] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials.," *Network (Bristol, England)*, vol. 9, pp. R53–78, Nov. 1998.

[11] J. Morizio, P. Irazoqui, V. Go, and J. Parmentier, "Wireless Headstage for Neural Prosthetics," *Conference Proceedings 2nd International IEEE EMBS Conference on Neural Engineering 2005*, vol. 2005, pp. 414–417, 2005.

[12] D. A. Borton, Y.-K. Song, W. R. Patterson, C. W. Bull, S. Park, F. Laiwalla, J. P. Donoghue, and A. V. Nurmikko, "Wireless, high-bandwidth recordings from non-human primate motor cortex using a scalable 16-Ch implantable microsystem.," *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, vol. 2009, pp. 5531–5534, 2009.

[13] T. A. Szuts, V. Fadeyev, S. Kachiguine, A. Sher, M. V. Grivich, M. Agrochão, P. Hottowy, W. Dabrowski, E. V. Lubenov, A. G. Siapas, N. Uchida, A. M. Litke, and M. Meister, "A wireless multi-channel neural amplifier for freely moving animals.," *Nature Neuroscience*, vol. 14, no. 2, pp. 263–269, 2011.

[14] "Multi Channel Systems - http://www.multichannelsystems.com."

[15] R. R. Harrison, P. T. Watkins, R. J. Kier, R. O. Lovejoy, D. J. Black, B. Greger, and F. Solzbacher, "A Low-Power Integrated Circuit for a Wireless 100-Electrode Neural Recording System," *IEEE Journal of Solid-State Circuits*, vol. 42, pp. 123–133, Jan. 2007.

[16] "Alpha Omega - http://www.alphaomega-eng.com."

[17] M. Rizk, I. Obeid, S. H. Callender, and P. D. Wolf, "A single-chip signal processing and telemetry engine for an implantable 96-channel neural data acquisition system.," *Journal of neural engineering*, vol. 4, pp. 309–21, Sept. 2007.

[18] F. Zhang and M. Aghagolzadeh, "An implantable neuroprocessor for multichannel compressive neural recording and on-the-fly spike sorting with wireless telemetry," *Biomedical Circuits and*, pp. 3–6, 2010.

[19] M. Abeles and M. Goldstein, "Multispike train analysis," *Proceedings of the IEEE*, vol. 65, no. 5, pp. 762–773, 1977.

[20] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-Gaussian variability.," *Journal of neuroscience methods*, vol. 69, pp. 175–88, Nov. 1996.

[21] Y. Ghanbari, P. Papamichalis, and L. Spence, "Robustness of neural spike sorting to sampling rate and quantization bit depth," in *2009 16th International Conference on Digital Signal Processing*, pp. 1–6, IEEE, July 2009.

[22] Z. Nadasdy, *Spatio temporal patterns in the extracellular recording of hippocampal pyramidal cells: From single spikes to spike sequences.* PhD thesis, Rutgers, The State University of New Jersey, 1998.

[23] P. T. Thorbergsson, H. Jorntell, F. Bengtsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Spike library based simulator for extracellular single unit neuronal signals.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2009, pp. 6998–7001, Jan. 2009.

[24] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Statistical modelling of spike libraries for simulation of extracellular recordings in the cerebellum.," *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, vol. 2010, pp. 4250–4253, 2010.

[25] R. Pettai, *Noise In Receiving Systems.* New York, NY: Wiley, 1984.

[26] R. Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering.," *Neural computation*, vol. 16, pp. 1661–87, Aug. 2004.

[27] S. Mukhopadhyay and G. C. Ray, "A new interpretation of nonlinear energy operator and its efficacy in spike detection.," *IEEE transactions on bio-medical engineering*, vol. 45, pp. 180–7, Feb. 1998.

[28] K. H. Kim and S. J. Kim, "A wavelet-based method for action potential detection from extracellular neural signal recording with low signal-to-noise ratio.," *IEEE transactions on bio-medical engineering*, vol. 50, pp. 999–1011, Aug. 2003.

[29] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application.," *IEEE transactions on bio-medical engineering*, vol. 51, pp. 905–11, June 2004.

[30] S. Gibson, J. W. Judy, and D. Marković, "Technology-aware algorithm design for neural spike detection, feature extraction, and dimensionality reduction.," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 18, pp. 469–78, Oct. 2010.

[31] Z. Nadasdy, R. Quian Quiroga, Y. Ben-Shaul, B. Pesaran, D. A. Wagenaar, and R. A. Andersen, "Comparison of Unsupervised Algorithms for On-line and Off-line Spike Sorting," *32nd Annu. Meeting Soc. for Neurosci*, no. 1996, pp. 3254–3254, 2002.

[32] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. New York, NY: Wiley, 2nd ed., 2001.

[33] A. Pavlov, V. a. Makarov, I. Makarova, and F. Panetsos, "Sorting of neural spikes: When wavelet based methods outperform principal component analysis," *Natural Computing*, vol. 6, pp. 269–281, Aug. 2007.

Paper V

Spike-Feature Based Estimation of Electrode Position in Extracellular Neural Recordings

Abstract

Detecting and sorting spikes in extracellular neural recordings are common procedures in assessing the activity of individual neurons. In chronic recordings, passive electrode movements introduce changes in the shape of detected spike waveforms, and may thus lead to problems with identification and tracking of spikes recorded at separate instances in time, which is an important step in long-term monitoring of individual neurons. Information about electrode movements after implantation is crucial to the evaluation of mechanical stability of different electrode designs. In this paper, we present a preliminary study of the relationship between electrode movements and the resulting movements of spike-features in feature space. We show that there is a characteristic relationship between the two movements and that this relationship can be modeled as a linear transformation between two coordinate systems. Finally, we show how the relationship can be used for estimating electrode positions based on measured spike waveforms without any prior knowledge about the type of neuron by introducing a learning procedure during electrode insertion.

Based on: P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Spike-Feature Based Estimation of Electrode Position in Extracellular Neural Recordings", *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, pp. 3380 – 3383, 2012.

1 Introduction

Extracellular recordings with chronically implanted microelectrodes are a common means of acquiring signals reflecting the activity of individual neurons in the central nervous system [1]. The recorded signal then consists of the spiking activity of near-by neurons (target neurons), the combined spiking activity of a large number of far-away neurons (noise neurons), thermal noise generated in the front-end electronics and local field potentials [2], [3].

When the target neurons are sufficiently close to the recording electrode, their spikes can be detected [4] and sorted [5] in order to reveal the firing patterns of individual neurons. Assuming that the detection has been successful, the sorting step involves extracting features from the spike waveforms and classifying similar waveforms as originating from the same neuron. Feature extraction is commonly carried out by projecting the spike waveforms onto a set of basis waveforms that can be obtained through e.g. principal component analysis (PCA) of the acquired spike waveforms.

Differences in spike waveforms arise from differences in neuron morphology and differences in electrode position relative to the different neurons [6], [7]. While spike sorting relies on these differences, they can become problematic in dynamic situations, i.e. where the recording electrode can move in relation to the target neuron(s). This becomes especially challenging when comparing identified units in recordings from a specific electrode that are executed at separate time instances. While small electrode movements can slightly change the feature space representation of detected spikes from a given neuron, larger movements can put that neuron out of range from the electrode and new neurons into range. Solving this problem is commonly referred to as spike-tracking [8], [9] and involves comparing units between separate recording instances and concluding that they either originate in the same neuron or in different neurons.

Gaining insight into how electrode movements are translated into spike movements in feature space would be of great benefit both in terms of solving the spike-tracking problem and in terms of being able to estimate electrode movements based on observed spike waveforms. In this paper, we present a preliminary study of the relationship between physical movements of the recording electrode and the corresponding movements of spike-features in feature space. Using mathematical models to simulate multi-electrode recordings, we demonstrate how electrode movements along a given path are translated into spike-feature movements along a similar path in feature space. Exploring this insight, we present a method for using recorded spike waveforms to estimate the electrode position, based on the relationship between the two domains. Our results show that there is a characteristic relationship between electrode movements in the physical domain and spike-feature movements in the feature domain.

This relationship is evident even when employing sub-optimal feature spaces. Our results also show that recorded spike waveforms provide information about the recording electrode position if a training procedure is carried out during implantation of the electrode.

2 Methods

2.1 Dataset

As test data, we used synthetic multi-electrode recordings where the electrode sites were placed along the paths of electrode movements we wished to test in each case. We used a recently developed simulation tool that employs dimensionality reduction techniques to compactly describe the spatial dependency of the measured spike waveform. This allows for an efficient simulation of multielectrode recordings with realistic properties [10]. The model was derived by compressing the information obtained when calculating spike waveforms in measurement points surrounding a compartment model of a CA1 pyramidal neuron using the simulation environment NEURON [6], [11], [12].

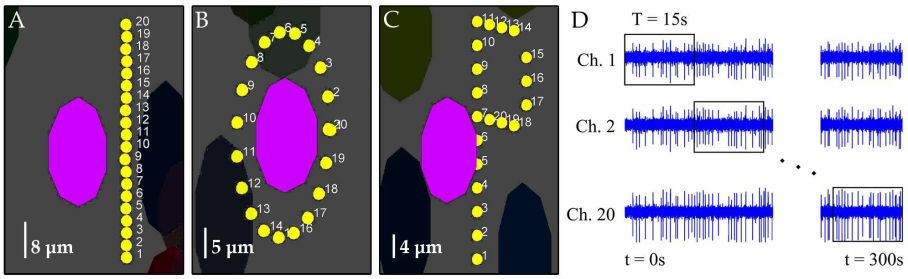


Figure 1: (a)-(c) Three different recording setups were simulated, one for each of the electrode array configurations shown in (a) - (c). One target neuron (purple ellipsoid) was placed close to the electrode array and approximately 500 interfering neurons (dark ellipsoids) were placed at random positions at least 150 μm away from the target neuron. (d) The effects of electrode movements were mimicked by extracting spikes from the recording channels from a time frame of length T s ($T = 15$ s in the figure) while successively sweeping the window across the channels.

Fig. 1 shows the target neuron and the electrode locations considered for each of the three test recordings. The electrode paths we considered were 1) linear movement, 2) movement along an ellipse and 3) movement along a P-

shaped path. Fig. 1 (d) shows how electrode movement was mimicked by extracting spikes in a time frame of duration T that was swept across the channels. This corresponds to the assumption that the electrode stays in place for the duration of the window, T. The linear movement was assumed to represent the practical case where electrodes are inserted and then assumed to move along one axis. The elliptic and P-shaped paths were included to show that complex electrode movements were translated to similar movements of spikes in the feature space.

Twenty electrode sites were simulated in each case and the duration of each recording was 5 minutes. One neuron was placed in the origin and approximately 600 noise neurons were placed at random positions, but at a minimum distance of 150 m from the target neuron. Each noise neuron was given a random mean firing rate between 1 and 50 spikes/second. A mean firing rate of 20 spikes/second was assigned to the target neuron. Spike times were generated by assuming gamma-distributed inter-spike intervals [13]. All spike times were stored at the time of simulation and used to extract spike waveforms.

2.2 Feature-Space Representation of Spike Waveforms

The P-dimensional feature-space representation of sampled spike waveforms of length N (N = number of samples) was obtained by projecting the spike waveforms onto a set of P (P = number of feature space dimensions) N sample long basis waveforms. To explore the effects of the selection of basis waveforms, we considered three sets of basis waveforms, where the j-th set of basis waveforms was contained in the columns of the $N \times P$ matrix $\mathbf{B}_j$. The columns of the matrix $\mathbf{B}_j$ were obtained as the first P basis waveforms from the principal component analysis of each of the following matrices:

1. The matrix containing the mean spike waveforms in each electrode position in its columns (optimal basis).
2. A matrix of the same size as that in 1), but whose elements were normally distributed random numbers (sub-optimal basis).
3. The matrix containing the three very first waveforms in the first electrode position.

The $P \times M$ feature space representation $\mathbf{W}_j$ of the spike waveforms in the $N \times M$ matrix $\mathbf{S}$ in the space spanned by the basis waveforms in $\mathbf{B}_j$ was thus obtained through the projection and normalization

$$\mathbf{W}_j = \frac{\mathbf{B}_j^T \mathbf{S}}{||\mathbf{B}_j^T \mathbf{S}||_2} \tag{1}$$

where $||.||_2$ denotes the Euclidean norm.

2.3 Comparison of Movement Paths between Spaces

To explore the similarity between electrode paths in the physical space and detected spikes in the feature space we defined the path measure d that summarizes the path in a given Q-dimensional space ($Q > 1$) in a 1-dimensional sequence of normalized Euclidean distances of points along the path to the mean point of the path. For a path whose i-th coordinate in the original Q-dimensional space is given by p_i and whose mean coordinate is given by p_0, the distance measure in the i-th point is given by

$$d_i = ||p_0 - p_i||_2. \tag{2}$$

The i-th point corresponds to electrode site i out of 20. For the feature spaces, we only considered the first three dimensions in this comparison. All distance measure sequences were normalized by first subtracting their respective mean values and then dividing by the Euclidean norm of the resulting sequence.

Although interpreting these one-dimensional path measures in terms of actual paths in a three-dimensional space is not straight forward, especially for more complex paths, they do provide a means for assessing the geometrical similarity between two paths in separate domains.

2.4 Spike-Feature Based Estimation of Electrode Position

Based on the observation that electrode movements along a given path are translated to spike waveform movements along a similar path in the feature space, we assumed that there existed a $P \times 3$ matrix $\mathbf{A}$ that transformed the 3-dimensional vector of Cartesian electrode coordinates to corresponding P-dimensional spike waveform coordinates in the feature space. This transformation is described by the linear model

$$\mathbf{B}^T\mathbf{S} = \mathbf{A}\mathbf{X} + \eta \tag{3}$$

where $\mathbf{B}$ and $\mathbf{S}$ are the basis- and spike waveform matrices respectively, $\mathbf{X}$ is the matrix containing the Cartesian coordinates of the electrode positions and η is a matrix containing noise or variations not captured by the transformation matrix.

For a known matrix of Cartesian measurement point coordinates $\mathbf{X}_0$, a corresponding matrix of measured mean spike waveforms $\mathbf{S}_0$ and a set of basis waveforms $\mathbf{B}$, an estimator for the transformation matrix is obtained by multiplying both sides of Eq. 3 by the Moore-Penrose psesudoinverse [14] of the

Cartesian coordinate matrix, $\mathbf{X}^+$, so that $\mathbf{X}\mathbf{X}^+ = \mathbf{I}$, from the right, or

$$\hat{\mathbf{A}} = \mathbf{B}^T\mathbf{S}_0\mathbf{X}_0^+ \tag{4}$$

Having obtained the estimate for the transformation matrix, it can now be used to estimate the electrode position for a given set of measured mean spike waveforms $\mathbf{S}$ by multiplying Eq. 3 with the Moore-Penrose pseudoinverse of the estimated transformation matrix from the left, or

$$\hat{\mathbf{X}} = \hat{\mathbf{A}}^+\mathbf{B}^T\mathbf{S}. \tag{5}$$

To perform and evaluate feature space based electrode positioning, we divided the test data for the linear electrode movement (Sec. II-A) into two parts - odd-numbered channels (training data) and even-numbered channels (test data). The mean spike waveforms and Cartesian electrode coordinates of the odd-numbered channels, $\mathbf{S}_0$ and $\mathbf{X}_0$ respectively, were used for estimating the transformation matrix according to Eq. 4. This corresponds to a training period during which the position of the electrode is known. The resulting transformation matrix was then used to estimate the electrode positions of the even number channels $\mathbf{X}$ by projecting the corresponding mean waveforms $\mathbf{S}$ onto the basis that was used to estimate the transformation matrix according to Eq. 5.

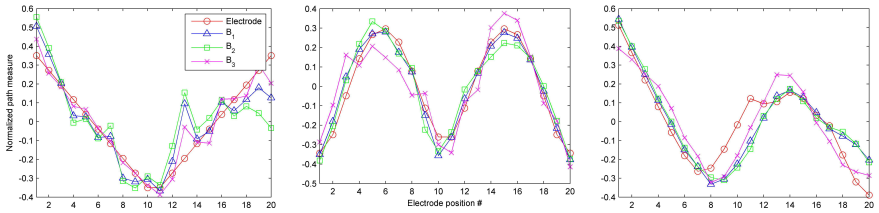
To examine the sensitivity of positioning to the choice of basis waveforms and the number of basis waveforms (number of feature space dimensions), we performed the above procedure using the optimal basis and the random basis (Sec. II-B) while successively increasing the number of dimensions and calculating the estimation error. The estimation error for each case was taken as the mean distance between true and estimated positions across all sites for that case.

3 Results

3.1 Comparison of Movement Paths

Fig. 2 shows the path measure d that characterizes the electrode movements and the movements of the spike-features of mean waveforms in the feature spaces spanned by the three different bases described in Sec. II-B. In all feature spaces, the path measure was similar to that of the electrode path, indicating that electrode movements gave rise to similar spike movements in feature space. This was evident even for the random basis.

Increasing the length of the time spent in each position (T in Fig. 1 (d)), and thus increasing the number of spike waveforms used for forming the mean



(a) (b) (c)

Figure 2: The path measure d for the electrode movements and corresponding movements of mean spike waveforms in the feature spaces spanned by three different sets of basis waveforms. (a) Linear path, (b) elliptical path, (c) P-shaped path. The path measures in the feature spaces were in all cases similar to those in the electrode movement space.

waveform in each position, increased the similarity between the path measures. As expected, the random basis required the largest number of waveforms to reach the high similarity shown in Fig. 2. The figure shows the case where the maximum amount of time is spent in each channel ($T = 15$ seconds).

3.2 Spike-Feature Based Estimation of Electrode Position

Fig. 3 shows the mean estimation error (in m) as a function of the number of feature space dimensions used in the electrode position estimation for the optimal basis and one realization of a random basis. The inset illustrates the true and estimated positions obtained using the 40-dimensional optimal basis.

The estimation error converged at approximately 8 dimensions when using the optimal basis. For the random basis, this limit was dependent on the realization, sometimes being lower than 8 and sometimes higher. In the case show in Fig. 3 the error converged at approximately 22 dimensions to approximately the same error as that obtained by using the optimal basis. For the case studied, the error converged onto a value of approximately 1ţm.

4 Conclusion

In this paper we have presented preliminary findings regarding the relationship between physical movements of a recording electrode in extracellular neural recordings and the corresponding movement of spike-features in a feature space spanned by various basis waveforms. We have shown that there appears to be a characteristic relationship between the movements in these two domains and

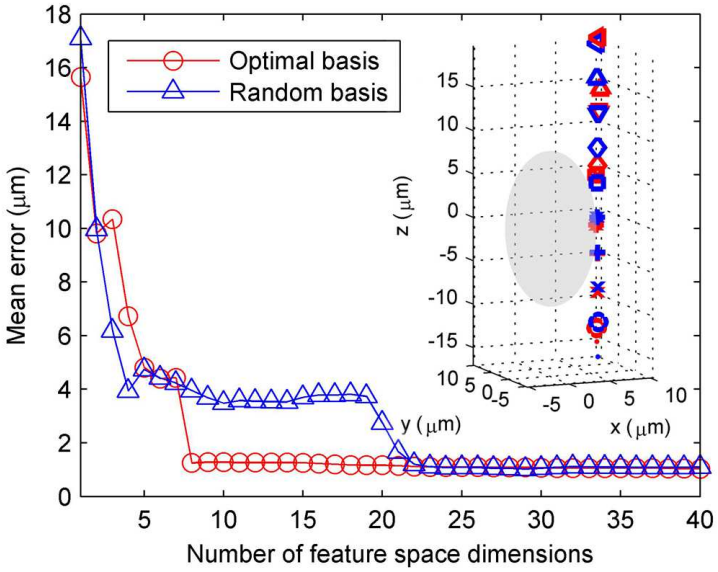


Figure 3: Feature-space based estimation of electrode position during linear movement. The mean positioning error generally converged to a minimum value for both the optimal basis (obtained through PCA of the entire set of mean waveforms in training positions) and the random basis (obtained through PCA of a random matrix). However, the random basis generally converged at a higher number of dimesion. The inset shows the true (red) and estimated (blue) positions when using the 40-dimensional optimal basis. For the sake of clarity, the position of each channel is indicated by a specific symbol.

that this relationship can be interpreted as a linear transformation between two coordinate systems, not necessarily of equal dimensions. We have shown that, for linear electrode movements, this transformation can be acquired during a training period and then applied to estimate electrode position based on the feature space representation of spike waveforms. Due to the introduction of a training procedure during electrode insertion, no prior information about the type of neuron is required. In our study, placing the electrode in the odd-numbered locations corresponds to the training period.

Future work involves verification of our findings by in vivo experiments, the full implementation of a practical framework for acquiring the coordinate transformation during electrode implantation and the utilization of the acquired transformation for post-implantation assessment of electrode movements. Fu-

ture work also involves the utilization of this type of modeling for the purpose of tracking identified units between recording sessions.

Although the present study only addresses electrode positioning along a linear path, our results indicate that the estimation procedure is directly applicable during three-dimensional electrode movements. However, this has not been confirmed and thus requires further investigation.

5 Acknowledgements

References

[1] G. Buzsáki, "Large-scale recording of neuronal ensembles.," *Nature neuroscience*, vol. 7, pp. 446–51, May 2004.

[2] P. Mitra and H. Bokil, *Observed Brain Dynamics*. Oxford University Press, USA, 2008.

[3] S. F. Lempka, M. D. Johnson, M. a. Moffitt, K. J. Otto, D. R. Kipke, and C. C. McIntyre, "Theoretical analysis of intracortical microelectrode recordings.," *Journal of neural engineering*, vol. 8, p. 045006, Aug. 2011.

[4] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application.," *IEEE transactions on bio-medical engineering*, vol. 51, pp. 905–11, June 2004.

[5] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials.," *Network (Bristol, England)*, vol. 9, pp. R53–78, Nov. 1998.

[6] C. Gold, D. a. Henze, and C. Koch, "Using extracellular action potential recordings to constrain compartmental models.," *Journal of computational neuroscience*, vol. 23, pp. 39–58, Aug. 2007.

[7] K. H. Pettersen and G. T. Einevoll, "Amplitude variability and extracellular low-pass filtering of neuronal spikes.," *Biophysical journal*, vol. 94, pp. 784–802, Feb. 2008.

[8] A. Bar-Hillel, A. Spiro, and E. Stark, "Spike sorting: Bayesian clustering of non-stationary data.," *Journal of neuroscience methods*, vol. 157, pp. 303–16, Oct. 2006.

[9] F. Franke, M. Natora, C. Boucsein, M. H. J. Munk, and K. Obermayer, "An online spike detection and spike classification algorithm capable of

instantaneous resolution of overlapping spikes.," *Journal of computational neuroscience*, vol. 29, pp. 127–48, Aug. 2010.

[10] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Computationally efficient simulation of extracellular recordings with multielectrode arrays," *Journal of Neuroscience Methods*, vol. null, Aug. 2012.

[11] M. L. Hines and N. T. Carnevale, "The NEURON Simulation Environment," *Neural Computation*, vol. 9, pp. 1179–1209, Aug. 1997.

[12] C. Gold, D. a. Henze, C. Koch, and G. Buzsáki, "On the origin of the extracellular action potential waveform: A modeling study.," *Journal of neurophysiology*, vol. 95, pp. 3113–28, May 2006.

[13] D. Heeger, "Poisson model of spike generation," *Handout, University of Standford*, pp. 1–13, 2000.

[14] C. Meyer, "Matrix Analysis and Applied Linear Algebra. 2000," *SIAM, Philadelphia*, vol. 1, no. 11, pp. 1829–1841, 2004.

Paper VI

Computationally efficient simulation of extracellular recordings with multielectrode arrays

Abstract

In this paper we present a novel, computationally and memory efficient way of modeling the spatial dependency of measured spike waveforms in extracellular recordings of neuronal activity. We use compartment models to simulate action potentials in neurons and then apply linear source approximation to calculate the resulting extracellular spike waveform on a three dimensional grid of measurement points surrounding the neurons. We then apply traditional compression techniques and polynomial fitting to obtain a compact mathematical description of the spatial dependency of the spike waveform. We show how the compressed models can be used to efficiently calculate the spike waveform from a neuron in a large set of measurement points simultaneously and how the same procedure can be inversed to calculate the spike waveforms from a large set of neurons at a single electrode position. The compressed models have been implemented into an object oriented simulation tool that allows the simulation of multielectrode recordings that capture the variations in spike waveforms that are expected to arise between the different recording channels. The computational simplicity of our approach allows the simulation of a multi-channel recording of signals from large populations of neurons while simulating the activity of every neuron with a high level of detail. We have validated our compressed models against the original data obtained from the compartment models and we have shown, by example, how the simulation approach presented here can be used to quantify the performance in spike sorting as a function of electrode position.

Based on: P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Computationally efficient simulation of extracellular recordings with multielectrode arrays", *Journal of neuroscience methods*, vol. 211, pp. 133 – 144, 2012.

1 Introduction

Recently, there has been great interest in the development of brain machine interfaces (BMIs) with the aim to control prosthetic devices, conduct basic research on the central nervous system (CNS) and to treat the symptoms of neurological disease. One way of performing signal acquisition in BMIs is to use chronically implanted microelectrode arrays [1] to measure the variation in extracellular potential resulting from discharges of action potentials in near by neurons. The extracellular representation of the action potential is usually referred to as a spike. Detecting spikes [2] in the extracellular signal and assigning them to their neurons of origin thus provides information about the activity patterns of individual neurons. The assignment part of that procedure is usually referred to as spike sorting [3]. Since the performance in these processing steps is what determines the quality of the extracted information, the algorithms used for spike detection and spike sorting play a crucial role for the function of BMIs. Apart from the purely functional aspect, they are also important in the context of compressing the information contained in the neural signal for e.g. wireless transmission and/or memory-efficient storage for off-line analysis.

The development of algorithms for information extraction is an important aspect of BMI development. During design and evaluation of such algorithms, test signals are needed with a priori known information content, in which the spike times of each individual neuron in the recording are known and can be compared with the output of the algorithms. In addition to having a priori known characteristics, the test signals need to have realistic signal properties and these properties need to be controllable to some extent. Realism is important for the future applicability of the results and controllability is important since it allows the algorithm designer to perform studies of algorithm performance in a wide range of scenarios that might be encountered in future applications.

The approaches to obtaining adequate test signals can be roughly divided into three categories, (1) simultaneous intra- and extracellular recordings, (2) purely synthetic recordings and (3) hybrid recordings. In simultaneous intra- and extracellular recordings, the intracellular membrane potentials of the cells of interest are measured directly and since the signal-to-noise ratio in these is normally high, they can be used as ground truth when assessing the performance in spike detection and sorting applied to the extracellular signal [4, 5]. This class of test signals provides a high level of realism – the signals in question being real. However, they lack in some practical aspects since keeping track of all true neuronal activity is difficult or even impossible in many cases. Besides these practical problems, controllability of the recording properties is

limited. Despite these downsides, simultaneous intra- and extracellular recordings could serve as ultimate benchmark signals in later steps of the algorithm design process.

Purely synthetic recordings are based on mathematical models of the signal generation process. The mathematical models can in turn be divided into two subcategories, (1) models based on compartment models of the neurons and (2) models based on fixed spike templates. Compartment models rely on more or less detailed models of the mechanisms involved in producing the action potential across the cell membrane and of the resulting signal measured outside the cell [6, 7, 8]. The extracellular signal is calculated by considering the voltage contribution of each point on each contributing neuron at each given time instance. The amount of details captured by such models thus leads to high computational demand, which makes them unpractical when modeling large populations of neurons. However, they are realistic in the sense that they do capture the variations in the spike waveform's shape that arise when placing the recording electrode in different measurement points [7]. This feature is of great importance when modeling recording setups with multiple and/or positionally unstable recording electrodes, both of which are important factors to consider during development of algorithms for spike detection and spike sorting in realistic scenarios.

Models based on fixed spike templates assume that the extracellular spike waveform measured from a given cell can be selected from a library of spike templates and then scaled according to the cells distance from the electrode [9, 10]. Apart from the amplitude scaling, template based models do not capture any spatial variations in the shape of measured spike waveforms. Therefore, despite their computational simplicity, they are not suitable for simulating recordings with multiple and/or positionally electrodes. A possible solution to this limitation is to first employ a compartment model to calculate spike waveforms on a three dimensional grid of measurement points surrounding the neuron and then to interpolate the resulting waveforms to obtain waveforms in measurement points not lying on the simulation grid [11]. Despite the increased level of realism introduced with this approach, it requires extensive waveform interpolation and may therefore not be suitable for simulating very large populations of neurons.

In hybrid recordings, synthetic spike trains are overlaid on real recordings of background noise [12]. They are thus advantageous in the sense that they have highly realistic signal properties, but lack in controllability for the same reasons as simultaneous intra- and extracellular recordings.

Considering the above, there is an obvious trade-off between realism and computational complexity when selecting among the available modeling approaches. Despite the ever increasing availability of computational resources

that indeed contributes to minimizing the impact of this trade-off, we reason that computational efficiency should be striven for. General availability to fast and simple ways of modeling complex recording scenarios would be of great value to researchers during the development of algorithms for signal processing. The possibility of quickly generating test data to match a specific recording setup would speed up the development phase and save valuable time.

In this paper, we present a novel, computationally and memory efficient approach to generating test signals that combines the detail of compartment models with the computational simplicity of template based models. To achieve this combination, we applied traditional dimensionality reduction techniques and polynomial fitting to compress the description of the spatial dependency in spike waveforms provided by compartment models. We used the *NEURON* simulation environment [13] to simulate an action potential in a compartment model of a CA1 pyramidal neuron originally modeled in [14] and used in [7] and computed the extracellular spike waveforms on a three dimensional grid of measurement points using the line source approximation (LSA) [15]. We then performed singular value decomposition (SVD) on the matrix containing the calculated spike waveforms and thereby obtained a set of basis waveforms describing the original spike matrix and their respective contributions to each of the original waveforms. Since most of the information describing the waveforms is contained in the first few (six) [16, 17] components of this decomposition, we achieved dimensionality reduction (compression) by discarding all other components. The result of this was a trivariate field of six dimensional vectors, whose elements described the basis waveform weights as functions of the measurement point coordinates relative to the neuron in question. To obtain a compact description of the spatial dependency of the basis waveform weights, we individually fit the elements of the weight vector field to polynomial functions of the measurement point coordinates. The modeling procedure was carried out for four different neuronal compartment models (cases A to D in [7]) and the parameters of the compressed models were optimized for each neuron to provide a good match between the spike waveforms provided by the NEURON simulations and our compressed models. The models were implemented into an object oriented simulation tool, written in *Matlab*, that facilitates fast and realistic simulations of multielectrode recordings with arbitrary geometries. Model validation was performed by comparing spikes from the original NEURON simulations with spikes generated by our models in terms of shape and amplitude, as well as by examination of synthetic signals in terms of noise properties. The applicability of our approach was evaluated in an example application by estimating the performance in spike sorting as a function of electrode position.

2 Methods

2.1 Neuron models

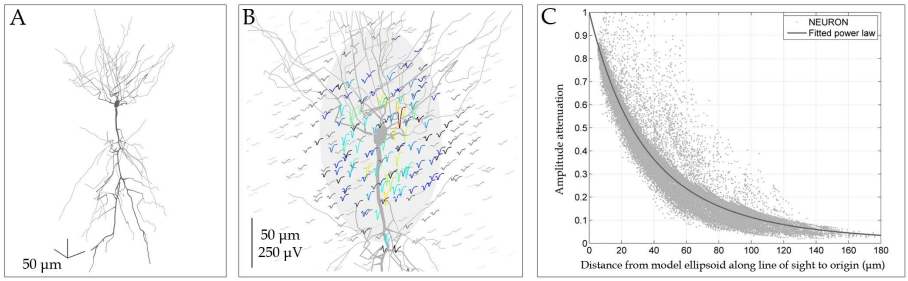


Figure 1: An illustration of the procedure of modeling the spatial dependency of the measured spike waveform for one of the neurons considered (neuron 1). (A) The CA1 pyramidal neuron model adopted from [7] was used to calculate extracellular spike waveforms in measurement points surrounding the neuron. (B) Spikes within an ellipsoid (overlaid ellipsoid) centered in the origin (cell soma) were used to derive the model. The ellipsoid was taken as the largest inscribed ellipsoid into the volume where spike amplitudes (maximum absolute amplitude) were at least A_{min} (typically around 20 μV. The spike waveforms are color coded according to their maximum amplitude (blue and red indicate low and high maximum amplitude respectively). (Note that for the sake of clarity, not all initial waveforms are shown here.) (C) Spikes with amplitudes below A_{min} (measured in points outside the model ellipsoid in (B)) were used to model the amplitude attenuation as a function of distance from the model ellipsoid along a line of sight from the measurement point to origin. (Continued on page 177)

Figure 1 illustrates the procedure we followed to derive the compressed neuron models. We used the CA1 pyramidal neuron compartment models employed in [7] as a starting point for obtaining spike waveforms on a three dimensional grid of measurement points around the neuron. An action potential was simulated in the model neuron with four different ionic channel densities (referred to as cases A to D in [7], referred to here as *neuron 1* to *neuron 4*) and the extracellular spike waveform was calculated in measurement points on a three dimensional grid surrounding the neuron using the line source approximation (LSA) [15]. The measurement points were distributed within a volume of $140 \times 140 \times 140$ μm and the spacing between the points was varied between 5 and 20 μm in each dimension (x, y, z). Close to the cell soma

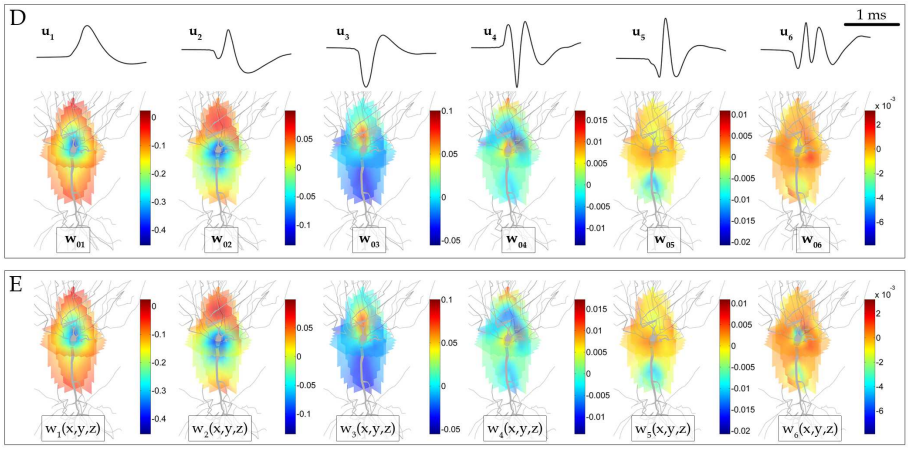


Figure 1: An illustration of the procedure of modeling the spatial dependency of the measured spike waveform for neuron 1 (continued from page 176). (D) Singular value decomposition was used to find an orthonormal set of basis waveforms $\mathbf{u}_n$ and their weights $\mathbf{w}_{0n}$ describing spikes within the model ellipsoid. (E) The weight distributions were then individually fit to trivariate polynomial functions of measurement point coordinates, $w_n(x, y, z)$.

($\{x, y, z\} \leq \{60\}\mu m$), the spacing was 5 μm and further away, it was successively increased to 10 μm and 20 μm. This resulted in a total of 42.875 initial measurement points.

To verify that this measurement point density was sufficient, we calculated the correlation coefficients between spike waveforms in all pairs of measurement points within a distance of 60 μm from the origin and then calculated the mean and standard deviation of the correlation coefficient as a function of distance between measurement points. The mean minus one standard deviation was above 0.99 for all neurons at a measurement point distance of 5 μm. At a measurement point distance of 20 μm, the mean minus one standard deviation of correlation coefficients was above 0.97. This indicates that the chosen measurement point densities were adequate to capture the spatial variation in spike waveforms.

The first step in the modeling procedure for each of the neurons was to find a volume within which the model would be derived. For points outside that volume, amplitude scaling was applied (discussed later in this section). The volume was taken as an ellipsoid, concentric with the cell soma, inscribed into the volume bounded by the measurement points in which the spike amplitude

exceeded a case-specific value of A_{min}. Spike amplitude was defined as the maximum absolute amplitude of the spike waveform. Finding the optimal value of A_{min} was part of a model selection procedure that is discussed later in this section. Having identified the model ellipsoid, spikes within the ellipsoid were arranged into columns of the spike matrix $\mathbf{S}_0$. $\mathbf{S}_0$ was then decomposed using singular value decomposition (SVD) according to

$$\mathbf{S}_0 = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{U}\mathbf{W}_0 \tag{1}$$

where the columns of the matrix $\mathbf{U}$ contain an ordered set of orthonormal basis waveforms describing the original spike matrix $\mathbf{S}_0$ and the columns of the matrix product $\mathbf{\Sigma}\mathbf{V}^T = \mathbf{W}_0$ contain the contributions (weights) of each of the basis waveforms in constructing the original set of spike waveforms in $\mathbf{S}_0$. Since most of the spike waveform variation is described by the first few basis waveforms, we discarded all but the first six components of the decomposition to achieve a dimensionality reduction [16, 17]. In order to assure that no information about spike waveform variability was lost by discarding the other components, we calculated the amount of total variance described by the first six components as

$$p_6 = \frac{\sum_{n=1}^{6} \sigma_n^2}{\sum_{n=1}^{N} \sigma_n^2} \tag{2}$$

where σ_n is the n-th singular value. This ratio was larger than 0.99 in all cases, indicating that the first six components adequately described the waveform variability.

We now modeled the weight of the n-th basis waveform as a trivariate polynomial function of the measurement point coordinates (x, y, z) in a coordinate system with origin in center of the cell soma, i.e.

$$w_n(x, y, z) = \sum_i c_{i_n} x^{\mathbf{e}_{i,1}} y^{\mathbf{e}_{i,2}} z^{\mathbf{e}_{i,3}} \tag{3}$$

where c_{i_n} is the i-th polynomial coefficient and $\mathbf{e}$ is a matrix whose i-th row contains the i-th combination of x, y and z exponents included in the model. For instance, if the i-th polynomial term is $c_i x^3 y^2$, the corresponding row in the exponent matrix $\mathbf{e}$ is [3 2 0]. The exponent matrix was constructed by assuming maximum orders of pure terms (including only one of the three coordinates) and mixed terms (including more than one of the three coordinates). These orders were referred to as N_{pure} and N_{mixed} and together with the minimum spike amplitude A_{min}, they determined the model properties and were selected to provide a good match between original and modeled spike waveforms (discussed later in the current section).

The polynomial fitting was performed by solving the equation system

$$(\mathbf{AD})\mathbf{C} = \mathbf{W}_0$$

$$\begin{bmatrix} x_1^{\mathbf{e}_{1,1}} y_1^{\mathbf{e}_{1,2}} z_1^{\mathbf{e}_{1,3}} & x_1^{\mathbf{e}_{2,1}} y_1^{\mathbf{e}_{2,2}} z_1^{\mathbf{e}_{2,3}} & \cdots & x_1^{\mathbf{e}_{I,1}} y_1^{\mathbf{e}_{I,2}} z_1^{\mathbf{e}_{I,3}} \\ x_2^{\mathbf{e}_{1,1}} y_2^{\mathbf{e}_{1,2}} z_2^{\mathbf{e}_{1,3}} & x_2^{\mathbf{e}_{2,1}} y_2^{\mathbf{e}_{2,2}} z_2^{\mathbf{e}_{2,3}} & \cdots & x_2^{\mathbf{e}_{I,1}} y_2^{\mathbf{e}_{I,2}} z_2^{\mathbf{e}_{I,3}} \\ \vdots & \vdots & \cdots & \vdots \\ x_L^{\mathbf{e}_{1,1}} y_L^{\mathbf{e}_{1,2}} z_L^{\mathbf{e}_{1,3}} & x_L^{\mathbf{e}_{2,1}} y_L^{\mathbf{e}_{2,2}} z_L^{\mathbf{e}_{2,3}} & \cdots & x_L^{\mathbf{e}_{I,1}} y_L^{\mathbf{e}_{I,2}} z_L^{\mathbf{e}_{I,3}} \end{bmatrix} \mathbf{D} \cdots$$

$$\begin{bmatrix} c_{1_1} & c_{1_2} & \cdots & c_{1_6} \\ c_{2_1} & c_{2_2} & \cdots & c_{2_6} \\ \vdots & \vdots & \cdots & \vdots \\ c_{I_1} & c_{I_2} & \cdots & c_{I_6} \end{bmatrix} = \begin{bmatrix} w_{1_1} & w_{1_2} & \cdots & w_{1_6} \\ w_{2_1} & w_{2_2} & \cdots & w_{2_6} \\ \vdots & \vdots & \cdots & \vdots \\ w_{L_1} & w_{L_2} & \cdots & w_{L_6} \end{bmatrix} \tag{4}$$

where L is the number of measurement points used in the fitting, I is the total number of polynomial terms in the fitted model, $\mathbf{A}$ is the multivariate Vandermonde matrix, $\mathbf{D}$ is an $I \times I$ diagonal matrix whose i-th diagonal element is the reciprocal of the Euclidean norm of the i-th column of $\mathbf{A}$, $\mathbf{C}$ is the coefficient matrix to be estimated (c_{i_p} is the estimated polynomial coefficient of the i-th term for the p-th basis waveform) and $\mathbf{W}_0$ is the original weight matrix (w_{l_p} is the weight of the p-th basis waveform in the l-th measurement point). The purpose of the matrix $\mathbf{D}$ was to scale the columns of the Vandermonde matrix to improve the conditioning of the problem. The number of measurement points was in all cases larger than the number of polynomial terms (i.e. $L > I$). The equation system was thus overdetermined and solving it yielded a least-squares solution.

For measurement points outside the model ellipsoid (in the far-field of the neuron) we assumed the measured spike waveform to be an attenuated version of the spike waveform measured in the point of intersection between the model ellipsoid and the line of sight from the measurement point to the origin. We assumed the attenuation g to be a power-law function of the distance r between the measurement point and the point of intersection, i.e.

$$g(r) = \frac{1}{(1 + a_{far} r)^{b_{far}}}. \tag{5}$$

The coefficients a_{far} and b_{far} were estimated by fitting the amplitudes of spikes with amplitudes below A_{min} to a power-law function of their corresponding measurement point distances (along the line of sight to origin) to the model ellipsoid. The power-law was estimated assuming the distance r to be in micrometers. Thus, the unit of the coefficient a_{far} is $[\mu m^{-1}]$. The form of the power-law was chosen to provide an attenuation of 1 at a distance of 0 from the

model ellipsoid. This way of modeling the spike waveforms in the far-field assured a continuous variation in the spike waveform when moving the electrode out of the model ellipsoid and between points outside the model ellipsoid.

As mentioned before, each neuron model was characterized by three parameters – A_{min}, N_{pure} and N_{mixed}. For each of the neurons, we performed the modeling procedure for all combinations of model parameters in the ranges $A_{min} \in [16, 26]\mu V$ (steps of 2 μV), $N_{pure} \in [10, 24]$ (steps of 1) and $N_{mixed} \in [2, 8]$ (steps of 2), resulting in a total of 360 models per neuron. The spike waveforms calculated by each of the models were compared with those obtained from the original NEURON simulations and a score was assigned to each model based on how the waveforms matched in terms of shape and amplitudes. To lower the computational demand during the comparison, we selected two random sets of measurement points to use in the comparison – 20% of the entire set of points within the near field (NF, inside the model ellipsoid) and 20 % of the entire set of points in the far field (FF). The following metrics were calculated to obtain the model scores:

- $e_{NF_1} = 1 - \text{mean(correl. coeffs. between spikes in NF)}$
- $e_{NF_2} = \text{std(correl. coeffs. between spikes in NF)}$
- $e_{NF_3} = \text{mean(abs. diff. between spike amplitudes in NF)}$
- $e_{NF_4} = \text{std(abs. diff. between spike amplitudes in NF)}$
- $e_{FF_1} = \text{mean(abs. diff. between spike amplitudes in FF)}$
- $e_{FF_2} = \text{std(abs. diff. between spike amplitudes in FF)}$

The metrics were normalized to range from 0 to 1, 0 indicating the closest match and 1 the worst match. Based on the normalized metrics, the following model scores were then defined ($\hat{\cdot}$ denotes the normalized metrics):

- Near field score:

$$s_{NF} = \sqrt{\hat{e}_{NF_1}^2 + \hat{e}_{NF_2}^2 + \hat{e}_{NF_3}^2 + \hat{e}_{NF_4}^2} \tag{6}$$

- Far field score:

$$s_{FF} = \sqrt{\hat{e}_{FF_1}^2 + \hat{e}_{FF_2}^2} \tag{7}$$

- Total score:

$$s_{tot} = \sqrt{s_{NF}^2 + s_{FF}^2} \tag{8}$$

We wanted to select a model that, apart from minimizing the total score s_{tot}, also minimized the difference between the near and far field scores. Thus, we selected the model that minimized the function

$$E = \sqrt{\Delta_s^2 + \hat{s}_{tot}^2} \tag{9}$$

where Δ_s is the normalized (0 to 1) absolute difference between near- and far-field scores and $\hat{s}_{tot}$ is the normalized (0 to 1) total score. This procedure consistently resulted in the automatic selection of a model that provided a high overall match with the original data while simultaneously performing well in both the near- and far-field.

Having selected the best model for a specific neuron, the model description was saved for implementation into the simulation algorithm. The main parameters included in the model were the basis waveforms (sampled at 25 kHz), the matrix product **DC**, the exponent matrix **e**, the axial radii of the model ellipsoid and the coefficients of the far-field attenuation power-law. The model parameters A_{min}, N_{pure}, N_{mixed} were also included for descriptive purposes. The model files were typically around 40kB of size which is three orders of magnitude smaller than the original spike matrix obtained from the NEURON simulations that were typically around 31MB.

The stored model parameters could now be used to efficiently calculate spike waveforms from neurons in a large set of arbitrary measurement points. Using the same procedure, we could also calculate the spike waveforms from a large set of neurons sharing the same neuron model in a single measurement point. Figure 2 illustrates this procedure. Assuming that we have a single electrode placed in (x_e, y_e, z_e) and N neurons where the n-th neuron is placed in (x_n, y_n, z_n), the waveforms from the neurons can be calculated in the following way:

Step 1: Calculate the relative positions of the electrode

For every neuron placed in (x_n, y_n, z_n), calculate the position of the electrode relative to that neuron, i.e.

$$(x'_n, y'_n, z'_n) = (x_e, y_e, z_e) - (x_n, y_n, z_n). \tag{10}$$

The problem is now that of calculating the spike waveform from a single neuron in N separate measurement points where the n-th point is (x'_n, y'_n, z'_n).

Step 2: Construct Vandermonde and attenuation matrices

For every (relative) measurement point, (x'_n, y'_n, z'_n), check if the point is

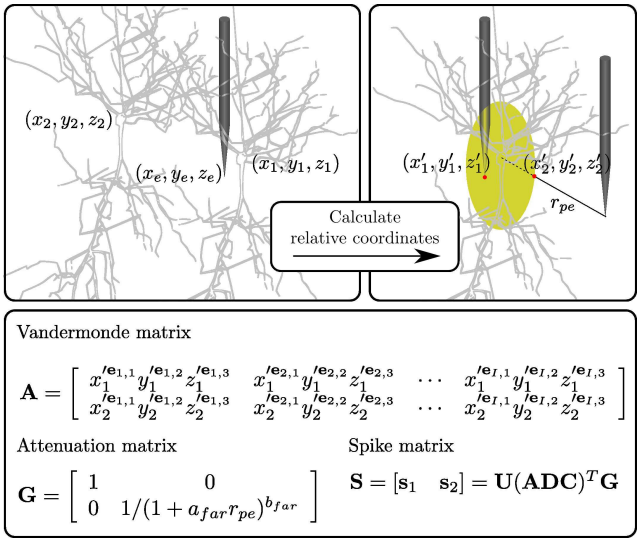


Figure 2: An illustration of how the derived models can be used to calculate the extracellular spike waveforms from two neurons measured with a single electrode. In the original coordinate system, the electrode is located in (x_e, y_e, z_e) and the neurons are located in (x_1, y_1, z_1) and (x_2, y_2, z_2) respectively. Assuming that the neurons are of the same type (share the same neuron model), the first step is to calculate the positions of the electrode relative to the two neurons, (x'_n, y'_n, z'_n). Relative measurement points inside the model domain of the neuron (yellow ellipsoid) are left unchanged (x'_1, y'_1, z'_1) and measurement points outside the model domain are replaced with the point of intersection between the model ellipsoid and a line of sight to origin. In the first case, the attenuation is set to 1 (element (1,1) in the attenuation matrix $\mathbf{G}$) and in the second case it is set to $1/(1 + a_{far} r_{pe})^{b_{far}}$ where a_{far} and b_{far} are estimated model coefficients and r_{pe} is the distance of the relative measurement point from the model ellipsoid along the line of sight to origin before it was replaced with the intersection point. The Vandermonde matrix is constructed using the exponent matrix $\mathbf{e}$ (obtained from the neuron model) and relative measurement points and finally the matrix $\mathbf{S}$ containing the spike waveforms in its columns is calculated with a simple matrix multiplication. The basis waveform matrix $\mathbf{U}$ and model coefficient matrix $\mathbf{DC}$ are parts of the derived model.

inside or outside the model ellipsoid by evaluating the quantity

$$r_{check} = \frac{x_n'^2}{r_{e_x}^2} + \frac{y_n'^2}{r_{e_y}^2} + \frac{z_n'^2}{r_{e_z}^2} \tag{11}$$

where $r_{e.}$ is the radius of the model ellipsoid along the -axis. If r_{check} is larger than 1, the point is outside the model ellipsoid and is thus replaced with the point of intersection between the line of sight to origin and the model ellipsoid. If r_{check} is smaller than or equal to 1, the point is inside the model ellipsoid and is left unchanged. If the point is outside the model ellipsoid, the n-th diagonal element of the $N \times N$ diagonal attenuation matrix $\mathbf{G}$ is set to $1/(1 + a_{far} r_{pe})^{b_{far}}$ where r_{pe} is the distance of the point to the ellipsoid along the line of sight to origin. If the point is inside the model ellipsoid, the attenuation value is set to 1. After performing the above check and replacing/keeping the relative measurement point coordinates, the l-th element of the n-th row of the Vandermonde matrix $\mathbf{A}$ is set to $x_n'^{\mathbf{e}_{l,1}} y_n'^{\mathbf{e}_{l,2}} z_n'^{\mathbf{e}_{l,3}}$.

Step 3: Calculate the spike matrix
The spike matrix $\mathbf{S}$ whose n-th column contains the modeled spike waveform from the n-th neuron can now be calculated as

$$\mathbf{S} = \mathbf{U}(\mathbf{ADC})^T \mathbf{G}. \tag{12}$$

Note that the above procedure assumes that all neurons share the same neuron model. If the volume to be simulated contains several types of neurons, the entire population of neurons can be divided into subpopulations according to type and the subpopulations can then be treated individually according to the above procedure.

2.2 Firing Models

Three models were implemented for generating spike times – gamma distributed inter spike intervals, bursting and correlated spike trains. In the current implementation, each neuronal population is assumed to have the same basic firing model, although the model parameters are set individually for each neuron. For instance, a population of neurons can have gamma distributed inter spike intervals, but each neuron in the population has an individual mean firing rate. For bursting neurons, inter-burst-intervals were assumed to be gamma distributed and the number of spikes within a burst was assumed to follow a Poisson distribution [18]. Changes in the spike waveform during a bursting period were not accounted for. We included the methods in [19] to generate correlated spike trains. Having used firing models to generate spike times for every neuron in the simulation volume, the measured signal at each recording channel was assembled by adding the calculated spike waveforms from each neuron at that channel at the spike times of that neuron in the same manner as described in [9].

2.3 Noise Models

We assumed noise to consist of two components, namely the spiking activity of distant neurons and thermal noise caused by random charge movements. This is a common way of modeling noise in extracellular recordings [10, 9, 20]. The thermal noise amplitude depends on recording bandwidth, temperature and input resistance of the recording electrode [21, 20] and we assumed it to be zero-mean normally distributed with a standard deviation determined by these parameters. We used the results presented in [20] to derive a quantitative model for setting the standard deviation. We approximated an extrapolation of the resistive part of the electrode impedance for an electrode size of 177 μm^2 to include frequencies from 100 Hz to 50 kHz and obtained an estimation of the power spectral density,

$$P(f) = 2kTR(f) \tag{13}$$

where k is the Boltzmann constant, T is temperature in Kelvin (set to 37°C) and $R(f)$ is the resistance as a function of frequency, f. The standard deviation of the thermal noise, $\sigma_{N_{th}}$, was then obtained as a function of recording bandwidth by taking the square root of the integral of the power spectral density over the recording bandwidth,

$$\sigma_{N_{th}}(f_B) = \sqrt{\int_0^{f_B} P(f)df} \tag{14}$$

where f_b is the recording bandwidth. A general description of this relationship was obtained by fitting $\sigma_{N_{th}}$ to a power-law function of $\log(f)$,

$$\sigma_{N_{th}}(f_B) = a_t \log(f_B)^{b_t}. \tag{15}$$

Physiological background noise was assumed to come from the spiking activity of distant neurons. To be able to make a distinction between the noise component of the signal and the spiking activity, we assumed the noise contributing neurons to be located at a minimum distance of r_i from the electrode (or origin) and we assumed them to have random mean firing rates selected from a uniform distribution between 1 and f_u spikes/second. The minimum distance r_i and the upper level of firing rates were then used to set the background noise level (see Section 2.5). Due to the computational efficiency of the methods described in Section 2.1, we were able to generate the background noise using the relative positions of the noise contributing neurons, thus employing the entire variability in spike waveforms described by the neuron models in the noise generation process also. Thus, although we make a distinction between

noise- and signal contributing neurons from the perspective of the recorded signal, the two categories of neurons were treated in exactly the same way in the simulation process. We did, however, include the possibility of not storing the true spike times for neurons far away from all recording electrodes, thus further decreasing the memory requirements and simulation time (See section 2.4).

2.4 Simulator Implementation

The simulator was implemented in *Matlab* using object oriented programming. We assumed the core components of the simulation to be the neuronal populations contributing to the signal, the array of electrodes recording the signals and a recorder that kept track of ground truth data and recorded signals. This abstract structure was implemented with three object models, one for each of these core components. A brief description of the properties of each object model follows:

The neuron class
: An object of the neuron class contains information about the properties of a population of neurons that share the same model, both in terms of the spatial dependency of spike waveforms and firing times. The information contained is the absolute coordinates of the neurons, the volume density within the population, a description of the volume containing the population, the spike model associated with the population and the firing statistics and spike times for the individual neurons. The neuron class has methods to generate spike times for its neurons based on the duration of the recording and the individual neuronal firing statistics. In order to simulate a specific recording setup where several types of neurons (in terms of spike and/or firing models) exist in specific regions, one neuron object is constructed for each population within the volume.

The electrode class
: An object of the electrode class contains the absolute coordinates of each electrode site and the spike waveforms from every neuron in every population calculated at the position each electrode. The electrode class contains methods to calculate the spike waveforms and to assemble the signal measured at each electrode site from the calculated spike waveforms and the spike times contained by the neuron objects.

The recorder class
: An object of the recorder class contains information about the structure of the HDF5 simulation file (see following paragraph) where the simulation

data is stored and methods to interact with the simulation file during and after simulation. This interaction includes writing the information contained in and generated by the neuron and electrode objects to the simulation file as well fetching the information once it is written to the file. The recorder class also contains several methods to visualize the simulation results.

In order to minimize memory requirements, we employed the HDF5 (*Hierarchical Data Format* file format [22] for data storage during simulation. The HDF5 file format is suitable for fast read and write access for large and complex datasets and allows database-like queries to be made once the file structure has been defined. Figure 3 illustrates the HDF5 file structure that we designed. When a *recorder object* is constructed in write mode, it creates a new HDF5 file for the recording that is to be simulated and then it provides read/write access to the simulation data as long as it exists. After simulation, a recorder object can be constructed in read mode with the name of the simulation file as input, thus allowing quick post-simulation access to all simulation data. The recorder object also allows for instance quick plotting of the synthetic signals, true spike waveforms as measured at the individual recording channels and the 3D geometry of the simulated volume.

2.5 Validation

The validity of our results was examined in terms of similarity between original and model-generated spike waveforms and noise properties of simulated recordings. The shapes of the spike waveforms were visually compared within the near-field (inside the model ellipsoid). Spike amplitude (maximum absolute amplitude) was examined as a function of distance from origin (cell soma) and by qualitative comparison of spike amplitude fields around the neuron. The amplitude fields were visualized by plotting three-dimensional isosurfaces around the neurons at spike amplitudes of 25, 50 and 100 μV. The spike amplitude distributions for the original data (NEURON generated spikes) were estimated by selecting approximately half of the original measurement points at random and calculating the mean and standard deviation of spike amplitudes in measurement points within 10 μm wide distance bins from the origin. For the spikes generated with our models, we used the same coordinates as for the true spikes, but with a small random shift. The random shift was introduced in order to make sure that the model captured the overall appearance of the amplitude distribution, even in measurement points that were off the original measurement point grid. In addition, evaluating the amplitude distribution in off-grid measurement points would reveal any potential problems

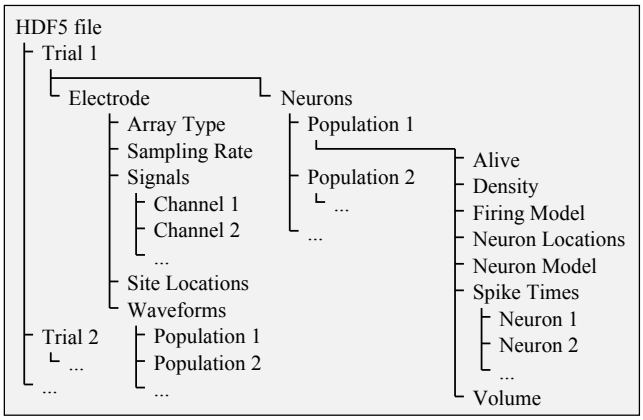


Figure 3: An illustration of how simulation data is organized into HDF5 files. In order to lower memory requirements, all information is written to the HDF5 file as soon as it becomes available during simulation. Upon starting a simulation, a new HDF5 file is created and a recorder object that provides read/write access to it is constructed. After the simulation is complete, it can be loaded in read-mode by calling the recorder object constructor function with the file name as input. The functions implemented in the recorder object provide fast access to all simulation data in a database-query type of way.

with "over-fitting". The spike amplitude isosurfaces were estimated in the same way, i.e. using an equally large random set of measurement points and applying a random shift for the model-generated spikes. Apart from these comparisons, we also included the metrics calculated during the model selection procedure (Section 2.1) as validity measures.

Noise properties of simulated single channel recordings were examined in terms of sample histograms, normalized power spectral density and standard deviation of noise. We assumed the noise properties to be mainly determined by the radius of the "silent volume" around the recording electrode and the distribution of firing rates among the neurons contributing to the background noise (see Section 2.3). To estimate the noise properties as functions of these parameters, we set up a recording scenario with a single electrode placed in the origin and we then created four populations of noise neurons (one population of each type of neuron) surrounding the electrode. The noise neurons were placed at random positions within a hollow cylindrical volume centered along the z-axis. The outer boundaries of the volume were defined by a cylinder with a radius of 250 μm and z between -250 μm and 250 μm. The inner boundaries

were defined by a cylinder with a variable radius r_i ranging from 50 μm to 150 μm and z between -150 μm and 250 μm. Assuming a neuronal density of 9.5×10^6 neurons/cm^3 [20], gamma distributed inter-spike intervals [9] and a minimum firing rate of 1 spike/second, we synthesized 30 second long noise recordings while varying the minimum distance of noise contribution neurons, r_i, and the upper limit of firing rates, f_u. r_i and f_u were varied between 50 and 150 μm and 5 and 80 spikes/second respectively. The recordings were synthesized at a sampling rate of 100 kHz, but were downsampled to 25 kHz and then bandpass filtered (300 Hz to 5 kHz). We then estimated the power spectral density using Welch's method, the sample amplitude histogram and the standard deviation of the resulting noise signal. Thermal noise was included since that was assumed to be an inevitable part of the recorded noise in a real situation. Besides allowing us to compare the noise properties of our simulator with those of previously reported simulators, this analysis provided basic means for controlling the noise properties by altering the parameters mentioned above.

2.6 Application Example: Spike Sorting Performance

The applicability of our work was evaluated by an example application in which we explored the effects of electrode position on the performance in spike sorting. Noise neurons were created in the same manner as described in the previous section. The inner radius of the the hollow noise cylinder was set to $r_i = 150\mu m$ and the upper limit of noise neuron firing rates was set to 50 Hz. Four target neurons (one of each type, cell 1 to 4) were placed in $(10, 20, -2)$ μm, $(-2, 18, 20)$ μm, $(-20, -5, 10)$ μm and $(16, -13, 15)$ μm respectively (Cartesian coordinates of cells 1 through 4, μm). All target neurons had gamma distributed inter-spike intervals and random mean firing rates between 1 and 10 Hz. Nineteen electrodes were placed along the z axis ($x = y = 0$) at positions ranging from $z = -30\mu m$ to $z = 60\mu m$ (5 μm spacing) and a 60 second long recording was synthesized (a close-up of the electrodes and the target neurons is shown in Figure 7 A).

Having obtained the HDF5 simulation file, we used the interface provided by the *recorder* class to extract the spike waveforms for each of the target neurons at each of the electrode sites at the known spike times. We thus obtained nineteen sets of extracted spike waveforms, each corresponding to one electrode position. The spikes from each position were then sorted separately and the sorting accuracy was estimated. Principal component analysis (PCA) [3] was used to extract spike features and the first two principal component weights (PC 1 and PC 2) were used to perform sorting of the spikes using K-means clustering [23]. Since we were only interested in comparing the performance in spike sorting while varying the electrode position, and not the absolute performance

of the selected spike sorting algorithm, we provided the true number of clusters (4 cells) to the K-means algorithm as input. We only employed the first two principal component weights in the clustering since that allowed for a straight forward visual interpretation of the spike sorting performance in terms of a two dimensional illustration of the PCA feature space representation of the spikes (Figure 7 D).

Having obtained the sorting results for a given set of spikes (a given position), the spike sorting accuracy was estimated in terms of true and false positive classification rates per cell (P_{TP} and P_{FP} respectively) and an overall sorting accuracy (P_{ID}). The true positive classification rate (P_{TP}) for a given cell in a given electrode position was calculated as the ratio between the number of spikes correctly assigned to that cell and the total number of spikes truly coming from that cell. False positive classification rate (P_{FP}) for a given cell in a given electrode position was calculated as the ratio between the number of spikes wrongfully assigned to that cell and the total number of spikes truly coming from any other cell. Overall spike sorting accuracy (P_{ID}) for a given electrode position was calculated as the ratio between the overall number of correctly classified spikes and the total number of spikes.

3 Results and Discussion

3.1 Model Parameters

The estimated neuron model parameters are summarized in Table 1. The minimum spike amplitude (A_{min}) included ranged from 18 to 24 μV, and the maximum degree of pure and mixed polynomial terms was 10 to 24 and 6 to 8 respectively. The model domain ellipsoid had a radius of approximately 45 to 65 μm in the $x-y$ plane and 104 to 142 along the z axis. A maximum distance of 50 μm between a neuron and the electrode is assumed for the neuron's spikes to be distinguishable from the background noise [1]. The model ellipsoids of all neurons approximately cover that range. For neuron 4, the x-axial radius is below 50 μm, which is explained by the smaller (in the $x-y$ directions) spike amplitude field for neuron 4 (see Figure 5 B), which in effect would lower the 50 μm distance threshold mentioned before.

The estimated power law describing the standard deviation of thermal noise as a function of recording bandwidth is shown in Figure 4. The parameters of the fitted power law according to Equation 15 where $a_t = 0.36$ and $b_t = 2.25$. The coefficient of determination between the standard deviations adopted from [20] and the fitted power-law was $R^2 = 0.98$, indicating a good match.

Table 1: A summary of derived model parameters. A_{min} is the spike amplitude threshold applied to determine the model ellipsoid volume. N_{pure} and N_{mixed} are the maximum orders of pure and mixed polynomial terms respectively and I is the resulting number of polynomial terms. (r_x, r_y, r_z) are the axial radii of the model ellipsoid and a_{far} and b_{far} are the coefficients of the amplitude decay power-law in the far field according to Equation 5.

Neuron	A_{min} (μV)	N_{pure}	N_{mixed}	I	(r_x, r_y, r_z) (μm)	(a_{far}, b_{far}) (μm^{-1}, unitless)
1	24	10	8	735	(53,58,104)	(6.8E-3,4.2)
2	18	16	8	753	(62,64,106)	(5.6E-3,4.3)
3	18	13	8	744	(65,78,142)	(7.4E-3,3.4)
4	22	24	6	397	(45,63,108)	(5.7E-3,4.1)

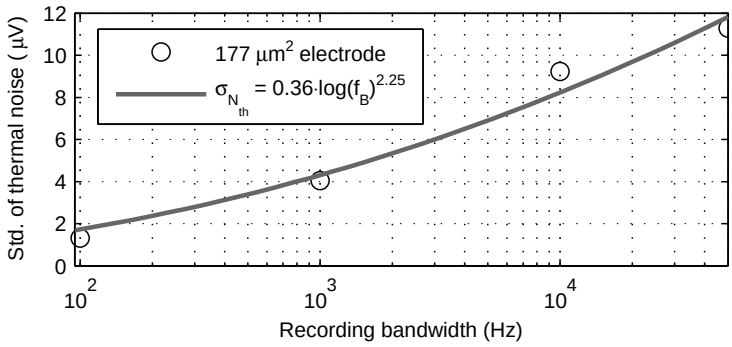


Figure 4: The standard deviation of thermal noise, $\sigma_{N_{th}}$ at a temperature of 37°C fitted to a power law function ($R^2 = 0.98$) of the logarithm of recording bandwidth f_B for an electrode of 177 μm^2 extrapolated from [20].

3.2 Validation

Spike Waveforms and Amplitude

All four models provided a good match in terms of spike waveforms and spike amplitudes when compared to the original data. Figure 5 A shows the true spike waveforms (black) and spike waveforms calculated by the neuron models (red dots) in an example set of measurement points. By visual inspection of these waveforms, we see that the models produce essentially identical waveforms to those generated by the original NEURON simulations. Mean correlation between true and modeled spike waveforms in the near field was larger than

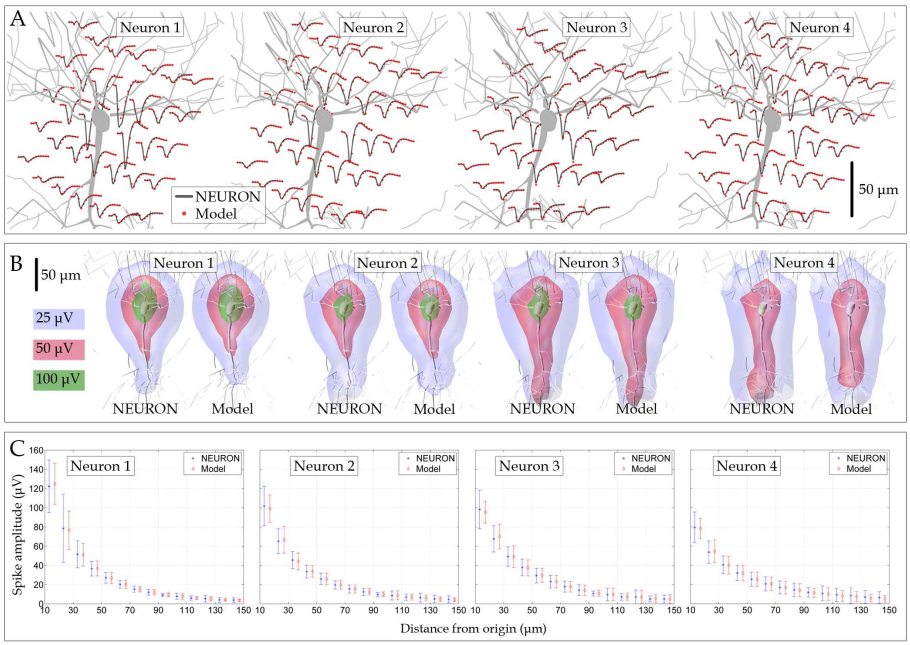


Figure 5: Validation of neuron models in terms of spike waveforms and spike amplitude. (A) Spike waveforms obtained with NEURON in an example set of measurement points (black) and spike waveforms calculated in those points by our models (red dots). (B) Spike amplitude fields displayed as spike amplitude isosurfaces at 25, 50 and 100 μV (blue, red and green respectively). (C) Spike amplitude (mean and standard deviation) as a function of distance from the origin (cell soma). The mean and standard deviation are taken across all spikes within 10 μm wide distance bins.

0.99 (standard deviation < 0.02) for all neurons (metrics e_{NF_1} and e_{NF_2}). Mean amplitude deviation in the near field was below 2 μV (standard deviation $< 5\mu V$) for all neurons (metrics e_{NF_3} and e_{NF_4}). Mean amplitude deviation in the far field was below 0.4 μV (standard deviation $< 2.1\mu V$) for all neurons (metrics e_{FF_1} and e_{FF_2}).

Figure 5 B shows the spike amplitude isosurfaces (25, 50 and 100 μV) for NEURON generated spikes and spikes generated by the compressed models. In all four cases, the compressed models capture the major features of the amplitude fields at all three amplitudes examined. This also applies to most "non-regular" features, such as the surface irregularities at the top of the 25

μV surface for neuron 4 and the following of the 50 μV surface along the axon for neuron 4.

Figure 5 C shows the mean and standard deviation of spike amplitudes as a function of distance from the origin (cell soma) within 10 μm wide distance bins. Also here, there is no noticeable difference between the original spike amplitudes and those produced by our models. In all cases, the models capture most of the local variations in spike amplitudes (for instance at local decrease in standard deviation of spike amplitudes at 90 to 100 μm for neuron 3). These local variations are caused by the non-uniform structure of the neuron, i.e. some points far away from the soma are in fact very close to other parts of the neuron.

Noise Properties

Figure 6 shows the noise level (σ_N), power spectral density (PSD) and sample histogram as functions of the upper limit of noise neuron firing rates, f_u, and minimum distance of noise contributing neurons, r_i. The figure shows that by varying those parameters we can control the amplitude and spectral properties of the recording noise. The figure also shows that we can obtain a good match with previously reported spectral properties and sample histograms [24, 10].

Increasing the maximum firing rate of noise neurons and decreasing the minimum distance to them increased the noise level and concentrated the noise towards the lower part of the spectrum, in which most of the spike energy is contained.

At large distances to the noise neurons (rightmost column in Figure 6), the change in noise neuron firing rate had less impact on the noise amplitude than at small distances (leftmost column in Figure 6). This observation can be interpreted in terms of how the variances of the contributions of individual noise neurons are influenced by their respective firing rates and distances from the recording electrode. In order to simplify this interpretation, we assume that the noise contributing neurons are statistically independent and that the variance of the spike train from a given neuron is approximately linearly dependent on the neuron's firing rate. Then, at a given distance, a linear increase in firing rate will cause a linear increase in variance. Since spike amplitude decreases with distance as a power law, this linear increase in variance with an increase in firing rate will be larger as the distance becomes smaller.

Thus, if the variance of the spike train from the n-th neuron σ_n^2 relates to the neuron's firing rate f_n and the neuron's distance from the electrode r_n as a power law function of the distance, scaled with the firing rate, or

$$\sigma_n^2 \sim \frac{f_n}{r_n^m} \tag{16}$$

where m is the amplitude power law coefficient, the variance of the total recording (sum of all N spike train variances due to the statistical independence assumption) relates to the properties of the individual neurons as

$$\sigma_N^2 \sim \sum_{n=1}^{N} \frac{f_n}{r_n^m}. \tag{17}$$

Therefore, an overall increase in the firing rate of noise neurons makes the standard deviation of noise (the square root of the variance) more sensitive to an overall decrease in the distance to noise neurons.

Application Example: Spike Sorting Performance

Figure 7 A shows the example recording scenario considered in our application example. The recorded signal at four example locations ($z = -20\mu m$, $z = 10\mu m$, $z = 30\mu m$ and $z = 50\mu m$) is shown in Figure 7 B along with extracted spike waveforms (mean waveforms $\pm$ standard deviation) for each of the four neurons at each of the four example locations. Figure 7 C shows the true and false positive classification rates (P_{TP} and P_{FP} for the individual neurons as well as the overall classification performance (P_{ID} and $1 - P_{ID}$) as functions of the electrode position (z). Finally, Figure 7 D shows the PCA feature space development (first two PC weights) for the extracted spikes at ten example positions.

As expected, spike sorting performance varied significantly with the electrode position, both in terms of overall performance (P_{ID}) and for individual neurons (P_{TP} and P_{ID}). Overall performance (P_{ID}) was maximal at $z = 30\mu m$, which also appeared to generally provide the best performance with regard to individual neurons.

The example locations in Figure 7 A were selected to demonstrate the varying similarity between the spikes coming from different neurons, depending on the electrode position. At $z = -20\mu m$, the true positive classification rates (P_{TP}) were low for neurons 1 and 2 in comparison to neurons 3 and 4. At the same position, the false positive classification rates (P_{FP}) for neurons 1 and 2 were high. Also, the true and false positive classification rates for neuron 4 were almost maximal and minimal, respectively, and for neuron 3, essentially the same applied. At $z = 30\mu m$, all neurons had similar true and false positive classification rates, those being high and low respectively. At $z = 50\mu m$ however, the performance was low for neurons 1 and 3, but high for neurons 2 and 4.

This varying performance can be explained both in terms of the varying similarity between spike waveforms (Figure 7 B, right part) and how the PCA clusters develop as the electrode position is altered (Figure 7 D). At $z = -20\mu m$,

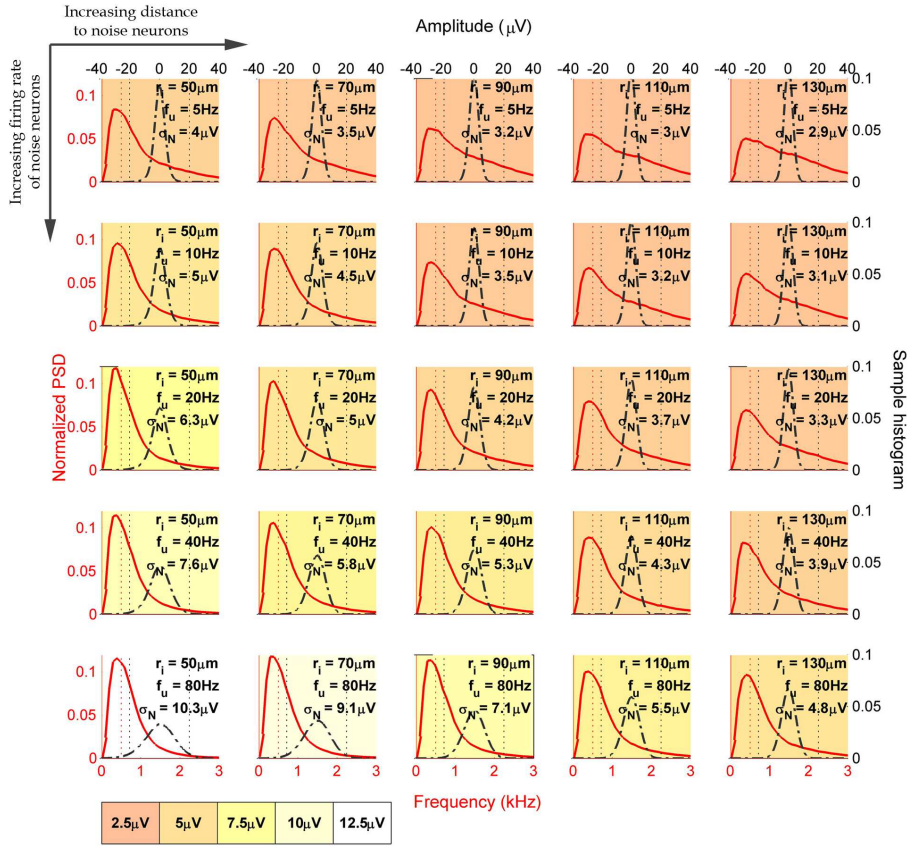


Figure 6: The recording noise properties as functions of the minimum distance of noise contributing neurons, r_i, and the upper limit of their firing rates, f_u. The individual firing rates of the neurons were selected from uniform distributions between 1 and f_u spikes/second. The red solid line show the normalized power spectral density (PSD) of the noise, the black broken line shows the sample histogram and the text inset shows the values of r_i and f_u for each case as well as the obtained standard deviation of the noise, σ_N for each case. The background color indicates the standard deviation of the noise.

the spike waveforms from neurons 1 and 2 were very similar and those from neuron 3 were somewhat similar to those from neurons 1 and 2. However, spikes from neuron 4 had a distinctive shape when compared to all other neurons.

This is clearly seen in the PCA feature space where clusters 1 and 2 overlap heavily, cluster 3 is close to, but not overlapping clusters 1 and 2, and cluster 4 is well isolated from all other clusters.

At $z = 30\mu m$, all waveforms had distinctive characteristics, which was also reflected in the PCA feature space, where all clusters were well isolated. At $z > 30\mu m$, spikes from neurons 1 and 3 became more and more similar, which was seen in the PCA feature space as a gradually increased overlap between clusters 1 and 3.

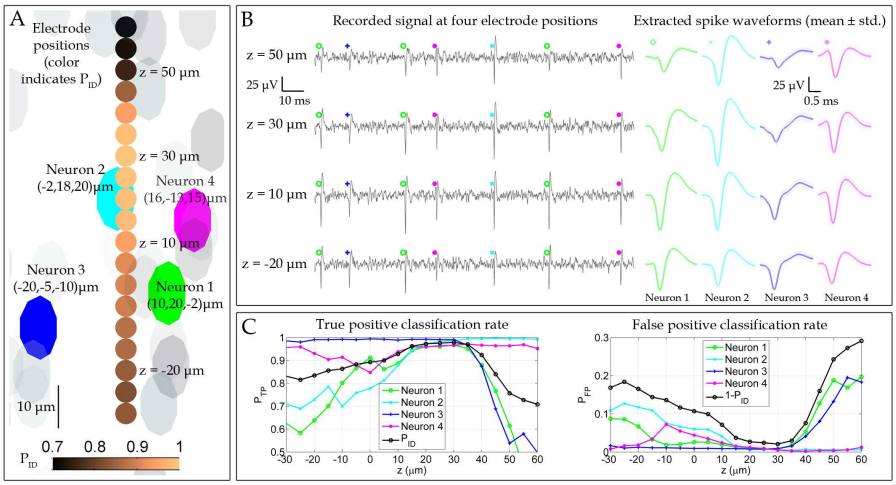


Figure 7: A demonstration of how our modeling and simulation can be used to evaluate spike sorting performance as a function of electrode position. (A) Nineteen electrodes (brown colored spheres along the center of the figure) were placed along the z axis ($x = y = 0$), each representing one electrode position to be evaluated. The electrodes were placed at $z = -30\mu m$ to $z = 60\mu m$ with a spacing of 5 μm. Four target neurons (neurons 1 to 4, green, light-blue, blue and purple ellipsoids) were placed close to the array of electrodes and noise neurons (gray ellipsoids) were placed far away. The size of the neurons corresponds approximately to the size of the cell soma in the NEURON model (see Figure 1). (B) Known spike times were used to extract spike waveforms from the recorded signals at each of the nineteen electrode locations and the extracted spike waveforms were sorted using principal component analysis (PCA) for feature extraction and K-means clustering for classification. (C) At each electrode location, true and false positive classification rates (P_{TP} and P_{FP}) were calculated for the individual neurons and the overall classification performance (P_{ID}) was estimated. (Continued on page 196)

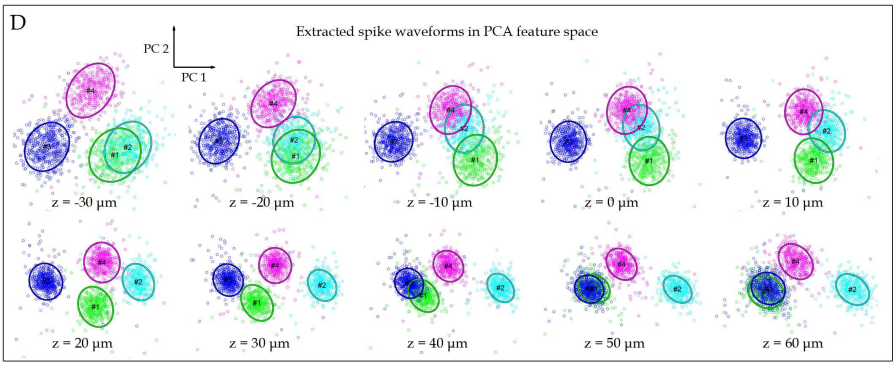


Figure 7: A demonstration of how our modeling and simulation can be used to evaluate spike sorting performance as a function of electrode position (continued from page 195). (D) The first two dimensions of the PCA feature space development at every other electrode position, showing how the overlapping of the clusters varied with the electrode location. The varying overlapping of clusters in (D) and the varying similarities/dissimilarities between spike waveforms in (B) relate directly to the varying classification performance that is evident in (C).

4 Conclusions

In this paper we have described a novel approach for generating synthetic test signals to facilitate the development and testing of signal processing algorithms for neuronal signals. Our approach combines the powers of compartment models and template based signal models to provide a computationally and memory efficient way of simulating large scale recordings without discarding the spatial variability in spike waveforms. We have shown that we can use traditional compression techniques to obtain a compact description of the spatial variability in measured spike waveforms predicted by compartment models and linear source approximations. The compressed models have been implemented into a simulation algorithm by which we generate synthetic spike trains as measured at an arbitrary number of electrode sites. The electrode sites can in turn be arbitrarily placed. The simulator has proved to be useful for providing synthetic multielectrode recordings in which the measured spike waveforms differ realistically between recording channels due to their different positions relative to the neurons. This property facilitates the development of algorithms for multichannel neuronal signal processing, the studying of the effects of electrode

array geometry on the performance in information extraction and the studying of algorithms to handle moving electrodes.

We have performed the modeling for four different compartment models and our validation procedures have shown that despite the heavy compression, we can use the model to recreate the major features of the spatial variation in spike shape and amplitude. They also show that by adjusting the minimum distance and maximum firing rate of noise contributing neurons, we can control the amplitude and spectral properties of the physiological background noise.

We emphasize that the modeling procedure we have presented here is in not restricted to compartment models of pyramidal cells. Due to the generic character of our method, constructing a database of compressed models for various types of neurons would provide a way of efficiently simulating the measured neuronal activity in specific brain structures where multiple types of neurons might be present.

In the initial compartment model simulations we generated 42.875 spike waveforms (corresponding to the same number of measurement points), each being 100 samples of length. A file containing the spike waveforms in these discrete measurement points was thus roughly 31MB of size. With our compressed models we are able to obtain a file size of around 40kB, or 775 times smaller than the original data matrix. Besides being smaller in size, the model is not restricted to discrete coordinates and thus does not require any waveform interpolation for off-grid measurement points as would the initial spike waveforms from the compartment model.

To underline the computational efficiency of our simulation approach, we measured the time it took to generate recordings of various lengths with one and four simulated recording channels respectively. The measured simulation time was approximately 1.2 seconds/minute/channel and we tested recording lengths up to 16 minutes. In [10] the authors reported that the simulation time was 270 seconds/minute for a single channel, or a factor 225 longer than what we observed with our simulator. With our previously reported simulator [9] we measured a simulation time of 16.5 seconds/minute for a single channel, or a factor of 13.75 times longer than the simulation time for the current simulator. Note that in contrast to the current simulation approach, the other simulators in this comparison neither capture the spatial variation in spike waveforms as a function of electrode position, nor do they facilitate the simulation of multiple electrode sites.

In comparison with other simulation approaches that have the ability to capture the spatial variation of spike waveforms, our simulator is significantly more efficient. An alternative approach would be to pre-calculate membrane currents for a given neuron model and use the LSA [15] to calculate spike waveforms in the given electrode locations, which corresponds to the first step in

the modeling procedure presented in this paper. When initially calculating the LSA, we obtained simulation times of around 0.1 second/waveform. In the application example presented in Section 2.6, a total of 744 neurons were present (740 noise neurons and 4 target neurons) and 19 electrode sites – requiring the total number of $744 \times 19 = 14.136$ spike waveforms to be pre-calculated. Assuming direct scaling of calculation time with the number of waveforms, the direct LSA approach would require a total time of approximately 24 minutes to calculate all spike waveforms at all electrode sites. However, using our method, the entire set of 14.136 spike waveforms was calculated in approximately 2.4 seconds, or a factor of 600 times faster.

Another alternative approach would be to pre-calculate the LSA on a grid of measurement points and interpolate the waveforms for off-grid measurement points, as discussed in the Introduction [11]. However, interpolating such a high number of waveforms from a grid of 42.875 measurement points would be significantly more demanding than calculating the matrix multiplication of Equation 12.

Judging from our results, we conclude that our current simulator is both very computationally and memory efficient and offers increased realism in terms of spike waveform variability compared to current state-of-the-art simulators. Future work includes improving the user interface of the simulator and making it available to the research community as a tool for providing multi-channel test signals with realistic properties.

Acknowledgments

This work was supported by a Linnaeus Grant from the Swedish Research Council (no. 60012701), a grant from the Knut and Alice Wallenberg Foundation (no. 2004.0119) and the Medical and Engineering Faculties at Lund University.

References

[1] G. Buzsáki, "Large-scale recording of neuronal ensembles.," *Nature neuroscience*, vol. 7, pp. 446–51, May 2004.

[2] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application.," *IEEE transactions on bio-medical engineering*, vol. 51, pp. 905–11, June 2004.

[3] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials.," *Network (Bristol, England)*, vol. 9, pp. R53–78, Nov. 1998.

[4] K. D. Harris, D. a. Henze, J. Csicsvari, H. Hirase, and G. Buzsáki, "Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements.," *Journal of neurophysiology*, vol. 84, pp. 401–14, July 2000.

[5] F. Franke, M. Natora, C. Boucsein, M. H. J. Munk, and K. Obermayer, "An online spike detection and spike classification algorithm capable of instantaneous resolution of overlapping spikes.," *Journal of computational neuroscience*, vol. 29, pp. 127–48, Aug. 2010.

[6] L. S. Smith and N. Mtetwa, "A tool for synthesizing spike trains with realistic interference.," *Journal of neuroscience methods*, vol. 159, pp. 170–80, Jan. 2007.

[7] C. Gold, D. a. Henze, and C. Koch, "Using extracellular action potential recordings to constrain compartmental models.," *Journal of computational neuroscience*, vol. 23, pp. 39–58, Aug. 2007.

[8] K. H. Pettersen and G. T. Einevoll, "Amplitude variability and extracellular low-pass filtering of neuronal spikes.," *Biophysical journal*, vol. 94, pp. 784–802, Feb. 2008.

[9] P. T. Thorbergsson, H. Jorntell, F. Bengtsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Spike library based simulator for extracellular single unit neuronal signals.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2009, pp. 6998–7001, Jan. 2009.

[10] J. Martinez, C. Pedreira, M. J. Ison, and R. Quian Quiroga, "Realistic simulation of extracellular recordings.," *Journal of neuroscience methods*, vol. 184, pp. 285–93, Nov. 2009.

[11] F. Franke, M. Natora, P. Meier, E. Hagen, K. H. Pettersen, H. Linden, G. T. Einevoll, and K. Obermayer, "An automated online positioning system and simulation environment for multi-electrodes in extracellular recordings.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2010, pp. 593–7, Jan. 2010.

[12] C. Pouzat, O. Mazor, and G. Laurent, "Using noise signature to optimize spike-sorting and to assess neuronal classification quality.," *Journal of neuroscience methods*, vol. 122, pp. 43–57, Dec. 2002.

[13] M. L. Hines and N. T. Carnevale, "The NEURON Simulation Environment," *Neural Computation*, vol. 9, pp. 1179–1209, Aug. 1997.

[14] C. Gold, D. a. Henze, C. Koch, and G. Buzsáki, "On the origin of the extracellular action potential waveform: A modeling study.," *Journal of neurophysiology*, vol. 95, pp. 3113–28, May 2006.

[15] G. R. Holt and C. Koch, "Electrical interactions via the extracellular potential near cell bodies.," *Journal of computational neuroscience*, vol. 6, no. 2, pp. 169–84, 1999.

[16] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-Gaussian variability.," *Journal of neuroscience methods*, vol. 69, pp. 175–88, Nov. 1996.

[17] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Statistical modelling of spike libraries for simulation of extracellular recordings in the cerebellum.," *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, vol. 2010, pp. 4250–4253, 2010.

[18] D. Heeger, "Poisson model of spike generation," *Handout, University of Standford*, pp. 1–13, 2000.

[19] J. H. Macke, P. Berens, A. S. Ecker, A. S. Tolias, and M. Bethge, "Generating spike trains with specified correlation coefficients.," *Neural computation*, vol. 21, pp. 397–423, Feb. 2009.

[20] S. F. Lempka, M. D. Johnson, M. a. Moffitt, K. J. Otto, D. R. Kipke, and C. C. McIntyre, "Theoretical analysis of intracortical microelectrode recordings.," *Journal of neural engineering*, vol. 8, p. 045006, Aug. 2011.

[21] R. Pettai, *Noise In Receiving Systems*. New York, NY: Wiley, 1984.

[22] "The HDF Group website - http://www.hdfgroup.org."

[23] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. New York, NY: Wiley, 2nd ed., 2001.

[24] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Variability of extracellular spike waveforms of cortical neurons.," *Journal of neurophysiology*, vol. 76, pp. 3823–33, Dec. 1996.

Paper VII

Compression of neural spikes with fixed generic bases for wireless brain-machine interfaces

Abstract

Brain-machine interfaces (BMIs) provide means of studying the neuronal mechanisms that govern behavior and they offer ways of overcoming disabilities caused by neurological disease. By removing the physical link between the subject and the external components of the BMI, wireless BMIs decrease the risk of post-surgical complications and increase the mobility of the subject. However, the information-carrying capacity of a wireless communication channel is limited by bandwidth, noise and various channel properties. When the number of recording channels in the BMI is increased, the channel's capacity immediately becomes a significant limitation. Data reduction techniques can be applied to overcome this limitation. First, the amount of raw acquired data can be minimized by minimizing sampling rate and resolution. Second, low complexity autonomous compression algorithms can be employed. In this paper, we address the second step in the data reduction procedure, focusing on compression with fixed compression bases. We show that detected neuronal spikes can be compressed with fixed generic compression bases without influencing spike reconstruction and spike sorting accuracies. Our results show that employing a fixed generic compression basis obtained by performing singular value decomposition on a matrix containing a large assembly of experimentally obtained spike waveforms that cover a wide range of shapes eliminates the need for implementing optimization procedures on the implant to pursue an optimal basis or the selection of optimal compression coefficients. This approach ensures a high degree of compression with minimal computational effort on the implant and thus also efficient utilization of the wireless channel capacity and minimum power consumption on the implant, both of which are highly desirable characteristics of wireless BMIs.

Based on: P. T. Thorbergsson, M. Garwicz, J. Schouenborg, A. J Johansson: "Compression of neural spikes with fixed generic bases for wireless brain-machine interfaces", manuscript to be submitted for publication, 2012.

1 Introduction

Brain-machine interfaces (BMIs) have become important research tools in neuroscience. By providing a bidirectional connection between the central nervous system (CNS) and the outside world, BMIs may be used for studying behavior at the level of neuronal mechanisms or control of external devices such as wheelchairs or a cursor on a computer screen. BMIs can be divided into different classes based on their level of invasiveness and means of signal acquisition. One such class employs intracranially implanted microelectrodes to pick up changes in extracellular voltage induced by activities of neurons surrounding them [1]. The signal resulting from such extracellular recordings is composed of spiking activity from neurons in the vicinity of the recording electrode, noise from distant neurons, local field potentials (LFPs) and thermal noise generated in the analog front-end electronics [2].

The detection, sorting and analysis of spikes in the recording allows the characterization of firing patterns of individual near-by neurons, that can in turn be correlated with events or learning processes in the motor or sensory domains [3]. The typical processing chain involves first detecting spikes and their timing in the recording, extracting the spike waveform that is assumed to extend over a short period (typically 2-3 milliseconds) of the signal immediately before and after the estimated spike time and finally using classification algorithms to sort the extracted waveforms and thereby assigning them to their neurons of origin. A spike alignment step is often included as well to increase the spike sorting accuracy [2]. Spike detection is commonly based on detecting the local increase in signal energy or amplitude followed by the occurrence of a spike. Spike sorting involves two steps – feature extraction and classification. The first step of those involves extracting features that are characteristic for the spikes and allow spikes from the same neuron to be grouped together in the second step.

Although the class of BMIs described above have great potential in both research and clinical applications, their level of invasiveness and need for wired connections are associated with risks of post-surgical complications and limitations of subject mobility. Wireless BMIs ideally solve both of these problems, since they remove the need for transcutaneous leads and they allow the subject to move around freely without restrictions posed by instrument cables.

Despite these advantages of wireless BMIs, they do not come without challenges as they are limited in terms of energy supply and information carrying capacity. The fundamental requirement that the implanted part of a wireless BMI is self-contained makes it necessary to consider low-power designs that maximize battery life and/or include means of wirelessly recharging the battery. Transmitting the raw signal from each electrode sampled at 25 kHz and

10 bits leads to a bitrate of 250 kbps per channel. According to Shannon, the maximum theoretical channel capacity for a Single-Input-Single-Output Additive White Gaussian Noise Channel is determined by the channel bandwidth, the signal-to-noise ratio (SNR) and the channel's transfer function [4]. For a lossless frequency-flat channel at a bandwidth of 1 MHz and an SNR of 0 dB, the maximum capacity is about 1 Mbps, which allows the raw data from four neural recording channels to be transmitted. Thus the channel capacity becomes an obvious bottleneck when increasing the number of neural measurement channels to hundreds or even thousands. The only ways to overcome this bottleneck are to either increase the channel capacity or to remove any redundancies from the data that are to be transmitted, the latter being the most effective and practically feasible. Apart from facilitating efficient use of the channel capacity, data reduction techniques also minimize the power used to transmit each bit of actual information.

Data reduction can be performed in two steps. The first step is to minimize the amount of raw acquired data by minimizing sampling rate and resolution and the second step is to implement low-power, automatic compression algorithms that extract only the relevant information from the recorded signal. The first step was addressed in [5], in which we found that a sampling rate and resolution of 16-31 kHz and 9 bits respectively were sufficient for maximizing performance in spike detection and spike sorting at realistic recording SNR. The second step – low-complexity autonomous compression algorithms that ensure maximal information transfer and channel utilization with minimal computational resources – is the subject of the present paper.

A common way of obtaining compression is to project the detected spike waveforms onto a set of sparsifying basis waveforms. Sparsification implies that the waveforms are mainly described by a small portion of the transform coefficients. By only considering these coefficients and discarding the rest, compression – or dimensionality reduction – is achieved.

The selection of a compression basis is crucial with regard to the sufficient represention of the compressed spike waveforms. Optimally, the compression basis is derived directly from the spike waveforms that are to be compressed by means of, for instance, singular value decomposition (SVD), ensuring that the majority of the data is described by a minimal number of compression coefficients – i.e. maximizing the sparseness of the data in the compression basis. The SVD provides compression coefficients that are ordered by significance and the coefficient selection simply involves selecting the first K coefficients. However, this approach requires that the computationally demanding task of finding the optimal compression basis is carried out for each scenario in which new spike waveforms are encountered and is thus unpractical in low-complexity autonomous implants.

Another approach, which is more feasible in terms of computational complexity on the implant, involves using a fixed generic compression basis. The basis then captures the "general" characteristics of spike waveforms that may be encountered in a wide range of scenarios and thus eliminates the need for adapting to the data in each case. Here, we divide the types of fixed compression bases into two major categories. The first category includes general transformation bases, such as the discrete Fourier transform (DFT), the discrete cosine/sine transform (DCT/DST) or the discrete wavelet transform (DWT). It also includes the transform representation of the simplest form of data reduction, namely that of downsampling, or discarding samples. These bases may be efficient in compressing spike data, but their common drawback is that the significance of the compression coefficients varies between cases, and therefore the implementation of an adaptive coefficient selection procedure is a prerequisite for optimal performance.

Fixed bases in the second category are derived from spike data by means of, for instance, SVD and are therefore similar to the optimal bases. However, these bases are not derived from the detected spike waveforms each time as in the optimal case, but from a large pre-recorded assembly of spike waveforms that cover a wide range of shapes. Thus, for a given recording scenario, it is likely that the majority of the information about the detected spike waveforms resides within the lower end of the coefficient spectrum, and the coefficient selection becomes straight-forward.

In [6] we used a large set of mean spike templates extracted from recordings in the cat cerebellum for the purpose of modeling extracellular recordings with known properties. In [7] we used principal component analysis and Gaussian mixture models to obtain a statistical model that could be used to interpolate the discrete experimentally obtained spike library used in [6]. In that second paper, we showed that six principal components were adequate to describe the spike library, that covered a wide range of spike morphologies. These results were in agreement with the results presented in [8], where a similar analysis was carried out. These findings indicate that the same principle could be used to obtain a generic fixed basis for compressing spike waveforms in a wireless BMI.

In this paper we study the compression of extracted spike waveforms with various system architectures, spike detectors and compression bases. We implemented various system architectures that involved spike detection, spike alignment, spike compression, spike reconstruction and spike sorting. The system architectures differed with regard to the presence and allocation of processing tasks. Spike detection was performed with *absolute value threshold detection* (ABS) and *nonlinear energy operator detection* (NEO). Five different compression bases were included. These were 1) the *optimal* basis in each case, 2) a fixed

basis derived (by SVD) from a large assembly of synthetic model-generated spikes (*fixed 1*), 3) a fixed basis derived (by SVD) from a large assembly of experimentally obtained spikes (*fixed 2*), 4) a *downsampling* basis and 5) a *Haar* wavelet basis. The simulator described in [9] was used to generate synthetic recordings with various noise levels and the recordings were then processed by all combinations of a system architecture, spike detector and compression basis and the results were compared in terms of spike reconstruction accuracy and spike sorting accuracy. These results allowed us to assess whether or not a given compression basis combined with a given spike detector and a given architecture would cause a decrease in performance in spike analysis compared to when the uncompressed spike waveforms were transmitted and analysed. Due to their generic nature and straight-forwardness in coefficient selection – both of which are beneficial characteristics in terms of lowering computational complexity on the implant – fixed compression bases of the second category were of special interest to us. The *fixed 2* basis represents this category and was therefore treated in more detail than the other compression bases. Our results show that compressing detected spike waveforms with a generic compression basis derived from an arbitrary set of spike waveforms can lead to similar spike sorting performance as the *optimal* basis, and only a somewhat lower accuracy in spike reconstruction.

2 Methods

2.1 Test Data

The simulator described in [9] was used to synthesize three nineteen-channel test recordings with varying signal-to-noise ratios (high, medium and low SNR). The recordings were five minutes long. In all three recordings, a linear array of nineteen evenly spaced electrodes was placed along the z-axis ($x = y = 0$, $-30\mu m \leq z \leq 60\mu m$, $5\mu m$ spacing). Noise neurons were placed at random positions (density of 9.5×10^6 neurons/cm^3 [10]) within a hollow cylinder concentric with the z-axis. The inner and outer boundaries of the hollow cylindrical volume were at 120 μm and 250 μm respectively and its floor and ceiling were at $\pm$ 250 μm respectively.

Four target neurons (one of each neuron model derived in [9]) were placed inside the hollow space of the noise neuron cylinder. For the high SNR recording, the neurons were placed at positions of (10,20,-2)μm, (-2,18,20)μm, (-20 -5 -10)μm and (16,-13,15)μm. For the medium and low SNR recordings, these coordinates were multiplied by factors of 1.5 and 2 respectively, i.e. moving each neuron along a linear path from the origin. Moving the neurons away

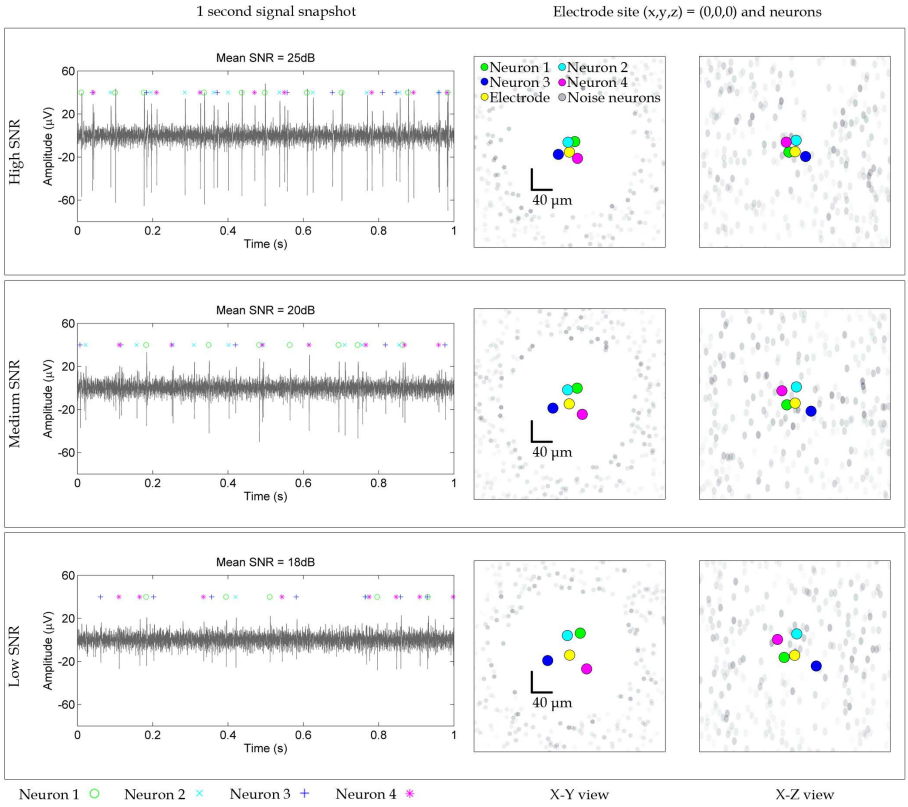


Figure 1: One of nineteen electrode sites and a snapshot of the corresponding signal from each of the multi-channel test recordings (high, medium and low). The SNR was varied by varying the distance of neurons 1 to 4 from the linear electrode array array placed along the z axis. The left part of the figure shows one second of the total of five minutes of recording with the true spike identities labeled and color coded. The right part of the figure shows the placements of the target neurons (colored dots around origin), noise neurons (gray dots far away from origin) and the electrode site $(x, y, z) = (0, 0, 0)$ (yellow dot in origin). The sizes of the indicators do not reflect the true sizes of the neurons and the electrode, but only their positions. Only the X-Y and X-Z views are shown. Note that each test recording contained eighteen additional electrode sites, arranged along the z-axis.

from the electrode array resulted in decreasing their spike amplitudes and thus decreasing the SNR since the noise neuron cylinder was not altered. The multiplication factors were chosen empirically to provide clearly varying SNRs and resulted in the neurons being at distances of approximately 20, 30 and 40 μm from the electrode array (in the $x-y$ domain) for the high, medium and low SNR respectively. Assuming that spikes from neurons within a distance of 50 μm can be detected [11], these distances are reasonable.

Figure 1 illustrates the arrangement of neurons and the electrode site used in each case for the first part of the performance estimation, i.e. $(x, y, z) = (0, 0, 0)$ (see later section) as well as a one second long segments of the signal in each SNR case. Note that each test recording contained eighteen electrode sites in addition to the one shown in Figure 1. Note also that the three test signals were generated individually, meaning that the actual locations of noise neurons and actual spike times of all neurons varied between the recordings. However, the statistical properties used to generate locations and spike times did not vary between the recordings. The recordings were sampled at 25 kHz and bandpass filtered between 300 Hz and 5 kHz.

All neurons were assumed to have gamma distributed inter-spike intervals [12]. For each noise neuron, a random mean firing rate was chosen from a uniform distribution between 1 and 50 spikes/second. For each target neuron, a random mean firing rate was chosen from a uniform distribution between 1 and 10 spikes/second.

Signal to noise ratio was calculated in a similar manner as described in [13]. For a given recording and a given electrode site, we defined the SNR for the n-th neuron as

$$SNR_n = 20 \log_{10} \left(\frac{s_{pp_n}}{\hat{\sigma_N}} \right) \tag{1}$$

where s_{pp_n} is the peak-to-peak amplitude of the mean spike waveform of the neuron measured at the electrode site and σ_N is the standard deviation of background noise estimated according to [14]

$$\hat{\sigma_N} = \text{median} \left(\frac{|v|}{0.6745} \right) \tag{2}$$

where v is the sampled signal. We then reported the mean SNR across the neurons for any given case.

2.2 System Architectures

Three different system architectures involving spike waveform compression were considered in our comparison (architectures 1 to 3 in Figure 2), in addition to

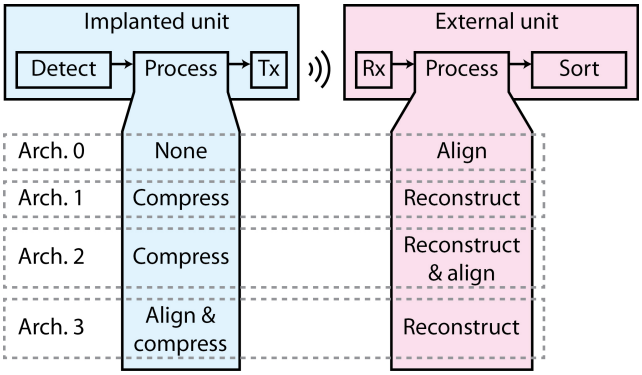


Figure 2: An illustration of the system architectures considered in the comparison. The "Tx" and "Rx" blocks refer to the wireless transmitter and receiver, respectively. In architecture 0 (the reference architecture) uncompressed spikes were transmitted, aligned and sorted. In architectures 1 to 3, spikes were compressed, reconstructed and sorted and in architectures 2 and 3, spike alignment was performed at different stages.

a reference architecture in which no compression was performed (architecture 0 in Figure 2). We assumed the main task of the system to be the characterization of the firing patterns of individual neurons through the detection and sorting of neuronal spikes in extracellular neuronal recordings. We assumed spike detection to be performed on the implanted unit and spike sorting on the external unit. Further, in order to minimize the computational burden on the implant, we assumed compression of detected waveforms to be primarily carried out with a fixed compression basis (see later section), so that no learning needed to take place on the implant.

The architectures varied in terms of the arrangement of processing tasks carried out between the spike detection and spike sorting tasks. The main intermediate processing tasks were assumed to be the compression and reconstruction of detected spike waveforms. A secondary – but nevertheless important – processing step, was assumed to be spike alignment. Each of these processing steps is discussed in more detail in the following sections.

Architecture 0 was included as the reference architecture, i.e. the case where the extracted spike waveforms were transmitted without compression and aligned on the external unit prior to spike sorting. In architecture 1, spikes were extracted and compressed on the implanted unit and reconstructed and sorted without alignment on the external unit. Architecture 2 was similar to architecture 1, but with the additional alignment step on the external unit

prior to spike sorting. In architecture 3, spikes were extracted, aligned and compressed on the implant and reconstructed and sorted on the external unit.

2.3 Spike Detection

We assumed *ABSolute value* (ABS) and *Nonlinear Energy Operator* (NEO) spike detection, both of which have been shown to provide a good combination of performance and computational complexity [15, 16, 5]. NEO has been shown to be more robust to background noise and provide less spike detection jitter which is beneficial for spike sorting, but its computational complexity is higher [15, 5, 17]. Spike detection jitter refers to the misalignment of extracted spike waveforms that arises when different spikes cross the detection threshold at different time instances within the waveform [2]. ABS is attractive due to its simplicity, but it requires an extra spike alignment step (see Section 2.4) due to the more severe spike detection jitter it introduces (see Figure 4). Spike duration was assumed to be 2.5 milliseconds.

In order to eliminate any errors in estimation of spike sorting accuracy caused by false positive detections, we used true spike times provided with the synthetic recordings to extract spike waveforms from the recordings and then introduced the spike detection jitter afterwards. Detection thresholds (see Sections 2.3 and 2.3) were used to find the detection time (time instance of threshold crossing) for each extracted waveform that did pass the threshold. Spikes that did not pass the threshold were discarded. We then used a discrete-time delay filter to shift each waveform in time to have the threshold crossing occur at the most frequent detection time across all the spikes. Spikes whose threshold crossing time deviated by more than 1.5 standard deviations from the most frequent value were discarded as outliers. These cases usually represented overlapping spikes. Spike waveforms were upsampled to a sampling rate of 100 kHz before introducing the jitter and were then downsampled to 25 kHz again afterwards. ABS and NEO spike detection are briefly described below.

ABSolute value Spike Detection

In ABS detection, a threshold of

$$T = 4\hat{\sigma}_N = 4 \cdot \text{median}\left(\frac{|v(n)|}{0.6745}\right) \tag{3}$$

where $\hat{\sigma}_N$ is the estimated standard deviation of background noise and $|v|$ is the digitized signal's amplitude, is applied to the absolute value of the signal [14]. This is equivalent to simultaneously applying a positive and a negative threshold to the raw signal.

Nonlinear Energy Operator Spike Detection

In NEO detection, a threshold T is applied to the nonlinear energy operator $\Psi(n)$ of the signal $v(n)$. The nonlinear energy operator is given by

$$\Psi(n) = v^2(n) - v(n+1) \cdot v(n-1) \tag{4}$$

and the threshold is taken as

$$T = 8 \cdot \overline{\Psi(n)} \tag{5}$$

where $\bar{\cdot}$ represents the sample mean [18, 15].

2.4 Spike Alignment

For the architectures that involved a spike alignment step (architectures 0, 2 and 3), spikes were aligned at their maximum absolute amplitude within a time interval of approximately half a millisecond after the detection time. This was assumed to correspond to aligning the spikes on the maximum value of the detected peak or valley. We chose maximum amplitude alignment due to its simplicity, since it only involves finding the maximum absolute value of the signal within a short time window. More sophisticated approaches have been reported, such as the center-of-mass alignment [2], which takes into account the entire waveform and is thus less sensitive to noise. However, it is more complex than maximum value alignment, and therefore less feasible to use in the implanted unit (architecture 3) from an energy consumption point of view. Therefore, we selected the approach that introduced the lowest complexity in the implanted unit and used that for the other architectures as well for the sake of comparison.

For architectures 0 and 2 (spike alignment on the external unit), spikes were upsampled to a sampling rate of 100 kHz prior to alignment and were then downsampled to 25 kHz after alignment. For architecture 3 (spike alignment on the implant), the alignment was performed at the initial sampling rate of 25 kHz in order to minimize the increase in computational complexity introduced by placing the alignment step in the implant. The alignment was performed using delay filters in the same way as when applying spike detection jitter (Section 2.3).

2.5 Spike Compression and Reconstruction

Spike compression was performed by projecting detected spike waveforms onto a sparsifying basis, referred to as *compression basis* and transmitting a fixed

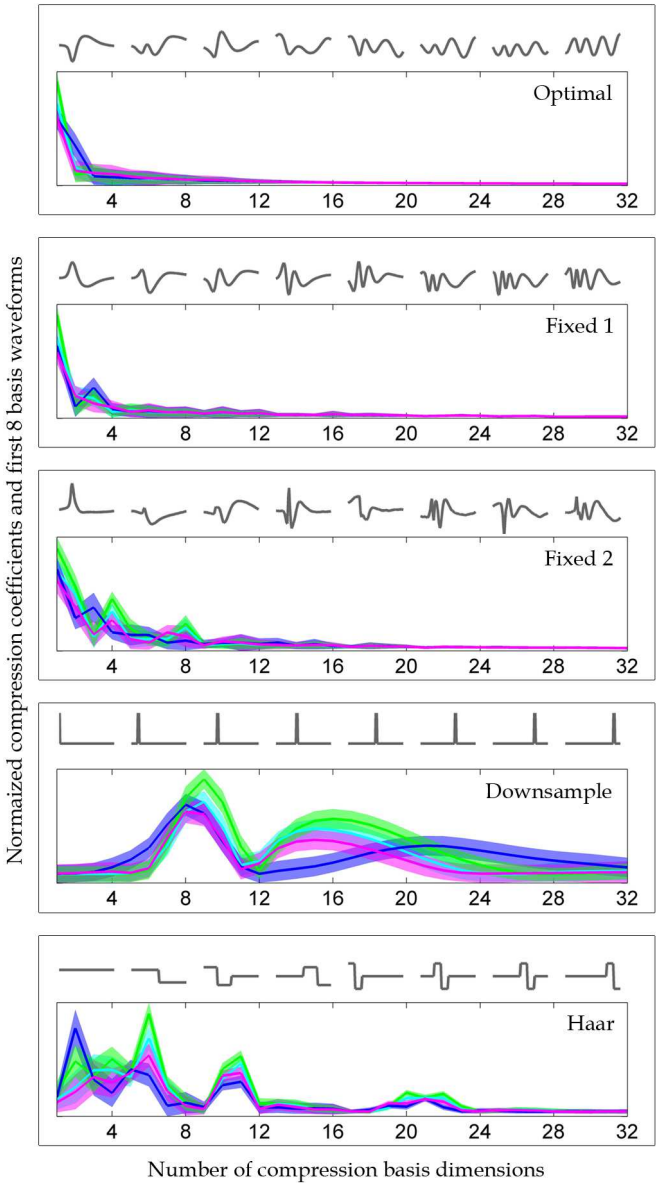


Figure 3: (Caption on page 217)

Figure 3: (Continued from page 216 caption) Eight basis waveform from the five compression bases that were included (*optimal*, *fixed 1*, *fixed 2*, *downsample* and *Haar*) and 32 compression coefficients of the spikes from the high SNR recording at electrode site $(x, y, z) = (0, 0, 0)$. The coefficients are given as mean plus/minus one standard deviation for the spikes of each of the four target units (green, light blue, dark blue and pink). The coefficient spectra show the varying sparsification levels provided by the different bases, the *optimal* basis providing the highest sparsification and the *downsampling* basis the lowest.

number of transformation (compression) coefficients. The transformation was obtained as

$$\mathbf{W}_c = \mathbf{B}_c^T \mathbf{S} \tag{6}$$

where the $M \times N$ matrix $\mathbf{S}$ contains the M sample long spike waveforms in its columns, the $M \times M$ matrix $\mathbf{B}_c$ contains the M sample long basis waveforms of the compression basis in its columns and the $M \times N$ matrix $\mathbf{W}_c$ contains the full set of transformation coefficients. Since compression involves discarding all but K of the total set of M ($K < M$) transformation coefficients, we introduced the $K \times N$ dimensionality reduction matrix $\mathbf{B}_d$. The transmitted coefficients were thus given by

$$\mathbf{W}_d = \mathbf{B}_d \mathbf{W}_c = \mathbf{B}_d \mathbf{B}_c^T \mathbf{S}. \tag{7}$$

The compression and dimensionality reduction bases were assumed to be known at the external unit and were used to reconstruct the spike waveforms according to

$$\hat{\mathbf{S}} = \mathbf{B}_c \mathbf{B}_d^T \mathbf{W}_d \tag{8}$$

where $\hat{\mathbf{S}}$ is the reconstructed spike matrix. For compression with the *downsampling* basis, the reconstruction involved an additional lowpass filtering step for interpolation (see Section 2.5).

We included five different compression bases in our analysis. These are briefly discussed below. Figure 3 illustrates the first eight basis waveforms of each basis and 32 compression coefficients in each basis for the high SNR recording at electrode site $(x, y, z) = (0, 0, 0)$. The distributions of compression coefficients show that all bases introduced sparsity, but to a varying degree. Apart from providing different levels of sparsity, the bases also provided different distributions of coefficients within the coefficient spectra. As touched upon in the introduction, the bases derived by SVD (*optimal*, *fixed 1* and *fixed 2*) have the attractive property of providing coefficients that are concentrated at the lower end of the spectrum, making the selection of coefficients straightforward (the first K coefficients are selected). This is in contrast to the *downsam-*

pling and *Haar* bases, in whose coefficient spectra the first K coefficients are not necessarily the most significant ones. Also, the SVD based bases generally tend to provide a greater degree of sparsification, indicating that they allow a smaller number compression coefficients to be transmitted.

Optimal Basis

This basis was found by performing singular value decomposition (SVD) on the matrix $\mathbf{S}$ containing the detected spike waveforms in its columns. The *optimal* basis was included as a reference case since it involves basis waveforms that are derived directly from the waveforms that are to be compressed. The basis waveforms were obtained as the columns of the unitary matrix $\mathbf{U}$ in the decomposition

$$\mathbf{S} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H. \tag{9}$$

Since the dimensions of the basis given by the SVD are arranged in a decreasing order of significance, the dimensionality reduction matrix $\mathbf{B}_d$ was taken as the first K rows of the $M \times M$ identity matrix. As mentioned previously, this is an attractive property of SVD-based compression bases and is due to the fact that most of the waveform information is concentrated in the lower range of the transform coefficients.

Fixed Basis 1

This basis was found by performing SVD on a matrix containing 40.000 synthetic spike waveforms obtained by calculating the measured spike waveform in 10.000 random measurement points surrounding each of the four model neurons addressed in [9]. This basis was assumed to represent the generic basis that was well tuned to the data, since it was derived from the same neuron models as the test data but not derived from the test data. The dimensionality reduction matrix was the same as that for the *optimal* basis.

Fixed Basis 2

This basis was obtained by performing SVD on the matrix of spike waveforms contained in the library used in [6]. Since this basis had no connection to the test data, it was assumed to represent the generic case, i.e. basis waveforms obtained from a set of spike waveforms extracted from an arbitrary set of recordings. The dimensionality reduction matrix was the same as that for the *optimal* basis. This basis was the most interesting one within the context of this paper, since it represents the generic compression basis where the derivation of the basis waveforms is entirely independent of the spike data that is to be compressed.

Downsampling Basis

This basis was included as the simplest form of data reduction, namely that where samples are simply discarded. The compression matrix $\mathbf{B}_c$ was taken as the $M \times M$ identity matrix and the dimensionality reduction matrix $\mathbf{B}_d$ was obtained by removing all but every R-th row from the $M \times M$ identity matrix where R was the downsampling factor obtained by rounding the ratio M/K to the nearest integer.

Having obtained the reconstructed spike matrix $\hat{\mathbf{S}}$ according to Eq. 8, the reconstructed waveforms were filtered in the frequency domain by a lowpass interpolation filter [19]. Note that since we wanted to examine the effects of simply discarding samples, no antialiasing filtering was applied prior to downsampling.

Haar Wavelet Basis

This basis was obtained by constructing the $M \times M$ Haar matrix, whose columns contain the discrete time Haar basis waveforms. Although not necessarily optimal, the dimensionality reduction matrix was taken as the first K rows of the $M \times M$ identity matrix. This choice was made since selecting the optimal transform coefficients to transmit would result in the need for implementing an optimization procedure on the implant, which would lead to a significant increase in complexity.

2.6 Spike Sorting

We performed feature extraction and clustering at the external unit with principal component analysis (PCA) and K-means respectively. In PCA, an ordered set of orthonormal basis waveforms is derived from the spike waveforms and the projections of the spikes onto the first P dimensions of this basis are used as features in spike sorting [20]. In K-means, data points are assigned to clusters that form gradually and ideally their means converge to the true cluster means [21]. We provided the true number of clusters (four neurons) as input to the K-means algorithm.

PCA is a widely used approach for feature extraction in spike sorting and has been shown to perform well in comparison to other feature extraction approaches, such as the discrete wavelet transform (DWT) and discrete derivatives (DD) [15]. For the DWT, this applies especially when the wavelet basis is badly tuned to the data [22] or when feature selection is not straightforward [5]. DD has been shown to provide similar performance as PCA, but as DWT, it requires a feature selection step [15]. We used the first three PCA weights as spike features. Since we assumed spike sorting to be performed at the external

unit, the need for prioritizing computationally simple spike sorting algorithms was essentially eliminated.

2.7 Estimation of Performance

System configurations (combination of a spike detector, system architecture and compression basis) were compared in terms of spike sorting accuracy and spike reconstruction accuracy. Spike reconstruction accuracy was calculated for each reconstructed spike as the maximum value of the cross-correlation function between the reconstructed spike and the mean spike for the neuron in question. This waveform similarity measure is similar to the one employed in [23]. Overall spike sorting accuracy (P_{ID}) for a given case was estimated in the same way as described in [9], i.e. in terms of the total percentage of spikes that were classified correctly.

First, we selected the channel corresponding to the electrode site $(x, y, z) = (0, 0, 0)$ in each of the test recordings (high, medium and low SNR) and estimated performance for all combinations of spike detectors, architectures and compression bases. This first step illustrated the relative performances of the alternative system setups at different noise levels.

Second, we focused on the generic compression basis, i.e. the *fixed 2* basis and estimated performance in spike sorting and reconstruction at all electrode sites for each test recording, using the ABS detector with architecture 3 and the NEO detector with architecture 1. These detector-architecture combinations were chosen due to their simplicity and their good performances according to the first part. We also included NEO detection with architecture 0 (no compression) as a reference.

Figure 4 illustrates the procedure for spikes from electrode site $(x, y, z) = (0, 0, 0)$ in the high SNR recording for ABS detection, the *fixed 2* basis and architecture 3. The spikes were aligned at their ABS detection threshold, aligned at the absolute maximum, compressed with an 8 dimensional *fixed 2* basis, reconstructed and sorted with PCA and K-means.

2.8 Computational Complexity on the Implant

We used similar complexity measures as those employed in [15], where one operation was defined as a one-bit addition. Subtraction was assumed to involve the same number of operations as addition and multiplication and division were assumed to involve ten times as many operations as addition. We assumed a wordlength of 10 bits and a sampling rate of 25 kHz, both of which are within reasonable limits for successful spike detection and spike sorting [5].

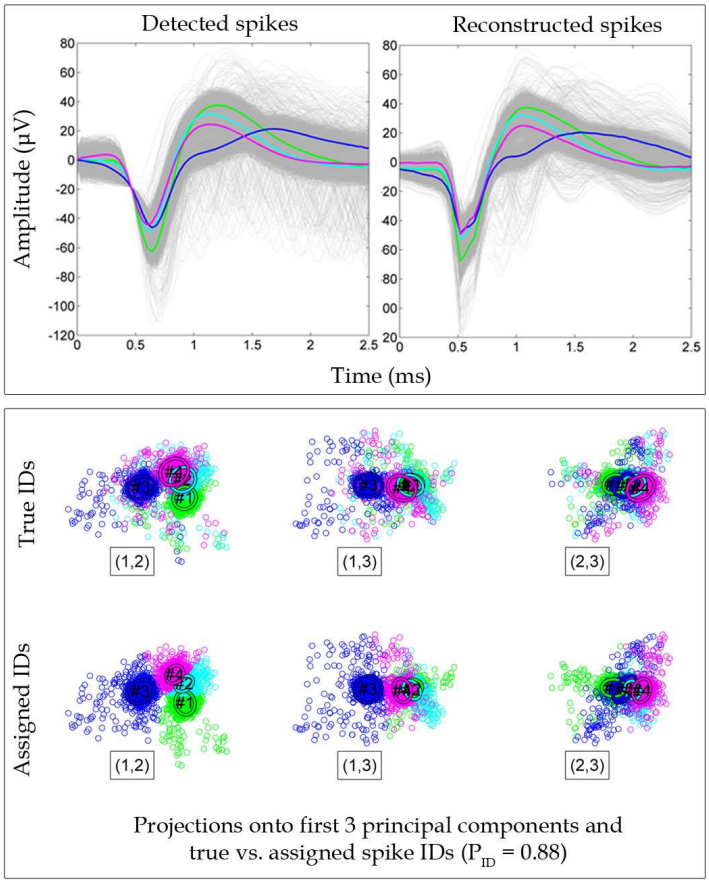


Figure 4: The upper panel shows detected and reconstructed spikes using the ABS detector and compression with the *fixed 2* basis (8 dimensions) and architecture 3 (high SNR, $(x, y, z) = (0, 0, 0)$). Mean spike waveforms are color coded according to their neurons of origin. The lower panel shows the projections of reconstructed spikes onto the first three principal components (marked (1,2), (1,3) and (2,3)). In the upper and lower rows, spikes in the PCA feature space are color coded according to their true and assigned identities, respectively. Clustering was performed with K-means. The overall spike sorting accuracy in this case was $P_{ID} = 0.88$.

Computational complexity was only considered for the cases studied in the second part of the performance estimation (see previous section).

3 Results and Discussion

3.1 First Part: All Architectures, Detectors and Compression Bases

Figure 5 shows spike reconstruction and sorting accuracies as functions of the number of compression basis dimensions for all cases studied at high, medium and low SNR (first step of performance estimation procedure, see Section 2.7) for the electrode position $(x, y, z) = (0, 0, 0)$.

As expected, all the best performances for all architectures were obtained at high SNR, both in terms of spike reconsruction accuracy and spike sorting accuracy. For a given combination of a spike detector and a compression basis, the spike reconstruction accuracy did not vary significantly between the architectures, but was noticeably dependent on SNR and the number of compression basis dimensions included. As expected, the *optimal* and *fixed 1* bases consistently provided the highest reconstruction accuracy, those being directly or indirectly mathematically related to the spike waveforms being compressed. Interestingly, the *fixed 2* basis (generic compression basis) provided only slightly lower performance than the *optimal* and *fixed 1* bases, indicating that spike compression with a fixed generic basis was indeed a feasible alternative. The *Haar* basis consistently performed the worst in terms of spike reconstruction. This is not surprising since no measures were taken to select the most significant Haar transform coefficients, but instead, the first coefficients were selected. This is generally not optimal since the *Haar* basis waveforms are localized in both time and frequency.

At high SNR, the spike sorting performance of architecture 1 with NEO detection gradually became similar to that of the reference architecture, architecture 0. This applied to all compression bases. However, when employing ABS detection with architecture 1, spike sorting performance was significantly lower than for the reference case for all compression bases. This difference is explained by the absence of the spike alignment block in architecture 1 and indicates that the spike detection jitter introduced by NEO detection does not negatively influence spike waveform compression or sorting, but rather directly provides an alignment that is beneficial in terms of both tasks. As noise level increased, the difference in performances between ABS and NEO detection with architecture 1 became less significant. Also at low SNR, spike sorting accuracy for architecture 1 generally fell behind compared to architecture 0.

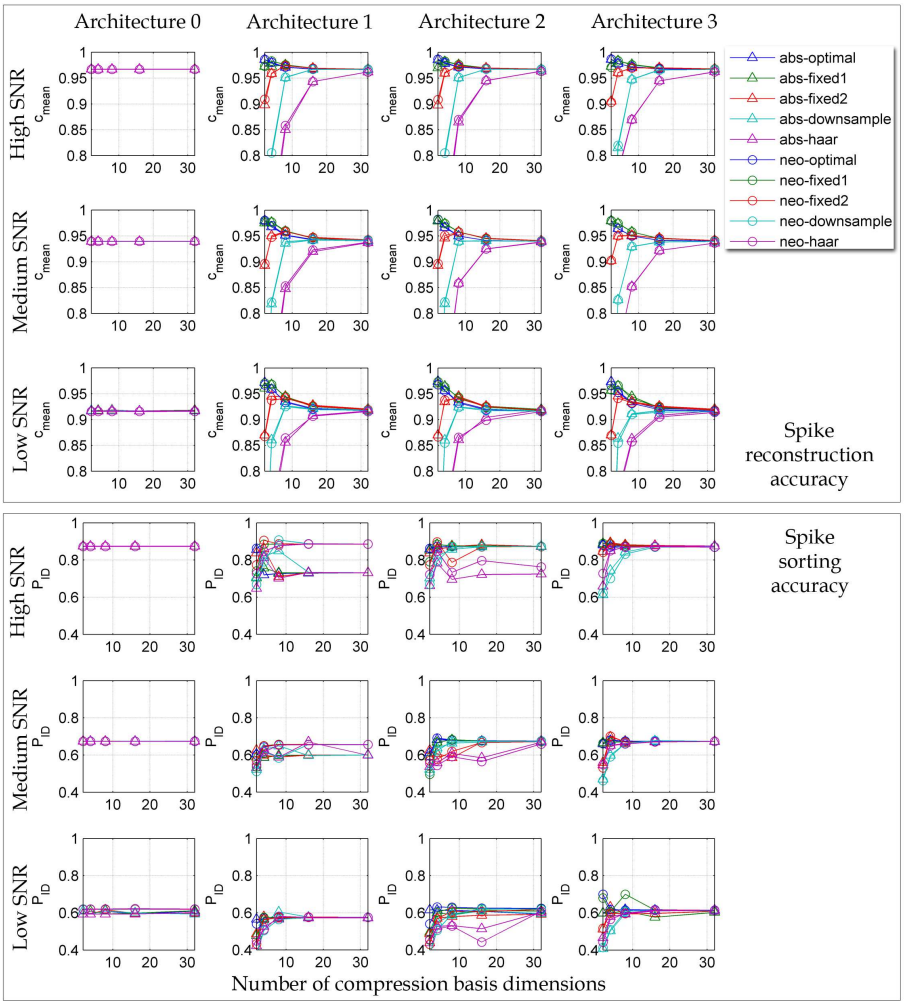


Figure 5: Spike sorting (upper panel) and spike reconstruction (lower panel) as functions of the number of compression basis dimensions at high, medium and low signal-to-noise ratio (SNR). Each column corresponds to each architecture studied (architecture 0 to architecture 3) and each subfigure shows the accuracy for each combination of spike detectors and compression basis.

Architecture 2 provided similar spike sorting performance for both spike detectors and all compression bases, with the exception of the *Haar* basis con-

sistently providing the lowest performance and NEO detection with the *fixed 2* basis converging at a somewhat higher number of compression basis dimensions than the other cases. This is explained by the suboptimal selection of compression coefficients for the *Haar* basis. This relative performance applied at all noise levels.

For architecture 3, the relative spike sorting performances between the configurations did not vary noticeably with SNR. For a given compression basis, the selection of a spike detector did not seem to influence the performance. This is not surprising, since the main difference between the detectors from the point of view of spike compression/reconstruction and spike sorting is the alignment, which is removed by the pre-compression alignment step in architecture The *downsampling* basis consistently provided the slowest converging performance.

3.2 Second Part: NEO and Fixed 2 with Architecture 1 and ABS and Fixed 2 with Architecture 3

Figure 6 shows the distributions of accuracies in spike reconstruction and spike sorting for NEO detection with architecture 0 (no compression, reference case), NEO detection with architecture 1 and compression with the *fixed 2* basis and ABS detection with architecture 3 and compression with the *fixed 2* basis at high, medium and low SNR. Significant difference ($p = 0.05$) between cases is indicated with stars and brackets.

In general, an decreased SNR led to a decrease in spike reconstruction accuracy, the decrease being the least significant when spikes were compressed with at least four dimensions. Thus, the benefit of performing compression also increased as SNR decreased. This is due to the noise reduction introduced by the compression. Note however, that the spike detection performance generally decreases with decreased SNR [15, 5] and this is not an indication that low SNR is beneficial – but rather that given a low SNR, compression is an efficient way of reducing noise in the detected spike waveforms.

As expected, increasing the number of compression basis dimensions increased the reconstruction accuracy for both compression architectures. Except at medium SNR and four compression basis dimensions, no significant difference was seen in the reconstruction accuracy for the compression architectures. In the single exceptional case, although significant, the difference was noticably small. Both compression architectures required at least four compression basis dimensions for the spike reconstruction accuracy to be equal to or higher than that of the reference architecture.

Spike sorting accuracy generally decreased with a decreased SNR and tended to decrease with a decreased number of compression basis dimensions

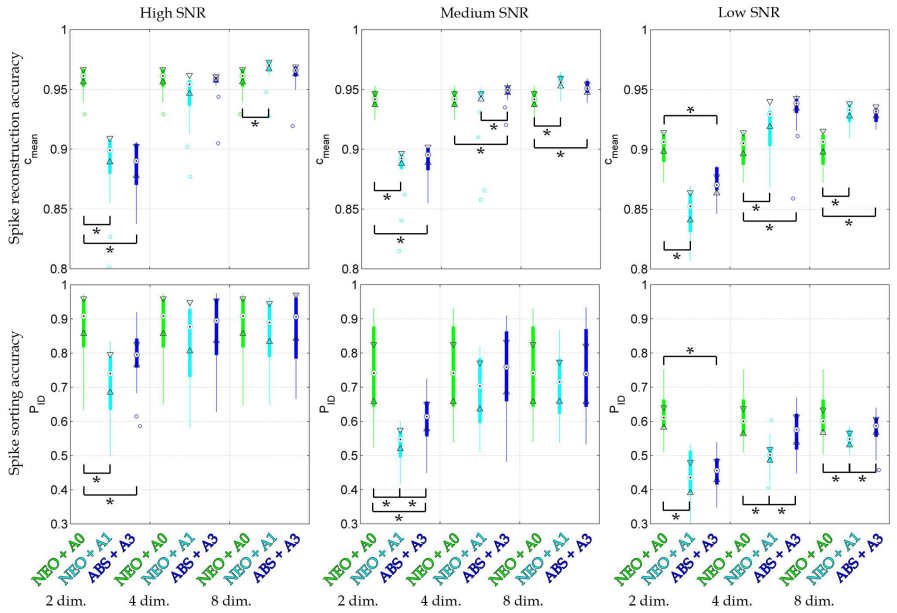


Figure 6: Spike reconstruction accuracy (upper row) and spike sorting accuracy (lower row) for NEO detection with architecture 0 (no compression, reference case), NEO detection with architecture 1 and compression with the *fixed 2* basis and ABS detection with architecture 3 and compression with the *fixed 2* basis at high, medium and low SNR. The distributions describe the performance across all nineteen electrode sites in each case. Median comparison intervals ($p = 0.05$) are marked with triangles and significantly different cases are marked with a star and a bracket.

employed. When two compression basis dimensions were used, both compression architectures provided spike sorting accuracy that was significantly lower than that of the reference architecture. When at least four compression basis dimensions were used at high and medium SNR, there was no significant difference in spike sorting performance between any of the architectures. At low SNR, the spike sorting accuracy of NEO with architecture 1 was the significantly lowest but no significant difference was seen between the performances of the reference architecture and ABS with architecture 3.

The computational complexity at the implant for NEO with architecture 1 and ABS with architecture 3 was 5.5 MOPS/spike/dimension and 0.51 MOPS/spike/dimension respectively. The large (factor 10) difference between

the complexities was due to the increased complexity introduced by performing spike detection with NEO.

4 Conclusions

In this paper, we have studied how various combinations of spike detectors, compression architectures and compression bases influence the performance in spike sorting and spike waveform reconstruction at various signal-to-noise ratios, compared to the case where uncompressed spike waveforms are transmitted and analyzed. Due to the need for minimizing the computational burden on the implant, we have focused on non-adaptive implant designs, i.e. designs where compression is performed with fixed compression bases. This eliminates the need for finding and maintaining an optimal compression basis on the implant.

We have shown that compression with a fixed generic compression basis, obtained by performing singular value decomposition on a set of empirically found mean spike waveforms, is possible, given the appropriate system architecture. The basis we used to show this (*fixed 2*) was obtained from extracellular recordings in the cat cerebellum and is thus entirely independent on the test recordings used in this paper (synthetic multichannel recordings using compressed models of CA1 pyramidal neurons).

Based on the comparison of all combinations of detectors, architectures and bases (part 1), we selected NEO detection with architecture 1 and ABS detection with architecture 3 as feasible candidates for implementation. Both candidate designs provided similar performance in both spike sorting and spike reconstruction. However, the complexity at the implant for ABS with architecture 3 was about 10 times lower than that for NEO with architecture 1, thus making that the most feasible alternative in terms of spike sorting and reconstruction accuracy. A possible argument for selecting NEO for spike detection would be it's superior performance at low SNR.

To summarize, for a future hardware implementation of a wireless BMI, we propose a spike compression architecture that consists of absolute value threshold detection, spike alignment at the implant and compression with a fixed basis that is derived from a large assembly of empirically found spike waveforms. Such a configuration has been shown to provide spike reconstruction and sorting accuracies that differ insignificantly from those obtained when no compression is performed, given that at least four compression basis coefficients are transmitted per detected spike waveform.

Transmitting four compression coefficients per detected spike waveform and assuming 10 bits per sample, this results in a data rate of 40 bits per trans-

mitted spike waveform, or a factor of 16 times less than when transmitting the uncompressed spike waveforms (assuming 25 kHz sampling rate and 2.5 ms spike duration). In order to simplify these comparisons, we do not consider overhead data such as timestamps and channel IDs. Assuming a mean of four neurons per recording channel and a mean firing rate of 10 spikes per second per neuron, this corresponds to a mean total data rate of 1.6kbps. Assuming the same maximum channel capacity as in the example taken in the introduction (1 Mbps), this would allow the transmission of spike data from 625 recording channels, or a factor of 156 times more than when transmitting the raw neural data (four channels maximum).

Acknowledgments

This work was supported by a Linnaeus Grant from the Swedish Research Council (no. 60012701), a grant from the Knut and Alice Wallenberg Foundation (no. 2004.0119) and the Medical and Engineering Faculties at Lund University.

References

[1] M. a. Lebedev and M. a. L. Nicolelis, "Brain-machine interfaces: past, present and future.," *Trends in neurosciences*, vol. 29, pp. 536–46, Sept. 2006.

[2] P. Mitra and H. Bokil, *Observed Brain Dynamics*. Oxford University Press, USA, 2008.

[3] M. Abeles and M. Goldstein, "Multispike train analysis," *Proceedings of the IEEE*, vol. 65, no. 5, pp. 762–773, 1977.

[4] A. F. Molisch, *Wireless Communications*. John Wiley & Sons, 2nd ed., 2012.

[5] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Minimizing data transfer with sustained performance in wireless brain-machine interfaces.," *Journal of neural engineering*, vol. 9, p. 036005, June 2012.

[6] P. T. Thorbergsson, H. Jorntell, F. Bengtsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Spike library based simulator for extracellular single unit neuronal signals.," *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology*

Society. IEEE Engineering in Medicine and Biology Society. Conference, vol. 2009, pp. 6998–7001, Jan. 2009.

[7] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Statistical modelling of spike libraries for simulation of extracellular recordings in the cerebellum.," *Conference Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society*, vol. 2010, pp. 4250–4253, 2010.

[8] M. S. Fee, P. P. Mitra, and D. Kleinfeld, "Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-Gaussian variability.," *Journal of neuroscience methods*, vol. 69, pp. 175–88, Nov. 1996.

[9] P. T. Thorbergsson, M. Garwicz, J. Schouenborg, and A. J. Johansson, "Computationally efficient simulation of extracellular recordings with multielectrode arrays," *Journal of Neuroscience Methods*, vol. null, Aug. 2012.

[10] S. F. Lempka, M. D. Johnson, M. a. Moffitt, K. J. Otto, D. R. Kipke, and C. C. McIntyre, "Theoretical analysis of intracortical microelectrode recordings.," *Journal of neural engineering*, vol. 8, p. 045006, Aug. 2011.

[11] G. Buzsáki, "Large-scale recording of neuronal ensembles.," *Nature neuroscience*, vol. 7, pp. 446–51, May 2004.

[12] D. Heeger, "Poisson model of spike generation," *Handout, University of Standford*, pp. 1–13, 2000.

[13] K. H. Kim and S. J. Kim, "Neural spike sorting under nearly 0-dB signal-to-noise ratio using nonlinear energy operator and artificial neural-network classifier.," *IEEE transactions on bio-medical engineering*, vol. 47, pp. 1406–11, Oct. 2000.

[14] R. Q. Quiroga, Z. Nadasdy, and Y. Ben-Shaul, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering.," *Neural computation*, vol. 16, pp. 1661–87, Aug. 2004.

[15] S. Gibson, J. W. Judy, and D. Marković, "Technology-aware algorithm design for neural spike detection, feature extraction, and dimensionality reduction.," *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 18, pp. 469–78, Oct. 2010.

[16] I. Obeid and P. D. Wolf, "Evaluation of spike-detection algorithms for a brain-machine interface application.," *IEEE transactions on bio-medical engineering*, vol. 51, pp. 905–11, June 2004.

[17] Z. Charbiwala, V. Karkare, S. Gibson, D. Markovic, and M. B. Srivastava, "Compressive Sensing of Neural Action Potentials Using a Learned Union of Supports," *2011 International Conference on Body Sensor Networks*, pp. 53–58, May 2011.

[18] S. Mukhopadhyay and G. C. Ray, "A new interpretation of nonlinear energy operator and its efficacy in spike detection.," *IEEE transactions on bio-medical engineering*, vol. 45, pp. 180–7, Feb. 1998.

[19] S. K. Mitra, *Digital Signal Processing: A Computer-Based Approach*. McGraw-Hill/Irwin, 2001.

[20] M. S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials.," *Network (Bristol, England)*, vol. 9, pp. R53–78, Nov. 1998.

[21] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. New York, NY: Wiley, 2nd ed., 2001.

[22] A. Pavlov, V. a. Makarov, I. Makarova, and F. Panetsos, "Sorting of neural spikes: When wavelet based methods outperform principal component analysis," *Natural Computing*, vol. 6, pp. 269–281, Aug. 2007.

[23] A. Jackson and E. E. Fetz, "Compact movable microwire array for long-term chronic unit recording in cerebral cortex of primates.," *Journal of neurophysiology*, vol. 98, pp. 3109–18, Nov. 2007.