AN INFORMATION-THEORETIC ANALYSIS OF SPIKE PROCESSING IN A NEUROPROSTHETIC MODEL

by

Deborah S. Won

Department of Biomedical Engineering
Duke University

Date: _____

Approved:

_____
Patrick D. Wolf, Supervisor

_____
Craig Henriquez

_____
Loren Nolte

_____
Paul Tiesinga

_____
Dennis A. Turner

Dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Department of Biomedical Engineering
in the Graduate School of
Duke University

2007

ABSTRACT

AN INFORMATION-THEORETIC ANALYSIS OF SPIKE PROCESSING IN A NEUROPROSTHETIC MODEL

by

Deborah S. Won

Department of Biomedical Engineering
Duke University

Date: _____

Approved:

_____
Patrick D. Wolf, Supervisor

_____
Craig Henriquez

_____
Loren Nolte

_____
Paul Tiesinga

_____
Dennis A. Turner

An abstract of a dissertation submitted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy
in the Department of Biomedical Engineering
in the Graduate School of
Duke University

2007

Abstract

Brain-computer interfaces (BCI) are being developed which provide volitional control of a motor prosthesis to patients who suffer from motor-debilitating diseases and conditions. Neuroprosthetic devices are controlled by activity from the patient's brain and bypass damaged parts of the spinal cord or peripheral nervous system. Spike sorting is a technologically expensive component of the signal processing chain required to interpret population spike activity acquired in a BCI. No systematic analysis of the need for spike sorting has been carried out, and little is known about the effects of spike sorting error on the ability of a BCI to decode intended motor commands. We developed a theoretical framework and computational methodology to examine the effects of spike processing on the information available to a BCI decoder.

Shannon information theory was applied to simulated neural data. Results demonstrated that reported amounts of spike sorting error reduce mutual information (MI) significantly in single-unit spike trains. We also studied how much information is available in a cluster of pooled signals. Indirect information analysis revealed the conditions under which pooled multi-unit signals can maintain the MI that is available in the corresponding sorted signals and how the information loss grows with dissimilarity of MI among the pooled responses. To examine these properties of information loss in non-sorted population activity within the context of a BCI, we simulated responses of 4 neurons with the commonly observed and exploited cosine-tuning property and with varying levels of spike error. Tolerances of angular tuning differences and spike sorting error were given for MI loss due to pooling.

These analyses revealed the degree to which mutual information loss due to pooling spike activity depended upon differences in tuning between pooled neurons and

the amount of spike error introduced by sorting. The theoretical framework and results presented in this dissertation will aid BCI system designers to make informed decisions based on a systematic understanding of the tradeoffs between a system with and without spike sorting.

Acknowledgements

"... [L]et him who boasts boast in this, that he understands and knows me, that I am the LORD who practices steadfast love, justice, and righteousness in the earth. For in these things I delight, declares the LORD." I can indeed boast in the steadfast love of the LORD, who has brought joy and richness to my life these past seven years through many professors, engineers, scientists, peers, friends, and family.

I have had the privilege of learning from and being supported by my committee members. Many thanks to Dr. Henriquez for discussions and insight, Prof. Nolte for his welcoming attitude and kind support, Dr. Turner for his enthusiasm and encouragement, especially in setting up animal studies in our lab, and Dr. Tiesinga for the time, feedback, and thought-provoking guidance he has provided for me. A special thanks to Dr. Wolf for sharing from his wealth of knowledge with me. His engineering skills have set a high standard for all of his graduate students, and he has demonstrated much patience in dealing with even the most engineering-challenged, such as myself. Thanks to him, I believe my ability to do science is improved, and I hope by this time, some of his writing skills have rubbed off on me. Thank you all for not giving up on me.

Coming in to the same office and lab day after day could never have been so enjoyable had it not been for stimulating conversations with the fun labmates with whom I have been able to work and laugh and play. Thanks to Chrissie Merdes, Iyad Obeid, Amy Goodman, Kit Au-Yeung, Liane Teplitsky, Ninita Brown, Chad Bossetti, Tom Jochum, and Michael Rizk. And what would life have been (other than chaotic and boring all at once) without Ellen Dixon-Tulloch, Ned Daniely, and Steve Callender? Thank you so much for keeping this lab running! Also a big thanks to other scientists and engineers who graciously gave their time and advice to contribute

to my graduate studies – thanks to Gary Lehew, Mike Wiest, Eric Thompson, David Krupa, Lucas Santos, Roy Crist, Mark Laubach, Joey O'Doherty, George Hugh, Jose Carmena, Erin Phelps, Misha Lebedev, Samar Mehta, Dan Hill, Garret Stanley, and Jean-Marc Fellous.

I could not say enough about the family and friends who have made my heart overflow with gladness and thanksgiving. I'd like to first thank my parents, who have continually provided and cared for me. They have let me benefit immensely from their hard work and allowed me every opportunity for a wonderful education. Much love and gratitude also to Carol, who has always been my most supportive cheerleader; Sandra, who has always been there to provide wise counsel and empathize with PhD research woes; Eugene, Dave, Elisabeth, Audrey, and Emily; Joyce, Will, Scott, Jenny, and all my friends from Crossroads and ICF. Thanks, especially, to Alex, the LaTeX guru, who has been there to laugh with me (and at me), to appreciate all the good gifts with which God blesses us, to be the encourager I never deserved, and to be the aroma of Christ for me. Thank God for letting His light shine through all these friends for me.

For my parents,

Joha Hyun and Jeong Hee Won

Oh sing to the LORD a new song;
sing to the LORD, all the earth!
Sing to the LORD, bless his name;
tell of his salvation from day to day.
Declare his glory among the nations,
his marvelous works among all the peoples!

For great is the LORD, and greatly to be praised;
he is to be feared above all gods.
For all the gods of the peoples are worthless idols,
but the LORD made the heavens.
Splendor and majesty are before him;
strength and beauty are in his sanctuary.

Psalm 96:1-6

Contents

List of Figures

List of Tables

Chapter 1

Background and significance

Motor dysfunction can result when the motor regions in the brain are still functional but communication between the brain and the muscles is disrupted. This occurs in conditions such as amyotrophic lateral sclerosis (ALS), brain stem stroke, muscular dystrophy, cerebral palsy, axonal neuropathies, and spinal cord injury (SCI) [2]. These motor-debilitating conditions together affect over 700,000 people in the U.S. alone [3]. An estimated 2 million people around the world suffer from SCI [4]. There are approximately 200,000 individuals in the U.S. suffering spinal cord injuries, with 11,000 new incidences of SCI per year [5, 6]. Fifty-two percent of these individuals are diagnosed with paraplegia while 47% are diagnosed with quadriplegia [4]. SCI, ALS, and similar diseases can lead to paralysis or even the locked-in syndrome, or the inability to interact with the outside world. The prevalence of the locked-in syndrome and paraplegia is not as high as that of the instigating diseases; yet, the impact of such motor dysfunction is profound. Motor debilitation can impact every aspect of life, disabling an individual from walking, feeding, bathing, dressing themselves, from adequately expressing themselves, and in general limiting their independence [7].

Motor prostheses are being developed which can be controlled by neural activity from the brain and bypass damaged parts of the spinal cord or peripheral nervous system. The ability to record activity from populations of neurons simultaneously has enabled the development of prosthetic devices controlled by patients' thoughts, a technology which has come to be known by many names – neuromotor prostheses (NMP), brain-machine interfaces (BMI), brain-computer interface (BCI), cortically-controlled prosthetics, or neurorobotics. NMPs will provide motor capabilities to

patients who otherwise would be unable to execute motor commands of their own volition and free them to interact with the world.

Many neuroscientists and biomedical engineers are working on these BCIs with approaches that differ in detail but are generally similar in mechanism: electrodes are implanted in the motor areas of the brain; amplifying and filtering hardware acquire and process extracellular signals; unit spike activity is discriminated from these signals; and a computational algorithm decodes intended motor output from these spike trains. There are non-invasive approaches to BCIs [8, 9], but here we focus on the cortically-controlled systems that utilize spike activity detected on implanted microelectrodes.

Implantation, acquisition, and processing capabilities have already been demonstrated for BCI systems. Extracellular signals from tens and even hundreds of neurons have been recorded simultaneously from multiple areas of primate cortex while the animal performs tracking tasks. Even with this relatively small sample of the estimated 10^{12} neurons in the brain, recordings from motor-related areas in primates have been exploited to predict 3-D limb position with above-chance accuracy (coefficient of determination R^2 ranging from .2-.8) [10, 11, 12]. The feasibility of controlling a 4-degree-of-freedom output from neural signals acquired through intracortical electrodes in a human subject has also been demonstrated [13, 14]. However, these demonstrations of feasibility leave the field years of quantitative and analytical research away from a fully-portable clinically-proven brain-controlled prosthetic device.

The available data and analysis has been insufficient thus far to inform a systematic design specification process for BCIs, and this data is very difficult to collect. Development would be aided by a better understanding of the information available in the population of extracellular signals detectable by microelectrode arrays and the effect of processing in a BCI on that information.

1.1 The spike sorting problem

Remarkable progress has been made in BCI technology, thanks in large part to landmark advances in neural recording technology. BCI research is not lacking in motivation; noteworthy results are the result of diligent engineering; and assumptions and design decisions have not been devoid of intuition or good decision-making. Yet, as with development of most technology, especially endeavors which involve the interface between human biology and devices, there is exploration involved. Improvements can only be made once pioneering work has laid the foundation; many areas of improvement come into clearer perspective only in hindsight. Some aspects can be made lesser of a priority until other developments have been made. In the case of BCI technology, it is only after great efforts have already forged the way that we can now come and make targeted explorations into various pieces of a BCI system. We focus here on the piece of that puzzle called spike sorting.

1.1.1 Purpose of spike sorting

In a BCI, the spike activity of many individual neurons are recorded simultaneously. Arrays of microelectrodes are placed into the brain in order to detect extracellular voltages produced by nearby neurons. The signal detected by an electrode placed in the extracellular space of a brain is due to a summation of currents from neighboring neurons. Depending on the distances from the electrode, action potentials from multiple neurons may be observable in the extracellular voltage. Spike sorting is the process of discriminating which action potential waveforms originate from the same neuron.

Action potential waveforms from different neurons will vary in amplitude or shape, and these differences can be exploited to label each detected action potential with its originating neuron. To illustrate the problem, a signal recorded from an extracellular

micro-electrode implanted in the rat motor cortex is shown in Figure 1.1. Tick marks above the waveform indicate detections of an action potential. The shape of the action potential depends on which neuron gave rise to it. The goal of spike sorting is to label each action potential with the neuron which produced it. Action potentials from 2 neurons were detected in this recording. The two neurons are distinguished by the yellow versus green tick marks.

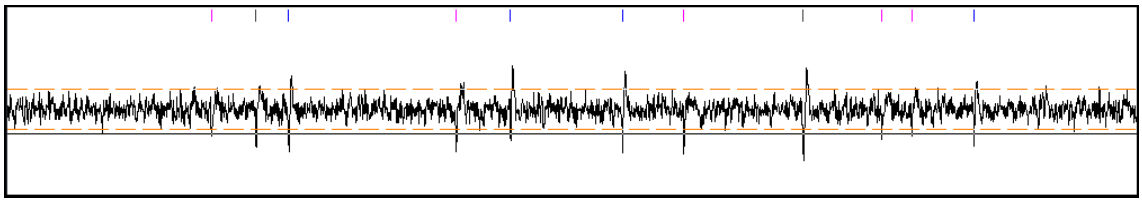


Figure 1.1: Example of a spike sorted waveform. An extracellular waveform acquired from an electrode near 2 neurons is displayed. The neuron's action potentials are distinguished from one another by shape. The goal of spike sorting is to correctly identify each action potential with the source neuron. A commercial software program, Plexon's Off-line Sorter, was used to process data acquired from a rat's motor cortex according to IACUC-approved protocol #A051-05-02. Principal component analysis (PCA) with K-means clustering was used to identify 2 neurons. The identity of the neurons are indicated by the differently colored tick marks above the waveform.

Microelectrode technology was a major breakthrough for studying and understanding ensemble encoding in the brain. Implantation of arrays of microelectrodes in the brain has become a popular and useful method for recording activity from populations of neurons simultaneously [15]. Chronic implants allow observation of long-term behavior, particularly useful for studying learning and memory. Now, there is also the possibility to record neural activity which encode intended motor output.

When the simultaneous spike activity of multiple neurons is desired, spike sorting is essential because microelectrodes detect spike activity from 2-3 neurons on average [15]. However, we do not know whether BCI systems rely heavily upon the activity of multiple individual neurons and whether they will operate significantly more effectively when sorted activity from neurons is used rather than when spike sorting is not used. Until neural encoding of motor control is better understood, we cannot

definitively say whether single-unit activity is essential to a BCI. Under the circumstances in which BCI developers found themselves, with an incomplete knowledge of how best to decode motor information, it was only reasonable to try to obtain spike activity with as much precision as possible.

However, spike sorting is proving to be a major challenge to implement in BCIs, as will be described further in section 1.1.3. As BCI technology advances, and the system moves toward becoming a portable and practical device for patients, we cannot afford to leave spike sorting in the system if it is not crucial to the effectiveness of a BCI. The aggregate activity of clusters of neurons may prove just as effective, and that is the question we set out here to begin answering. The implications could in turn improve the prospects of a BCI being implemented as a practical and effective system, since spike sorting adds a substantial computational burden and must be supervised.

1.1.2 Current methodology

Spike sorting has been a major component of the multi-electrode array BCI since its inception [16]. Researchers have devoted much time and effort to its development since the 1960s [17], and now neurophysiologists must devote pain-staking attention to perform spike sorting on multi-electrode recordings. Spike sorting generally consists of three stages (see fig. 1.2). The first stage is to extract the action potential, or spike waveform, with some threshold detection scheme. The detector waits for the amplitude of the extracellular signal to cross a pre-determined threshold, and then extracts the samples around this threshold-crossing, roughly 1 ms in duration. The second stage is feature extraction. A feature of the spike waveform is extracted, commonly in a lower-dimensional space, which can be used to identify distinct neurons. The feature might be the amplitude of every sample in the waveform for template

matching, which provides no reduction in dimensionality, or the highest-ranking principal components, for example.

The final step is classifying the action potentials based on these features. The classification of action potentials detected on the same channel can be carried out with a range of methods including principal component analysis (PCA), time-amplitude window template matching, and wavelet transforms [18, 19, 20, 21, 22]. Regardless of the specific method, the general idea is to map each detected waveform onto the feature space and determine the identity of the spike based on its location in the feature space. The most commonly-used method is PCA, in which each detected spike is projected onto the first 2 or 3 principal components [18, 21]. The waveforms belonging to the same neuron are expected to share enough similarities with each other that their projections would cluster in roughly the same region of principal component space, and have distinct enough features from those belonging to a different neuron that their projections would cluster in a separate region of the PC space. Based on training data acquired previously, cluster boundaries are drawn manually or set automatically using a clustering method, such as K-means clustering, in which statistical methods are used to determine the center of the clusters and circular boundaries a certain distance from the center, or a Bayesian maximum likelihood rule [19].

Each of these methods are dependent on calibration using training data and on parameters set by the user. The user can set a threshold or use an auto-set threshold, which is based on the absolute voltages of previously recorded segments. Every time discriminated spike trains are to be acquired, the experimentalist is required to monitor action potential waveforms on every channel and set appropriate thresholds and cluster boundaries or ensure that old ones are still valid according to his or her criteria. The inter-spike interval histogram, which give the number of times two consecutively occurring spikes in a cluster occur within a certain time interval, can aid in

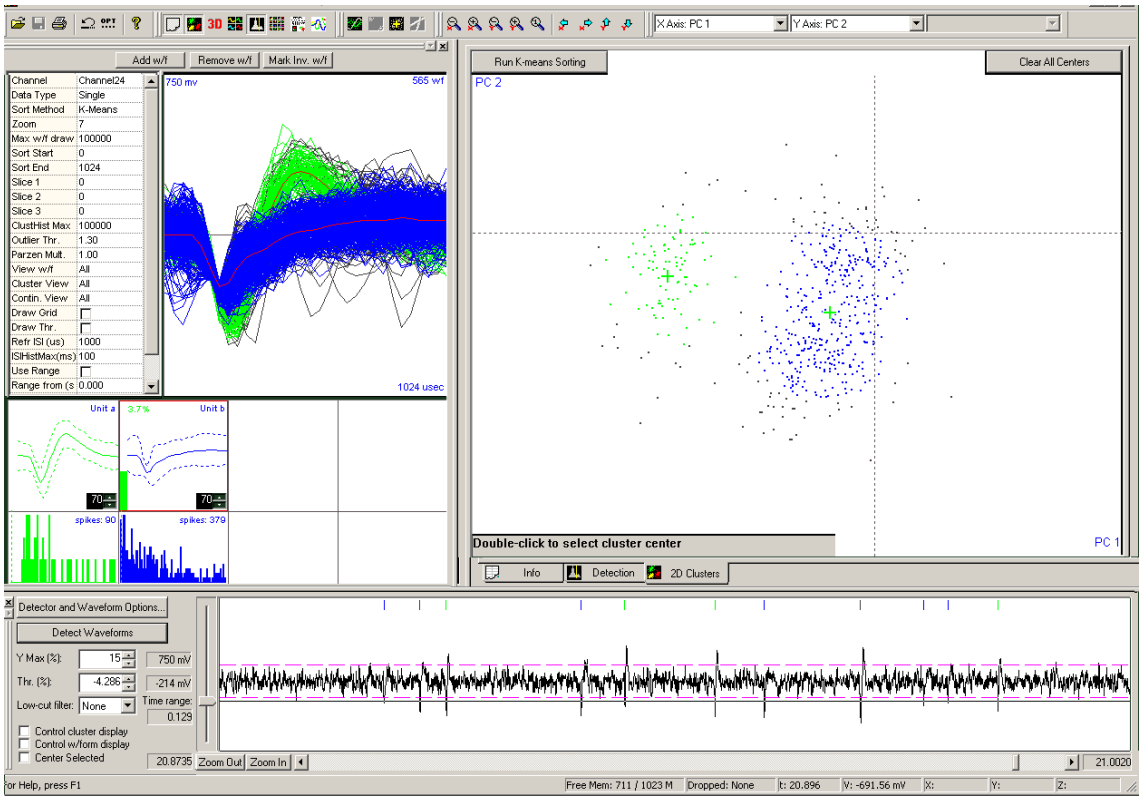


Figure 1.2: Spike sorting first requires that action potentials be detected from the recorded extracellular signal, which is the white signal shown in the lower panel. The solid black line at the lower end of the signal indicates the level at which the detection threshold was set. The tick marks in the same panel directly above the extracellular waveform indicate negative-slope threshold crossings. A 1-ms waveform around this threshold crossing is extracted to yield detected action potential waveforms, as seen in the top left panel. The action potential waveforms are aligned on a pre-determined point (e.g., the global maximum after threshold crossing), and PCA is performed on the waveforms. PCA allows the spike waveform to be reduced to a lower dimensional space, in which action potentials will cluster according to similarity in shape. The projections of the waveforms onto this lower-dimensional space is seen in the top right panel. The user can select different methods to define cluster boundaries for separate neurons. In this case, the green dots are from waveforms that are classified as belonging to unit a; the blue dots are all classified as unit b. The tick marks above the extracellular waveform are also color coded according to whether the action potential was classified as unit a or b.

adjusting the boundaries, since spikes from the same neuron should not occur within the refractory period. However, the adjustment of cluster boundaries often relies heavily upon visual appearance. When the cluster boundaries are being determined, the user can adjust cluster boundaries and designate individual waveforms that fall within those boundaries as artifact or noise waveforms to be discarded. A cluster it-

self can also be designated as an artifact unit and likewise be discarded. The cluster boundary settings require frequent attention by an experienced user. The detection step may very well introduce error that can cause different effects than classification error. While classification error can introduce spikes that are from correlated spike trains, detection error is more random, and therefore, could have more severe effects than misclassification, but we address this issue in Sec. 3.3.2.

In order to have the capability to determine the identity of each action potential, the extracellular signals must be processed to retain the essential features of the waveform. Most of the studies that have advanced BCI technology thus far and are furthering BCI development have been performed in primate subjects that are tethered to large commercial electronics to carry out spike sorting. The extracellular signals detected on the implanted microwires are buffered, amplified, and then typically processed by Plexon, Inc.'s Multichannel Acquisition Processor (MAP) or a comparable processing system. The MAP system must filter and sample the signals well enough to perform spike sorting on the multi-unit signal on each channel. The acquired signals are filtered through six low-pass poles ($f_c = 8kHz$) and four high-pass poles ($f_c = 80 - 250Hz$), digitized at 32k-samples/s, and discriminated using a computationally-intensive, *supervised* hybrid time-amplitude/PCA algorithm [23]. These signals are then processed by a host PC using algorithms that decode intended limb movement which would serve as control inputs to a prosthetic device. This particular system consists of a data acquisition box which is powered off $120V_{ac}$, with dimensions measuring 19"x19"x13", has a pre-amplifier box which requires a 12V supply measuring 12"x5"x3", and a host PC powerful enough to run the on-line Sort Client (comparable to the Dell Optiplex GX620).

1.1.3 Limitations imposed on BCI by spike sorting

Although this and other commercial hardware systems are currently the most effective way to acquire discriminated multi-channel neural population signals, the amplification, filtering, and digitizing it performs will need to be implemented in hardware that does not require the user to be tethered to the hardware; i.e., in a device that is portable and low-power. The inclusion of spike sorting significantly increases the demand on these systems, limiting the number of channels that can be processed and the portability of the system, as described in this section.

An analog front-end and digital telemetry prototyping system was built in the Wolf lab to implement these functions in a portable system [24, 25]. This system and the effort put into developing it demonstrate the high demands placed on this signal conditioning and digitizing hardware. The system did not actually perform spike sorting but conditioned signals for spike sorting with the following capabilities: to provide the gain and filtering needed to emulate the output signal quality of the MAP, buffered waveforms digitized at 12-bit resolution and time-division multiplexed 16 channels for wireless transmission at 31.25k samples/s. Without any implementation of spike sorting, the device still weighed 235g at a size of $5.1cm \times 8.1cm \times 12.4cm$. Sixteen channels of *in vivo* data have been successfully transmitted to a host PC from a 9-m range at an estimated power consumption of 4W. Extrapolating this system to 100s of channels would have required device sizes and power impractical for a neural prosthetic application. Clearly, efficient design specifications are necessary to develop a system that can be used by patients. Using 96 channels where 80 could have sufficed, for instance, unnecessarily would add roughly an additional pair of lithium ion batteries every 3 hours, 235g of weight, and thousands of dollars in IC fabrication costs to the device.

Through several years of dedicated research, demonstrable progress has been made

toward a fully wearable system capable of acquiring the desired hundreds of channels. A 32-channel digitizing headstage [26] has been designed and built to condition signals so that they are suitable for spike sorting. This headstage is compatible with an implantable communications module (ICCM) which can transmit signals via wireless telemetry to an external wearable communications module (WCM). Spike sorting requires the waveform of the action potentials to be saved and transmitted. In order to provide the downstream processors with the data necessary for spike sorting, 32 12-bit sample waveforms are extracted and 8 bits of each sample transmitted every time an action potential is detected on any of the channels, whereas detection of spike waveforms without spike sorting requires just 1 bit for every action potential; essentially, spike sorting increases the amount of transmitted data by 256 times, assuming 8-bit samples are sufficient for spike sorting. With the finite memory of the FPGA and bandwidth, more processing required per channel equates to a smaller capacity for channels. This might be avoided by implementing spike sorting in the ICCM before transmitting. However, implementing spike sorting in implantable hardware would also be a challenge, as mentioned in Section 1.1.2 and further indicated in Chapter 3. Furthermore, eliminating the need for the reproduction of the neural waveforms could reduce the amount of filtering needed, which also might reduce power consumption and silicon area of the signal-conditioning IC. The feasibility of implementing a spike sorter that can perform reliably and without supervision in an implantable system has not yet been demonstrated.

It has been assumed that the motor information acquired in a population of recorded neural activity increases as the number of units acquired increases; in this light, spike sorting would appear to aid BCIs to do the best job of reproducing intended motor output, since it maximizes the number of units acquired in each set of implanted arrays. In one pioneering study on tracking arm movement from

cortically-derived signals, it was claimed that the sorted activity of 200-1000 neurons would be needed to achieve R^2 values of 0.9. The push for increasing the number of units acquired has been encouraged by such claims but without further supporting evidence. Development of BCIs without further investigation of these claims is not only cost-ineffective, but is potentially a hindrance to true progress. The size of major components in a BCI system as well as the run-time of the decoding algorithm increase significantly with the number of units processed. Computational demands would be high for a standard PC to perform the prediction of 3-dimensional hand position from hundreds of spike trains in real-time in a patient's home. Filtering, discriminating, and processing hundreds of channel of neural spike trains can become an unwieldy problem as the desired number of neurons increases.

From the standpoint of developing hardware to meet the exacting demands of a BCI, removing spike sorting from the system requirements would provide great relief and would make BCI hardware development a more tractable problem. Nevertheless, we clearly cannot leave out spike sorting until we know that it would not significantly reduce the effectiveness of the system. Nor should we devote large amounts of resources into implementing spike sorting without better understanding its effects in a BCI. Thus, we aim to explore these effects to help shape our thinking about spike sorting's role in a BCI.

1.2 Historical background of spike sorting

From the literature documenting the pioneering research in spike discrimination, it appears that spike sorting efforts emerged through a logical progression of electrophysiological studies. Electrophysiologists began their studies in the brain with single electrode intracellular recordings to investigate the properties of individual neuron's electrical behavior and signalling mechanisms. In order to study the interactions and

functional associations between adjacent neurons, they then used the same technique to record serially from individual neurons [27]. Aside from being tedious and inefficient, this method of studying behavior of neural populations can produce misleading results, since simultaneous recordings contain state information that serial recordings would not. Efforts to overcome this problem naturally led to recording from individual units simultaneously, as stated by Gerstein in his 1966 publication: "in order to study neuronal interactions, it is necessary to sort such compound data so that the activity of each observed neuron as a function of time is known individually" [28].

Hence, the first multi-unit recordings proceeded from single-unit electrophysiological studies with the goal to observe specifically the behavior of multiple single units recorded simultaneously and the temporal interactions between them [29, 30]. Thus, spike discrimination has been viewed as a task necessary to rectify a problem in extracellular neural recordings. "A quite successful approach [to the analysis of nervous system functions] has been to study simple systems... where one can identify the individual activity of many of the neurons involved. Extension of this approach to more complex systems would be very desirable, but it requires new techniques for monitoring the activity of individual neurons in a large population of simultaneously active cells," [31] namely spike discrimination. At the time that population recording was extended to BCI applications, it would have been reasonable to specify sorting of spike trains as a requirement. When BCIs were first being developed, many questions remained outstanding, as regards encoding of motor information and what aspects of neural activity are crucial to extracting that information. The literature does not document the decision process that led to incorporating spike sorting in BCIs. Questions that remained outstanding included whether the interactions between individual neurons were really encoding information, whether that information was being exploited in the decoding algorithms of BCIs, and whether aggregate activity

of multiple neurons could contain just as much information.

While spike sorting may have been necessary for studies of interaction between individual neurons, system-level studies or applications may not require this burdensome task. Despite the different goals of current high-density recording applications compared to the original goals of electrophysiology, the purported requirements of the acquisition and processing system have remained unchanged.

1.3 Overview of the dissertation

In a BCI, hundreds of microwires are implanted to acquire neural signals which are then processed to produce predictions of desired motor output (namely limb position and/or force). Each microelectrode can detect action potentials from multiple neurons. Spike sorting is the process of separating action potentials from different neurons based on distinguishing features of the extracellular waveform. In current brain-machine interface (BCI) systems, each extracellular signal acquired from an implanted intracortical microelectrode is spike sorted into multiple trains of action potential time stamps, and each sorted spike train purportedly arises from a single neuron or unit. Spike sorting should lead to benefits to warrant its costly burden on electronics and computation in a BCI.

The effect that spike sorting has on information will likely depend on the code that is used by the neurons from which signals are recorded. As evidenced by ongoing debates fueled by a rich history of studies, contending theories on how motor control is encoded in neural signals are not likely to be whittled down to one consensus theory before decisions need to be made on how to acquire and process the neural signals for brain-controlled interface applications. In the meantime, information-theoretic analysis will provide a theoretical understanding for how neural information content can depend on spike sorting under various assumptions about encoding schemes.

The fundamental question we would like to address is what benefit is gained by spike sorting in a neuromotor prosthetic system? Does the increase in information gained by using labelled-line activity over using pooled activity warrant the cost of spike sorting error and the effort and resources needed to implement spike sorting in an implantable BCI device? In this dissertation, we endeavored to begin the process of understanding how spike sorting affects the information obtained in BCIs.

The BCI system was viewed in terms of information processing. From this information-theoretic perspective, we note two major reasons to question the benefit of spike sorting in light of the burden its implementation creates for a BCI: (1) signals that are processed via spike sorting may lose significant information because of classification error; (2) signals that are processed without spike sorting may not lose significant information when waveform identity is lost. In this dissertation, we examine the losses due to these processing steps with information-theoretic analysis. First, we determine the potential for spike sorting error to cause information loss. We then determine what information does remain in a non-sorted multi-unit signal, and compare this information to its corresponding sorted signal. We make further comparisons in the context of cosine-tuning with an additional cluster of units and with varying degrees of spike error.

Owing to the evolution of multi-unit recordings from a history of neurophysiological studies via single unit recordings, the necessity of extracting single-unit signals from multi-unit electrograms has been taken for granted [30, 29]. Promising results of neural computational analysis on spike-discriminated multi-unit data encourage scientists to continue employing spike sorting [32, 10, 11, 12]. However, examination of the impact on information due to sorting error and of the potential of multi-unit signals to carry information revealed that BCIs, as they stand today, can be expected to extract similar amounts of information from non-sorted spike activity as they do

from spike sorted activity in realistic conditions. Nevertheless, we believe these conditions, while possible, are quite restrictive and would be present only unreliably. We describe those analyses in the following chapters. In so doing, we undertake the task of developing our understanding of the effect spike sorting could have on the ability of a BCI to interpret motor commands from neural spike activity.

Chapter 2

A theoretical framework

Spike sorting might be necessary to optimally interpret motor intention from neural activity. However, its implementation is too costly for NMP development to proceed based on this untested assumption. Section 1.1.3 described the obstacles posed by spike sorting on hardware implementation in a BCI. There are other significant costs associated with spike sorting, both with implementing it and with the effect it has on the information obtained, as will be described in chapter 3.

The benefits of spike sorting should be examined to determine whether the extra cost that would be imposed beyond purely detecting spikes is warranted. In this chapter, we establish our approach to this problem and describe the measure we use to analyze the benefits of spike sorting. We describe, in an information-theoretic framework, the processing that takes place from the electrodes to the input of the decoder in a BCI. We view the differences between sorted and non-sorted signals in this framework.

2.1 A theoretical view of spike sorting

BCIs aim to extract and exploit information about some target signal, such as desired limb movement, referred to as θ in this section, from neural population activity ($\mathbf{P}$). There are many steps involved in processing neural signals to obtain predictions of the target signal θ, and every step can incur information loss.

Let us imagine a population of neural signals $\mathbf{P}$ as encoding information about some input signal, $\boldsymbol{\theta}$, as in Fig. 2.1. The signals are processed as they are acquired and conditioned by hardware and passed on to a decoder. In Fig. 2.1, the output of

this processing is generically referred to as $\mathbf{P_{proc}}$. What takes place in a BCI between the brain and the decoder can be very generally conceived as a processing block. We know, according to the well-known data processing inequality, that there can only be information loss, and not gain, between the original population activity and the input of the decoder [33]. This principle states that no amount of signal processing on an original signal $\mathbf{P}$, which carries information about $\boldsymbol{\theta}$ (Fig. 2.1), can add information, as described in Eqn. 2.1 for the system illustrated in Fig. 2.1.

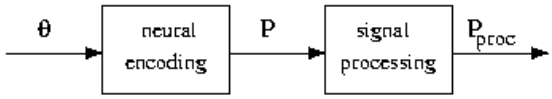


Figure 2.1: The brain encodes some stimulus or target signal θ, producing a population of neural signals $\mathbf{P}$. $\mathbf{P_{proc}}$ represents the neural signals after signal processing.

$$I(\theta; \mathbf{P_{proc}}) \leq I(\theta; \mathbf{P}) \tag{2.1}$$

Fig. 2.2 shows a block diagram depicting the main processing steps in acquiring spike sorted and non-sorted signals in a BCI. Individual neurons may encode some information about $\boldsymbol{\theta}$, and each produces an extracellular current, i_k. The main information in this set of extracellular currents, i.e. $\{i_1, i_2, \ldots i_n\}$, is believed to be represented in the occurrences of the neurons' action potentials [34, 35] and thus represented by spike trains of a binary code.

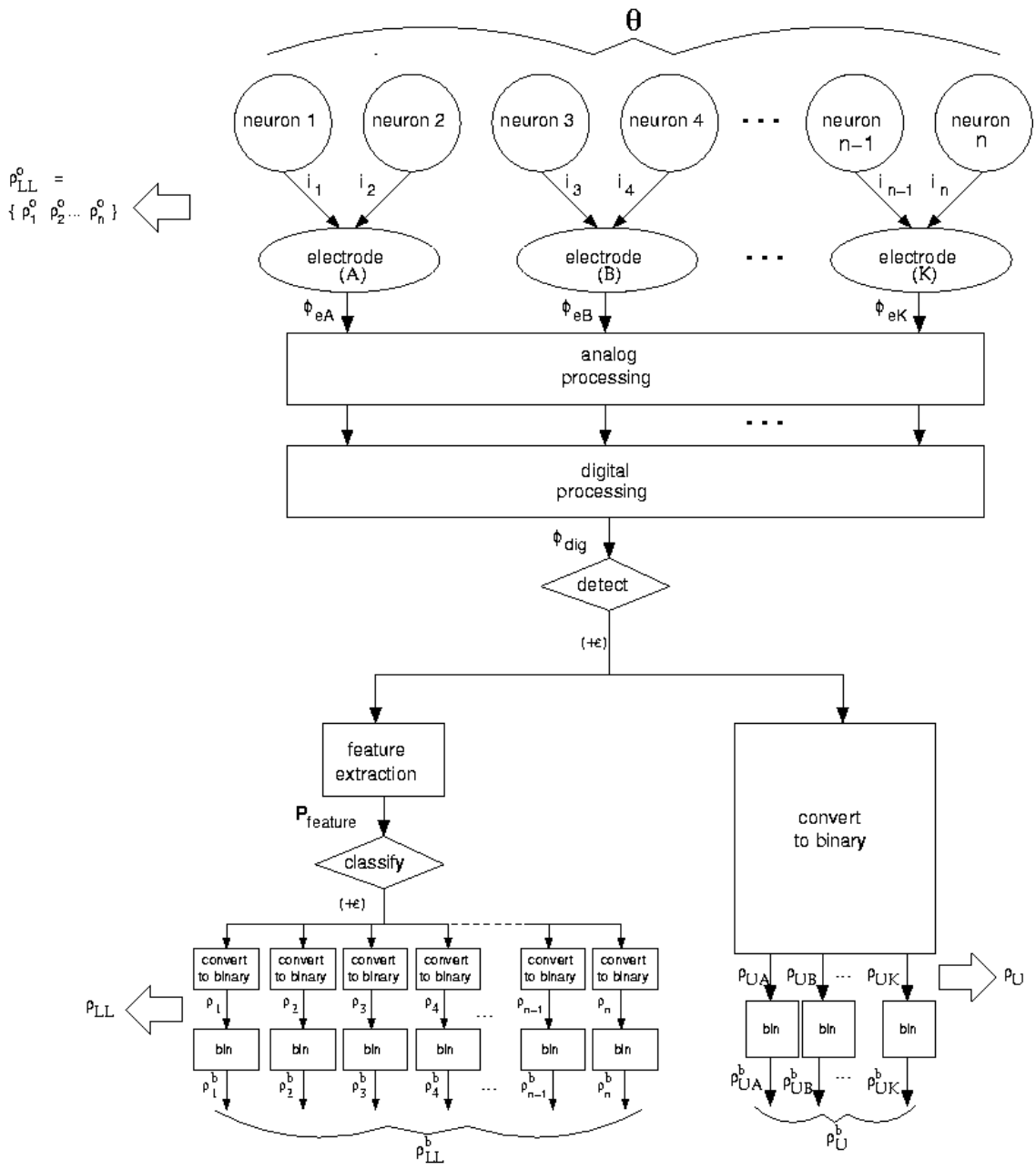
We can imagine the neuron k as producing the spike activity, ρ_k^o, which is the main information-carrying component contributing to i_k. The set of spike trains resulting from the extracellular currents that are detectable by the electrode array, i.e., $\{\rho_1^o, \rho_2^o, \ldots \rho_n^o\}$, is analogous to $\mathbf{P}$ from Fig. 2.1, which represents the unadulterated population activity by which the brain has encoded information about the signal of interest, $\boldsymbol{\theta}$. Each electrode implanted in the brain measures the extracellular potential, ϕ_e, resulting from the summation of these currents from typically 2-3

neighboring neurons [15, 36]. Analog hardware conditions and digitizes these signals for further digital signal processing. A detector then determines when the amplitude of the digitized extracellular signal crosses a threshold. Although detection may be possible to implement in analog hardware, it is currently implemented in the digital portion.

Upon a detection, the processing paths diverge. To perform spike sorting, the action potential waveform is retained and used for feature extraction and classification of the source neuron. The output is a set of spike trains, $\{\rho_1, \rho_2, \ldots \rho_j \ldots \rho_n\}$, labelled with neuron j's identity. This set collectively is called the labelled-line signal and denoted by $\boldsymbol{\rho}_{LL}$. Further processing in BCIs aims to extract information from $\rho_{\mathbf{LL}}$, the population of n spike sorted signals, which consists of the set of spike trainssorted off all K electrodes. As seen in the block diagram of Fig. 2.2, spike sorting is an attempt to reproduce the original information-carrying neural signals, namely the spike trains of individual neurons. These spike trains $\rho_{\mathbf{LL}}$ are typically binned in 20-100ms time

---

Figure 2.2 *(following page)*: Processing steps in a BCI. Conventional processing including spike sorting is shown on the left branch. Processing in a BCI without spike sorting is shown on the right branch. $\boldsymbol{\theta}$ is the target signal about which the BCI aims to extract information. Electrodes are implanted to record signals from neurons which are presumed to encode information about $\boldsymbol{\theta}$. Each neuron j produces an extracellular current i_j. These contribute to the extracellular potential detected by a nearby electrode. The extracellular voltage ϕ_e is processed by analog hardware which provide buffering, gain, and filtering for the neural signals. These analog signals are digitized and further processed by digital hardware. To perform spike sorting, the action potential waveform is retained and some feature of the spike waveform is extracted to be used for classification. Each spike is classified as belonging to one neuron or as a noise spike. The output of the classifier is a set of labelled spike trains, $\{\rho_1, \rho_2, \ldots \rho_j \ldots \rho_n$, collectively known as $\boldsymbol{\rho}_{LL}$, the "labelled-line" signal. More computation is then performed on these spike trains to decode $\boldsymbol{\theta}$. Typically, the first step in the decoder is to bin $\rho_{\mathbf{LL}}$ in 20-100ms time windows to produce $\rho^{\mathbf{b}}_{\mathbf{LL}}$. To produce a non-sorted signal, the action potential waveform is discarded, and only the times of spike occurrences are retained, yielding a single spike train ρ_{Uk} for each channel k. These multi-unit spike trains might also be binned in the decoding process to yield ρ^b_{Uk}.


θ
neuron 1
neuron 2
neuron 3
neuron 4
neuron n−1
neuron n
i1 i2 i3 i4 in−1 in
electrode (A)
electrode (B)
electrode (K)
ϕeA ϕeB ϕeK
analog processing
digital processing
ϕdig
detect
(+ε)
feature extraction
Pfeature
classify
(+ε)
convert to binary
ρ1 ρ2 ρ3 ρ4 ρn−1 ρn
bin
ρ1b ρ2b ρ3b ρ4b ρn−1b ρnb
ρLL
ρLLb
convert to binary
ρUA ρUB ρUK
ρU
ρUAb ρUBb ρUKb
ρUb
ρLL° = { ρ1° ρ2° ρn° }

windows to produce $\rho^{\mathbf{b}}_{\mathbf{LL}}$. Hence, $\rho^{\mathbf{b}}_{\mathbf{LL}}$ consists of the sequence of spike counts for each sorted unit in the population.

To produce a non-sorted signal, the action potential waveform is discarded after detection, and only the times of spike occurrences are retained, yielding a single spike train ρ_{Uk} for each channel k. Leaving out spike sorting would similarly produce a population of binned spike counts $\rho^{\mathbf{b}}_{\mathbf{U}}$, but spike counts of action potentials detected on each electrode, not the action potentials of each sorted unit. Without spike sorting, the digitized extracellular waveform is converted to a binary signal. Thus, all these steps that comprise the larger signal processing block of Fig. 2.1 to produce $\mathbf{P_{proc}}$, representing binned non-sorted signals in this case, potentially lead to information loss.

As we have just described, spike sorted signals are also the output of a processing chain, and can also be represented by $\mathbf{P_{proc}}$. Thus, the processing inequality also holds, and we know that there can be no more information about $\boldsymbol{\theta}$ in the spike sorted signals than in the original neural population. The signal processing block represented in Fig. 2.1, however, is different in the case of non-sorted and sorted signals. We see that spike sorting, similar to the path producing non-sorted signals, involves signal processing by which information is necessarily lost, or at best, maintained.

There are two places along this signal processing path at which decisions are made and spike error can be introduced. When digitized extracellular waveforms reach the detection step, the detector determines occurrences of events, upon which further processing takes place. At this point, action potentials can be missed or a threshold crossing can be falsely detected as an action potential, creating detection error. Along the spike sorting pathway, a classification decision is made depending on the outcome of processing features of the action potential waveforms. At this step, spike error can again be introduced if spikes belonging to one neuron, or non-

biological artifact spikes, are falsely classified as belonging to another.

Both spike sorted and non-sorted signals are several processing steps away from the original population activity $\mathbf{P}$. Spike sorting requires a different processing path from that which produces unsorted signals. Therefore, nothing can be strictly said about which set of signals have more information. What *can* be said is that

$$I(\theta; \boldsymbol{\rho}^b_{LL}) \;\leq\; I(\theta; \boldsymbol{\rho}_{LL}) \;\leq\; I(\theta; \boldsymbol{\phi_{dig}}) \tag{2.2}$$

$$I(\theta; \boldsymbol{\rho}^b_{U}) \;\leq\; I(\theta; \boldsymbol{\rho}_{U}) \;\leq\; I(\theta; \boldsymbol{\phi_{dig}}) \tag{2.3}$$

It has been assumed that the information loss from pooling signals is much greater than the loss incurred by spike sorting, but in actuality, it is not known whether $|[I(\theta; \boldsymbol{\rho}^b_{LL}) - I(\theta; \boldsymbol{\rho}_{\mathbf{LL}})] + [I(\theta; \boldsymbol{\rho}_{LL}) - I(\theta; \mathbf{P}_{\mathbf{feature}})] + [I(\theta; \mathbf{P}_{\mathbf{feature}}) - I(\theta; \mathbf{P}_{\mathbf{dig}})]|$ is less than, greater than, or equal to $|[I(\theta; \boldsymbol{\rho}^b_{U}) - I(\theta; \boldsymbol{\rho}_{\mathbf{U}})] + [I(\theta; \boldsymbol{\rho}_{U}) - I(\theta; \mathbf{P}_{\mathbf{dig}})]|$. From an information-theoretic perspective, spike sorting does not necessarily lead to information gain over non-sorted signals, and the uncertainty of what information in $\boldsymbol{\rho}^o_{LL}$ is contained in $\boldsymbol{\rho}_{LL}$ is apparent in chapter 3. Before delving into the analysis, we describe the measure we use to quantify the benefit of spike sorting.

2.2 Definition of mutual information

Mutual information is a measure of the reduction in uncertainty about a target signal $\boldsymbol{\theta}$ by observing some messenger signal $\boldsymbol{\rho}$ [37]. Claude Shannon developed a mathematical definition of this quantity for applications in communication systems. The basis for mutual information is the quantity defined as *entropy*. Eqn. 2.4 gives the definition for the entropy, $H(\boldsymbol{\theta})$, of a signal $\boldsymbol{\theta}$. It is dependent on the probability distribution of $\boldsymbol{\theta}$, which we denote as $p(\theta)$. Henceforth, we will denote the probability

of a random variable X as $p(X)$, the conditional probability of a signal X conditioned on Y as $p(X|Y)$, and the joint probability of X and Y as $p(X, Y)$.

$$H(\boldsymbol{\theta}) = -\sum_{\boldsymbol{\theta}} p(\theta) \log_2(p(\theta)) \tag{2.4}$$

Entropy is a measure of uncertainty or randomness of a signal. A signal $\boldsymbol{\theta}$ representing the outcome of coin flips, for example, has 1 bit of entropy. The number of bits of entropy can be interpreted as the fewest number of "yes/no" questions required to determine the outcome (e.g., "Was it a 'Head'?").

Any signal $\boldsymbol{\rho}$ which is represented by discrete symbols can be thought of as a carrier for the information in $\boldsymbol{\theta}$. If by observing $\boldsymbol{\theta}$, the entropy of signal $\boldsymbol{\rho}$ is reduced, then there is (positive) mutual information in $\boldsymbol{\theta}$ and $\boldsymbol{\rho}$. The degree to which the entropy is reduced depends on the encoding and quality of the communication system. The entropy of $\boldsymbol{\theta}$ given observations of $\boldsymbol{\rho}$ is called the conditional entropy, $H(\boldsymbol{\theta}|\boldsymbol{\rho})$; it is the average uncertainty of $\boldsymbol{\theta}$ given observations of $\boldsymbol{\rho}$. The conditional entropy $H(\boldsymbol{\theta}|\boldsymbol{\rho})$ is defined in Eqn. 2.5

$$H(\boldsymbol{\theta}|\boldsymbol{\rho}) = -\sum_{\boldsymbol{\rho}} \sum_{\boldsymbol{\theta}} p(\theta, \rho) \log_2(p(\theta|\rho)) \tag{2.5}$$

The mutual information, $I(\boldsymbol{\rho}; \boldsymbol{\theta})$, is then given by the difference in the total entropy and conditional entropy, as show in Eqn. 2.6. This states that the mutual information (MI) is the reduction in uncertainty about ρ after observing θ as compared to having no knowledge of θ; since MI is the *reduction* in uncertainty, we take the negative of the difference in parentheses in equation 2.6.

$$\begin{aligned} I(\boldsymbol{\rho}; \boldsymbol{\theta}) &= -(H(\boldsymbol{\rho}|\boldsymbol{\theta}) - H(\boldsymbol{\rho})) && (2.6) \\ &= H(\boldsymbol{\rho}) - H(\boldsymbol{\rho}|\boldsymbol{\theta}) && (2.7) \end{aligned}$$

Equations for continuous signals are analogous (Eqn. 2.8 and 2.9).

$$H(X) = -\int_X p(x)\log_2(p(x))dX \tag{2.8}$$

$$H(X|Y) = -\int_Y \int_X p(x,y)\log_2(p(x|y))dX\,dY \tag{2.9}$$

To view MI in terms of neural responses, we can state that total entropy, or formal information, would describe the overall capacity of the neural response to encode at all; it quantifies how many "code words" can be formed from the "alphabet" of responses which neurons can output. Mutual, or attribute-specific, information quantifies how much of a particular stimulus parameter, e.g. magnitude, can be extracted from the output; it quantifies how many neural response words are formed which encode that stimulus parameter. We take a closer look at how it can be applied to our problem of measuring the benefit of spike sorting.

2.3 The utility of mutual information

Neural encoding and decoding have attracted the interest and efforts of many researchers. Perception, consciousness, attention, and movement all involve some level of control by electrical signals generated by neurons; these signals have been understood to represent sensory stimuli, motor commands, and the like. As researching capabilities to access information in neural systems grow, the need for quantitative analysis grows. Mutual information is one approach used to analyze neural systems [35]. It is already becoming a more widely accepted measure in BCIs [9]. Several reasons led us to choose mutual information for our simulation studies on spike sorting of neural population spike activity, as outlined in this section.

- **Analogy between communications systems and neural networks:** The mechanism by which neurons communicate has been studied using electrophysiological techniques to record and analyze signals from neurons. It is commonly accepted that networks of neurons convey information at a system level by generating action potentials, or neural spikes [38, 35]. While mutual information was originally developed for telecommunications systems, it is applicable to the questions addressed here insofar as neural encoding and decoding is a form of transmitting and receiving information. A message signal, whether it is a sensory stimulus or a command signal to control motor output, is communicated via electrical activity generated by neurons. At a cellular level, each neuron acts as a communication system, receiving presynaptic currents as input and generating post-synaptic currents to transmit information to downstream neurons. In the motor system, a command signal is communicated to the actuators.

  The appropriateness and relevance of mutual information can be seen in the framework of a BCI. In a BCI, the intended motor output is the signal to be communicated; the neural population activity serves as the carrier of the motor signal; and the BCI device receives the information communicated by the acquired neural signals. One goal in a BCI is to recover as much of the information encoded by the neural responses in real-time using some computational decoding algorithm.

- **Multi-dimensional signals:** An important advance in multi-neuron recordings has been the ability to study populations of neurons. However, the measures that are often used in single-unit signal analysis are often limited in their application and not easily extended to multi-dimensional signals. While it can be useful to examine pairwise correlations between neurons, it is not easy to assess what all the pairwise correlations in a population mean as a whole. Signal-

to-noise ratio is another useful quantity in assessing communication channels, but to our best knowledge, there is no analogous measure for SNR of a population of channels. As further described in Sections 5.4.4 and 6.3, signal-to-noise ratio (SNR) has limited capability for our multi-dimensional problem. In contrast, MI is not restricted to 1-dimensional signals, and has been noted as a suitable way to analyze neural codes in populations of neurons [39].

- **Non-linearities are included:** The relationship between neural responses and their stimuli is commonly represented and analyzed by linear correlation or coherence [40, 41, 42, 43, 44, 45]. Although there is still a question as to whether the primary information can be extracted by linear decoding, we desire a measure that would capture information conveyed in the non-linearities of the code if it exists. Mutual information is able to capture the information in non-linear codes.

- **Signals of different domain:** Shannon information can provide a measure of correlation between signals of different domains; for example, a continuously varying stimulus signal and a discrete binary signal representing the neuronal output, as in Section 3.3.1. MI also provides a way to quantify the effects of spike sorting error and gain insight into the value of spike sorting without assuming an understanding of the mechanism of neural processing.

- **No decoder needs to be specified:** Our aim is to compare information available in recordings of extracellular spike activity for neuroprosthetic applications with and without spike sorting. The ideal experiment would be to record non-sorted spike trains from implanted electrodes in motor regions while recording motor output during a motor task, and compare performance of a decoder that uses the non-sorted multi-unit spike trains and the corresponding

sorted spike trains. However, such an experiment would require the choice of a decoder. As mentioned earlier, the precise nature of motor encoding in the brain and the decoder that would maximize extracted information is not agreed upon. Meanwhile, "[t]his direct method [of estimating mutual information rates] makes no assumptions about the neural code and is exact in principle (Borst and Theunissen 1999)" [46].

2.4 Limitations of mutual information

Mutual information is theoretically a versatile and relevant tool, as shown in the previous section. However, when it cannot be computed analytically, it is only an estimate of the information. The accuracy of the estimate depends on the amount of data that can be acquired [47]. Bias estimates are required to correct for limited amounts of data, which is particularly a problem with *in vivo* data [48].

Mutual information analysis does not necessitate specification of a precise encoding model, or presume anything about the decoding process. However, the signal carriers, whether spike counts or spike patterns, angular position or muscle activation, etc., still need to be specified.

While the application of MI to neural responses is conceptually straightforward, interpretating MI values is not simple. High information values will not necessarily indicate how to access the information. It does not tell us how to design a BCI decoder. One crucial piece of the puzzle that will affect the comparison of a BCI's performance with and without spike sorting but is not included in these analyses is the decoder that is used. That is because we first want to understand what information is available in sorted and non-sorted signals before we attempt to try out various decoders to determine the information extracted by a BCI.

2.5 Implementation

Different methods are available for calculating MI. Direct methods require knowledge of the joint stimulus-response probability distributions, whereas indirect methods can approximate information decoded from the neural response on a single presentation of the stimulus.

2.5.1 Direct information

The quantity $I(\boldsymbol{\theta};\boldsymbol{\rho})$ is called *mutual* information because of its commutative property; i.e., it tells us how much information $\boldsymbol{\theta}$ and $\boldsymbol{\rho}$ convey about each other. This property becomes apparent when equations 2.4 and 2.5 are substituted into equation 2.6 to yield

$$I(X;Y) = \sum_{Y}\sum_{X} p(x,y)\log_2 \frac{p(x,y)}{p(x)p(y)} \tag{2.10}$$

Note that interchanging x and y in equation 2.10 does not change this equality. Therefore, while we have defined $I(\boldsymbol{\theta};\boldsymbol{\rho})$ to be the reduction in entropy of $\boldsymbol{\theta}$ given observations of $\boldsymbol{\rho}$, it is equivalent to the reduction in entropy of $\boldsymbol{\rho}$ by knowing $\boldsymbol{\theta}$ ($H(\boldsymbol{\rho}) - H(\boldsymbol{\rho}|\boldsymbol{\theta})$). In our computations, we computed the total entropy of $\boldsymbol{\rho}$, then the entropy $H(\boldsymbol{\rho}|\boldsymbol{\theta})$ for mostly arbitrary reasons, but in part because it was easier to divide the data according to the known target values when determining the conditional probability. $H(\boldsymbol{\rho})$ represents how much disorder we will observe in the responses without regard to the target signal; then, viewing the responses with regard to the target signal that gave rise to it reveals some order, which results in $H(\boldsymbol{\rho}|\theta)$ being smaller than the total entropy of $\boldsymbol{\rho}$. This decrease in disorder is the mutual information.

In order to compute information directly, the total and conditional probabilities of $\boldsymbol{\rho}$ need to be determined. If the distribution is not known, then it can be estimated

empirically as p_{emp}. This is done by determining the empirical quantities for the probability $p(\boldsymbol{\rho})$ and conditional probability $p(\boldsymbol{\rho}|\boldsymbol{\theta})$. To obtain these quantities for a variable $\boldsymbol{\rho}$, the frequency of each discrete value ρ was tallied to calculate the empirical probability of $\boldsymbol{\rho}$ taking on the value ρ.

$$p_{emp}(\mathbf{X} = \mathbf{x}) = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{k} \delta(t_i - \tau_j) \tag{2.11}$$

where N equals the number of samples, δ(t) is the Kronecker delta function, and $\{\tau_j\}$ denotes the k occurrences of the variable X taking on the value x. These MI computations were coded both in Matlab and in C, and the specifics of the implementation varied slightly, but the general methodology remained the same in both implementations.

In the original application of Shannon information, the signals were communicated using a finite set of discrete symbols. For a continuous signal or a signal with an infinite number of possible values, the signal must be quantized before the entropy can be computed empirically. The signal X was quantized into n_q quantization levels, such that the quantization level of any value of X ranged from 0 to $n_q - 1$. The quantization level of a signal value x restricted to the range $[x_{min}\ x_{max}]$ is given by equation 2.12 and used to represent the values of signal X when computing the probabilities.

$$i = \frac{x - x_{min}}{x_{max} - x_{min}} n_q \tag{2.12}$$

The obvious caveat is that that the fewer quantization levels that are used, the less precise the representation of the signal distribution is; however, the more quantization levels used, the more data must be acquired to adequately sample the signal at each value.

The quantized signals were then tabulated to construct a joint occurrence matrix. We can think of this as an $n_{q\boldsymbol{\theta}} \times n_{q\boldsymbol{\rho}}$ matrix, which stores the number of occasions on which the pair (θ_i, ρ_j) occurred simultaneously. First, the target signal $\boldsymbol{\theta}$ is sorted, and $\boldsymbol{\rho}$ is put in the same order as $\boldsymbol{\theta}_{sorted}$. Then $\boldsymbol{\rho}$ within each value of $\boldsymbol{\theta}$ is sorted. The sorted signal can then be scanned for changes in the value of $\boldsymbol{\rho}$ to determine how many instances each value (ρ_j) occurs while $\boldsymbol{\theta} = \theta_i$. The row vector obtained by summing across rows and then dividing by the total number of data points would give the total probability of $\boldsymbol{\rho}$, with $p_{emp}(\boldsymbol{\rho} = \rho_j)$ given by the normalized sum in column j. The conditional probability of $\boldsymbol{\rho}$ given $\boldsymbol{\theta}$ is given by the quotient of each row divided by the sum of the given row.

2.5.2 Indirect information

Indirect information refers generally to information estimates that are computed by means other than a direct method. For example, information calculated indirectly could be an estimate of the information in neural activity that is found by computing the information that is decoded from the neural signals. It can be computed for various purposes but the common aspect is that the information values are obtained without having to compute the joint histogram of a population response, which is exceedingly computationally burdensome.

One purpose it serves is to provide a lower bound on information in the neural signals themselves when it is difficult or impossible to determine confidently the information directly. For example, suppose we wanted to examine the information in a population of neurons. The information could be computed between the target signal and the output of a decoder which estimates the target signal from the population activity. Computing information between the decoded and actual target signal bypasses the need for computing the joint histogram.

Another purpose might be to determine the information that a particular decoder can provide. For example, if we desire to know how much information can be linearly decoded from neural signals [46], we could apply the optimal linear decoder to the neural signals and then calculate the information between the output of the decoder and the target signal which was encoded in the neural signals. One method that has been commonly applied for a continuously time-varying target signal is to use the Wiener-Hopf optimal filter to decode the signal and then use equation 2.13 for a Gaussian channel [49, 50, 51]. A Gaussian channel is a linear time-invariant system in which the amplitude of the input is Gaussian-distributed. The neural input-output system is then modelled as a Gaussian channel with noise. This is done by using a continuously-time varying signal as the input to an integrate-and-fire (IAF) neuron. The stochastic nature of the neural response arises from the variable threshold of the IAF neuron; the threshold was normally-distributed and randomly chosen after each occurrence of a spike. In this system, the mutual information rate is given by $\dot{I}(\boldsymbol{\theta};\boldsymbol{\rho})$ in Eqn. 2.13:

$$\dot{I}(\boldsymbol{\theta};\hat{\boldsymbol{\theta}}) = \frac{1}{2}\int_0^\infty \log_2(1+SNR(f))df \leq \dot{I}(\theta;\rho) \tag{2.13}$$

where $SNR(f) = P_\theta(f)/P_{neff}(f)$, the ratio of the power spectrum of the signal $\boldsymbol{\theta}$ to that of the effective noise (or input-referred noise, respectively.

2.5.3 Validation

Simple 1-dimensional (1-D) systems and n-D binary systems were simulated in Matlab to validate direct mutual information computation. Although these systems were non-biological in nature, they served as known systems whose information values could be derived analytically. These Matlab-generated signals were input to the Matlab

functions developed previously; they were also saved to files to be used as inputs to the C functions to compute mutual information. The output of the C routines were compared to the output of the Matlab code for identical signal data to validate the C code.

1-D system test case: Two random signals were generated and arbitrarily designated as a "stimulus" and a "response". The stimulus signal was simulated by drawing randomly from a uniform distribution ranging from 0 through 4. An independent random signal was generated that was uniformly-distributed from 0 through 6. It was verified that the sorting and histogramming required for the computation of mutual information (Sec. 2.5.1) was performed properly. The mutual information was correctly calculated to be 0 and the total entropy of the signals was correctly calculated to be $log_2(5) = 2.3$ bits and $\log_2(7) = 2.8$ bits, respectively.

Binary system test case: A simple binary system with a 2-dimensional target s and 3-dimensional "response" signal r was generated, which were again independent. Correct sorting was again demonstrated, as illustrated in Fig. 2.3. The upper panel shows the target signal, and the bottom panel shows the response signal after sorting in ascending order. The step sizes are roughly equal because both are drawn from a uniform distribution. The shape of the sorted response signal across target signal values is similar because the response signal is independent of the target signal values.

The conditional entropy and total entropy were found to be $H(r) = 3.0$ bits and $H(r|s) = 3.0$ bits. When inputting signal s as both the input and output signals, the C function correctly computed the total and conditional entropies to be $H(s) = 2.0$ bits and $H(s|s) = 0$ bits. Similarly, when using r as both the input and output of the system, the computed entropies were $H(r) = 3.0$ bits and $H(r|r) = 0$. The entropy values were computed in Matlab on the same signals, yielding equal results as the C functions.

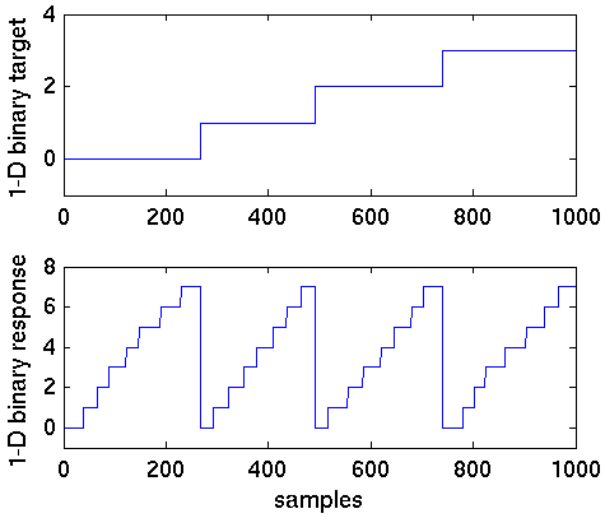


Figure 2.3: The sorted 2-D binary target and conditionally sorted 3-D binary response were converted to 1-D multi-valued signals, to show the proper sorting of signals. **a)** [top] Sorted target signal in ascending order. **b)**[bottom] Response signal sorted conditioned on the target signal.

Correlated signal test case: A binary stimulus signal and a response with 16 possible values was generated from a random uniform distribution. The response ranged from 0 to 7 when the stimulus s = 0, and from 8 to 15 when s=1. $H(r)$ was correctly calculated to be 3.99 bits and $H(r|s) = 2.99$ bits. Different combinations of input/output were computed and compared against the analytically calculated values as shown in Table 2.1.

Table 2.1: Results of test case of correlated binary multi-dimensional signals. Total entropy of the signal designated as the output signal and conditional entropy of the output signal conditioned on the input signal is tabulated, and shown adjacent to the true analytical values.

s_{in}	s_{out}	$H(s_{out})$	True $H(s_{out})$	$H(s_{out}\|s_{in})$	True $H(s_{out}\|s_{in})$
s	s	1.00	1	0.00	0
r	r	3.99	4	0.00	0
s	r	3.99	4	2.99	3
r	s	1.00	1	0.00	0

2.5.4 Computational considerations

Simulation studies not only provide one way of addressing questions about the effects of spike sorting on neural decoding in a controlled environment, they also reduce the bias in information-theoretic measures by providing the long duration of signals required for these estimates that are difficult to obtain *in vivo*. Limited *in vivo* data has been an obstacle to calculating MI accurately for populations of neural activity [52, 53, 54]. In previous neurophysiological studies, mutual information estimates have been devised to correct for the limited sample sizes [55]. The measures have also been demonstrated on single-unit or 1-dimensional activity but not on populations of neural activity even in simulation studies [56, 57]. Mutual information computations have been performed on pairs of neural signals *in vivo* but with non-sorted signals [58]. The indirect information estimate (equation 2.13) is a well-trusted method that is useful for initially gaining an understanding of the information that sorted signals provide [51, 35], and was well-suited to the exploration of linearly decoded information in chapter 4.

Once the initial analyses allowed us to gain a general understanding of the main factors in MI loss due to pooling, we were able to construct more realistic simulations and use the direct calculation of information. This allowed us to compute MI in simulated multi-unit neural signals with confidence, but this required very long simulation lengths. Direct computations of MI were performed in chapter 5 to measure the MI between a target signal θ which represented the direction of movement and a 4-unit population of neural spike activity, which exhibited cosine-tuning properties. These signals will be described further in chapter 5, but for now, we note some of the main issues with implementing the computations. Matlab was unable to handle simulation lengths long enough for accurate computation of MI in the 4-unit population signals. It also was generally incapable of completing construction of the signals that were

used in these computations. Thus, C code was written to both generate the data signals of sufficient length and directly compute MI.

2.5.5 Improving efficiency

C code was straightforward to implement for small data lengths. However, for data lengths that required megabytes of storage, memory capacity was a concern as well as simulation time. As versions of C code progressed, the simulations were able to handle longer data lengths, and shorter durations of time were necessary for the information computations.

The main programming developments that enabled these improvements in memory usage and run time are described below.

Changing sort routine: The quick sort function provided in [59] was efficient for most cases. However, when original data was already well-sorted, the call depth increased by order n. This was a problem for the center out task, in which movements were created in one direction for very long segments. The function was changed to include a recursive check of whether the particular segment of data was already sorted. If so, then it need not continue iterating.

Dynamic memory allocation: Allocating memory dynamically allowed space to be freed up once it was used. This was particularly necessary during sorting routines which could iterate.

Data type: Short integer types were used, but caution had to be used to prevent undesired consequences. For instance, indices could not be changed to short integer types. We also had to ensure that the maximum spike counts were not limited by the data type range (256 in the case of short integers). Binned counts needed to be limited to values between 0 and 255. This would be a high count

for any reasonable firing rate and bin size, but when a firing rate analysis was performed, we had to be sure to either change the data type and be limited to compute information in the single units and pairs but not in the full 4-unit population; or we had to limit the firing rate.

Linked list: It was believed that one way that memory could be used to its fullest capacity was to use an infinite array structure. The infinite array was defined as a data structure which contained a block of fixed size to hold contiguous data, and contained a pointer to the next block.

Now that we have described how MI computations were implemented, we describe how we applied MI to our question of the benefits of spike sorting and describe the findings.

Chapter 3

The costs and benefits of spike sorting

The primary reason for the uncertainty in how a BCI with spike sorting would compare with one without is two-fold: 1) it is theoretically possible for non-sorted signals to provide equal information with sorted signals in a BCI and 2) in practice, spike sorting does not perfectly replicate the spike activity in the brain, thus creating potential for information loss that processing without sorting would not create. From the point of view of information processing described in the previous chapter, we define our two main questions, referring again to Fig. 2.2:

1. Without spike sorting, how much information is ρ_U able to retain even after discarding information in the waveform shapes?

2. When spike sorting is performed, how much information is lost due to sorting error?

In this chapter, we begin to investigate the latter question. We describe the challenges of performing spike sorting and its unreliability. Evidence is then provided through simulation results for the informational cost spike sorting can potentially incur due to sorting error.

3.1 Spike error

Even without the constraint of implementation in a digital signal processing IC or a portable system, researchers have struggled to develop an automated spike sorter. Spike sorting has attracted the devoted attention of many researchers for decades in order to make improvements to the ability to discriminate action potentials [29, 30,

60, 61, 18, 19]. The signal-to-noise ratio (SNR) of biological signals is often too low for individual neurons to produce action potential waveforms which segregate nicely into separable clusters in some feature space. It is also difficult to evaluate spike sorting performance, to compare spike sorters, and therefore to design improvements, because the actual discriminated spike times are not truly known [62].

Furthermore, even with manual supervision of spike sorting, the process introduces error. The one study that analyzed error between spike sorted signals and truly known signals (through simultaneous intracellular and extracellular recordings) reported typical errors ranging from $0 - 30\%$ but there were instances of false positive rates as high as 43% and false negatives as high as 67.3%. This study was done using tetrodes, bundles of 4 electrodes, whereas most BCI applications use arrays of single electrodes. Tetrodes would reduce the channel density but were designed for the sake of improving spike sorting performance [20]. Thus, using single electrodes instead of tetrodes would only be expected to cause higher sorting error than what was reported in this study. Also, a spike sorter in a BCI would be expected to have less supervision, and thus, greater error.

Another estimate of typical spike sorting error reported false positive and false negative error rates of 23% and 30%, respectively [63]. This estimate was made using simulated data with known action potential times and noise. This is the most effective way currently to evaluate spike sorting, indicating the difficulty of validating spike sorter performance.

Spike sorting is a challenging task for many reasons. If spikes from neighboring neurons occur very closely spaced in time on the same channel (e.g., if the depolarizing peaks are within 1ms of each other), the second spike will not get detected. When a threshold crossing (detection) occurs, the 1ms of waveform containing the spike is extracted and no other detection can be made within this period (and for a specified

period after the window to allow for the refractory period). An example of this false negative error can be seen in Fig. 3.1. Action potentials can even overlap, and discrimination of superimposed action potentials has been considered by some to be an insurmountable task [29]. Errors seem to occur most consistently during population bursts and when normally quiet neurons spike [20]. Also, spike sorting relies at least partially on human judgment. Spike waveforms may also change over time, requiring a human operator to check threshold and boundary settings before each session.

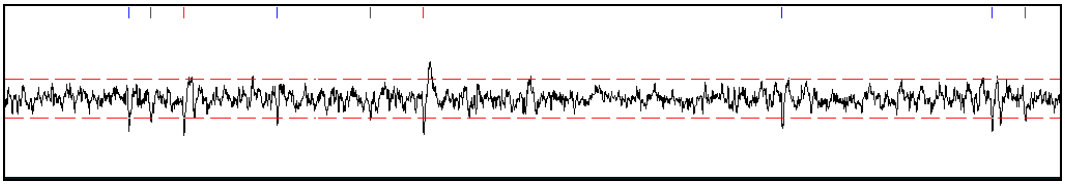


Figure 3.1: An example of spike sorting in which action potentials overlap in time preventing correct sorting. The tick marks above the continuous signal show instances of spike detection. At the time of the 2^{nd} to last tick mark, two action potentials occur very closely in time, but only the first of the pair is detected. Data was collected according to IACUC approved protocol.

It is important to recognize that there is a possible source of spike error in non-sorted signals. The first step in sorting signals is detection of action potentials. This step is necessary regardless of whether the detected spikes are then classified. Depending on the SNR of the recordings and detection threshold settings, action potentials might be missed altogether or artifact spikes may be detected. Spike sorters provide the potential advantage of leaving out misdetected false alarm spikes by classifying them as a "noise" unit. If they can be sorted out from the true action potentials, they can then be thrown out before the rest of the sorted spike trains are used in a decoder. Non-sorted spike trains will contain detection error, some of which spike sorting may eliminate. However, rates of detection error have not been reported, and most of the literature concerning spike sorting focus on classification error.

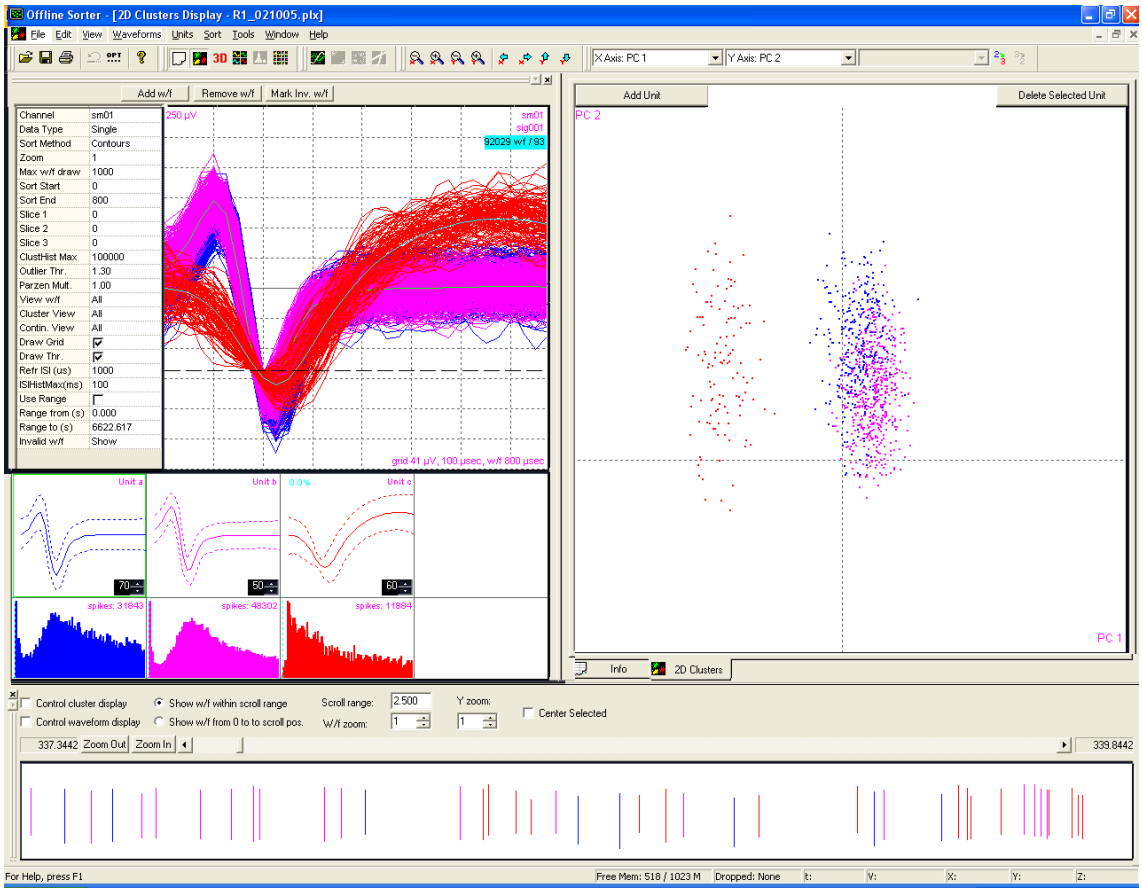


Figure 3.2: An example of spike sorting in which boundaries have been drawn to distinguish between two units that overlap siginificantly in cluster space. Data collection and sorting by Mike Wiest, Ph.D., Duke Neurobiology [1].

3.2 Variability

As described previously, spike sorting requires the setting of many parameters, which subjects the process to user-subjectivity and variability. This creates inconsistencies in the units that are extracted from user to user and potentially from session to session.

The difficulties of spike sorting have been recognized and the efforts to devise a reliable spike sorter extend back decades and continue today [29, 30, 18, 19]: "[S]pike sorting is heavily operator dependent, and consequently, subjective and time consuming. Spike sorting can be especially difficult when the signal to noise ratio is low

or when there there are non-stationaritites within the neuronal signal, such as variations in background neuronal activity" [64]. One study examined the variability of a typical spike sorting method currently used in BCI applications and concluded that spike sorting is highly user-dependent and there is much uncertainty even for a given user; i.e., the same user will sort the same data set differently on different sessions [63]. The study by Harris et al. [20] also revealed differences in error depending on the user.

Each scientist who performs spike sorting must bring some degree of his own art to the process. Figure 3.2 shows detected action potentials from one channel of rat somatosensory data that was spike sorted by a scientist experienced in multi-electrode recording and spike sorting [1]. The boundaries could be set in a variety of ways without objective criteria. Because of the large degree of overlap in cluster space, it is apparent that it would take extreme expertise to sort out more than one neuron from the cluster of yellow and green points on the right in Fig. 3.2. If BCI performance depended heavily on unit activity being correctly sorted, there will be little predictability in its performance, since spike sorting outcome is inconsistent.

The shortcomings of spike sorters are a testament to the difficulty of the problem. Admirable efforts have been made in spike sorting research. Assumptions have been based on intuition and sound judgment. Because of the success that *has* been acheived in spike sorting, it is easy to overlook the consequences of imperfection in spike sorting or to neglect returning to the detailed problems. Sometimes problems can only be foreseen or dealt with once other larger or immediately important ones have been tackled, as was the case in BCI technology. The fundamental components of a BCI have been developed. Now, we can backtrack and find room for improvements. Spike sorting development has likewise proceeded with sound judgment even in the absence of a fully-informed design process. However, the shortcomings of spike sorting may

prevent optimal BCI performance, and we should pause before implementing spike sorting in future generations of BCIs.

Given the task of maximizing the information obtained from a population of neurons, spike sorting appeals to logic and intuition. Separating spike trains according to their originating sources would maximize the number of distinct carriers of information and allow interactions between neurons to be exploited or studied. If spike sorting could perfectly replicate the spike activity produced by the neurons in the recorded population activity, then it would provide the maximum information available in the recorded neural spike activity. Little work has been published which analyzes how imperfections in replicating unit spike activity affect extracted information, or on how much of that maximum information non-sorted signals could provide.

The studies described in this dissertation address the question of how much information is available in recordings of extracellular spike activity for neuroprosthetic applications with and without spike sorting. The ideal experiment would be to record non-sorted spike trains from implanted electrodes in motor regions while recording motor output during a trained motor task, and compare performance of a decoder that uses the non-sorted multi-unit spike trains and the corresponding sorted spike trains. However, too many variables exist to effectively carry out these costly experiments without first gaining understanding and better defining what decoder and measure of performance should be used. For example, because error-free spike sorting cannot be guaranteed with currently available spike sorters, a truly controlled experiment is not possible. Simulation studies can help motivate these types of experiments, and help in deciding the best approach to obtain and interpret experimental results.

3.3 Effect of spike error on information

As described in section 2.1, in the absence of spike sorting error, if the information is primarily contained in the spike times of individual neurons, and spike sorting reproduced the spike times of the neurons, we could imagine the information loss during the processing in a BCI to be minimal. However, it is possible that error caused by spike sorting could cause significant information loss that might not be incurred if spike sorting were eliminated from the processing. The aim of this investigation was to quantify the effects of incorrectly discriminated spikes on encoded information so as to set tolerable limits on sorting error and gain insight into the importance of sorting. Since detection error would be present, perhaps in greater quantities, even if spike sorting were eliminated from BCI processing, and detection error may have a different effect, we also examine the effect of detection error. We examine the changes in information in ρ_1 of Fig. 2.2 with varying amounts of spike sorting error. To do so, changes in the Shannon information content were observed as spike sorting error was added to simulated neural spike output. This study focuses on the information processing path, highlighted in bold in Fig. 3.3, of a sorted neuron with varying values of ε.

3.3.1 Methods

To compute the information transmission rate, we utilized a signal-reconstruction paradigm commonly done in information-theoretic analysis [51, 49, 50]. The neural response was generated via a simple integrate-and-fire neuron with variable threshold to make the output spike train, $\rho(t)$, a stochastic process [51, 49]. The mean threshold was adjusted to 15mV to produce an average firing rate of 32 spikes/s. Each time the neuron fired, the threshold voltage was set to a new value chosen from a normal distribution with a standard deviation equal to 5% of the mean threshold. A 1000-

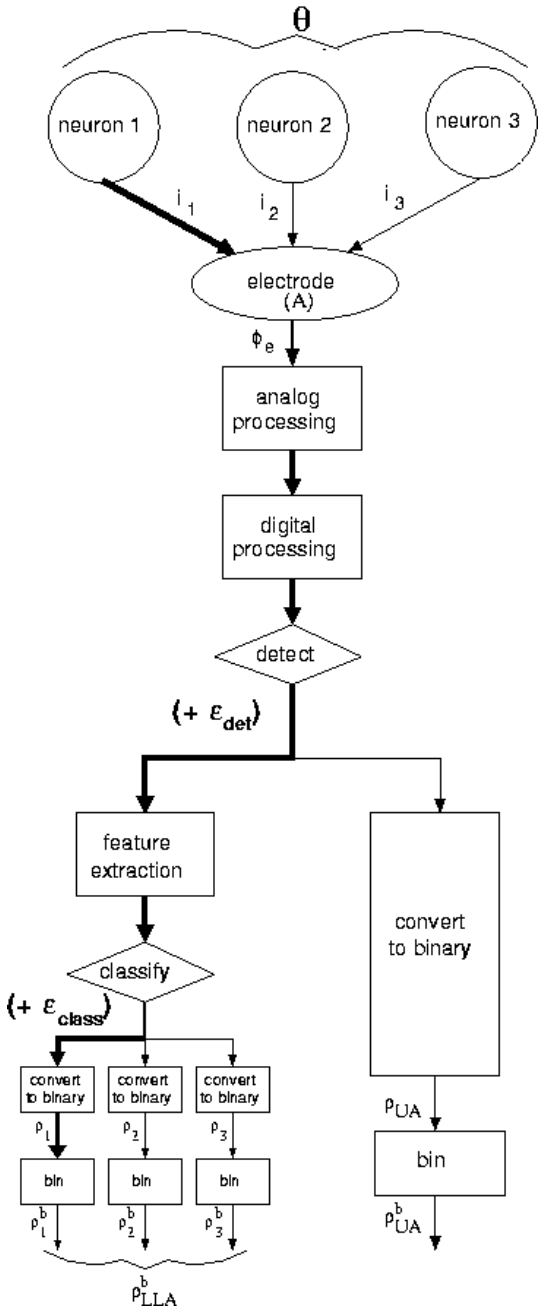


Figure 3.3: Spike sorting is one piece of the processing in a BCI system. The process of sorting can add spike error, as indicated by $(+\varepsilon)$ in this diagram. The analysis in section 3.3 focuses on the effect of sorting error on the information path highlighted in bold.

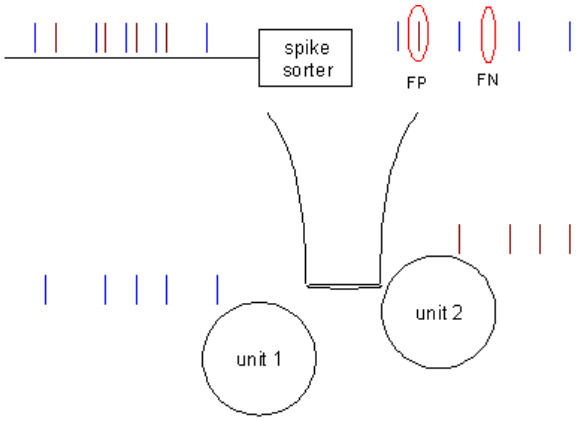


Figure 3.4: The current prototype for multi-neuron acquisition and processing systems involves sorting the multi-unit signal on each channel into single-unit signals before most of the data is discarded and only the onset times of action potentials from each unit are retained. The goal is to retain as much of the relevant information as possible in the original spike trains. The information may be about the stimulus that elicited the response or the behavior that was to be elicited by the neural activity. Ideally, the spike output from the neurons are reproduced at the output of the acquisition system, and the information transmitted to the computational decoders remains unchanged. However, spike sorting does not exactly replicate the spike trains due to the error inevitably associated with spike sorting.

sample/s Gaussian signal was band-limited with a 12-pole low-pass filter with a 20Hz cutoff. This signal simulated the input signal $\theta(t)$ into the neuron. The continuously-varying signal can be viewed as the extracellular presynaptic current containing the essential information of an environmental stimulus or a behavioral target output signal. This approach allows viewing the low-level information without having to know the encoding mechanism of some environmental variable. In the context of a BCI, $\theta(t)$ would represent motor intent of movement.

For each generated response, the optimal linear estimate of the stimulus $\theta(t)$ was computed by applying Wiener-Hopf filtering to $\rho(t)$ [65]. From the error between θ and ρ, the signal-to-noise ratio (SNR) was computed using a Bartlett-windowed periodogram estimate of the signal and input-referred noise power spectra, $P_s(f)$ and $P_{neff}(f)$, respectively. The gain G for calculating input-referred noise, $n_{eff} = \hat{\theta}/G - \theta$, was found using a linear regression between the power spectra of θ and $\hat{\theta}$, as in [49].

The mutual information rate between θ and $\hat{\theta}$, $\dot{I}(\theta;\hat{\theta})$, was then calculated according to equation 2.13 [51, 49].

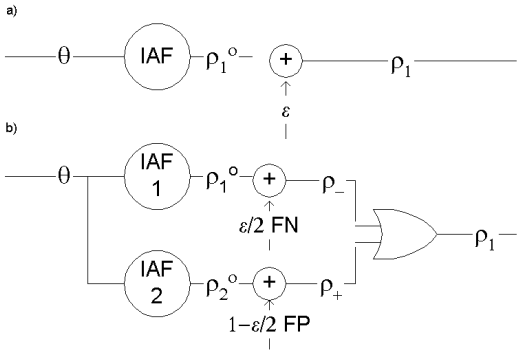


Figure 3.5: An integrate-and-fire (IAF) neuron with a variable threshold produces a spike train ρ_i^o in response to input stimulus θ. False positives (FP) and false negatives (FN) were added at a percent error rate of ε resulting in a spike train with error ρ_i. A neural spike train that has been spike sorted can contain indiscriminate misdetection error (**a**) and misclassification error (**b**). In the former case, random spikes are added and deleted. With misclassification, spikes are generated by two neurons; some spikes generated by unit 1 are misclassified as belonging to unit 2, leaving spikes in ρ_-, while some spikes generated by unit 2 are misclassified as belonging to unit 1 (ρ_+). The spike train with false negatives (ρ_-) and the false postives from unit 2 (ρ_+) are merged to create the spike train with misclassification error ρ_1. This spike train is designated by the spike sorter as unit 1's spike train but contains sorting error. The amount of error is given by ε. In this illustration, half the spike error is false negative error and the other half, false positive.

To simulate misclassification error, false negatives (FN) were added by randomly selecting a certain number of samples and changing the "0"-valued samples to "1". An equal number of samples were selected from a 2nd spike train and added into the original spike train, ρ_1^o, as illustrated in Fig. 3.5 [bottom]. The second spike train, ρ_2^o was generated with the same method as ρ_1^o. The %FN error rate was defined as the percentage of true spikes that were misclassified as "0"s. The %FP was also referenced to the number of true spikes in ρ_1^o; i.e., for a FP rate of ε_{class}% the number of samples changed from "0" to "1" was equal to ε_{class}% of the number of spikes in ρ_1^o. Thus, the % error is, in general, the ratio of the number of incorrect spikes to the

total number of correct spikes, expressed as a percentage:

$$\varepsilon = 100 \left(\frac{\Sigma(\rho_1 - \rho_1^o)}{\Sigma \rho_1^o} \right) \tag{3.1}$$

For comparison, misdetection error was simulated and was generated slightly differently than misclassification error (Fig. 3.5 [top]). In this case, false positives did not come from another neuron but were randomly added. In other words, FPs and FNs were added by randomly selecting a certain number of samples and changing the "0"-valued samples to "1" for FP and vice-versa for FN. The stimulus was then reconstructed again with the spike train ρ_1 containing sorting error. Initially, only one type of error was added to each vector, either FP or FN, as done in Fig. 3.9. Then, non-discriminated error was added by performing a logical *not* operation to ε% of the samples, randomly selected (Fig. 3.10). Misdection errors were defined as in Eqn. 3.1. The coding fraction γ was also calculated. Coding fraction is defined as

$$\gamma = 1 - \frac{\epsilon}{\sigma} \tag{3.2}$$

where ϵ is the root mean square error of the estimate and σ is the standard deviation of the stimulus.

3.3.2 Results

The relative effect of spike sorting error on information content of a single-unit spike train was quantified by calculating the changes in mutual information rate. These effects were also related to the changes observed in average firing rate and coding fraction.

Misclassification error was specified as the total FP and FN error, and mutual information was found to decay exponentially, quickly at very low error rates and saturating at roughly 19% error (Fig. 3.7). At percent errors greater than 19%, the

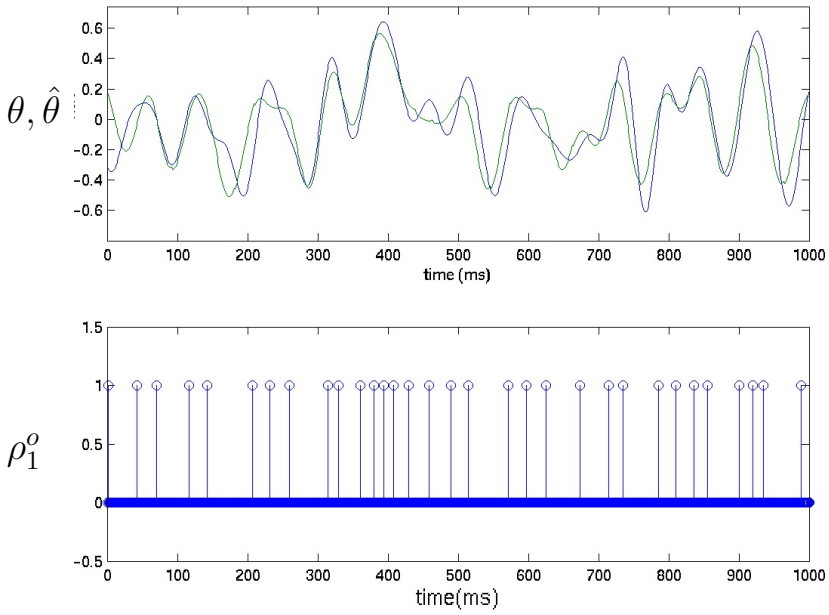


Figure 3.6: Stimulus signal $\theta(t)$ (top, blue) is input into IAF neuron which outputs a spike train. The spike train is contaminated by random sorting error. Output with no error is shown in bottom panel. The spike train is then input into an optimal reconstruction filter to yield stimulus estimate $\hat{\theta}(t)$ (top, green).

coding fraction γ dropped below 0 (Fig. 3.9). Since γ provides a measure of how much of the stimulus variance is accounted for by the estimate variance, the error at which γ becomes less than 0 is the point beyond which spikes act as completely random occurrences, uncorrelated with the stimulus input. Beyond 19% error, the information rate leveled off, and averaged to 10 bits/s. In order to approximate the error that can be tolerated before information content is degraded by $\frac{1}{2}$ of that for no error, an exponential curve was fit to the mutual information rate data (Fig. 3.7). The linear regression on $\ln(\dot{I})$ against percent errors ε yielded $\tilde{I}_{\theta;\rho_1}(\varepsilon) = 14.9e^{-9.0\varepsilon} + 10$. ($\dot{I}$ was adjusted by the asymptotic value of 10 bits/s). According to this equation, $\dot{I}$ is predicted to decrease to half its maximum value at 13.3% misclassification error. Thus, to retain half the information in the original spike train for $\lambda = 32$ spikes/s, no more than 4.3 spikes should be in error per second.

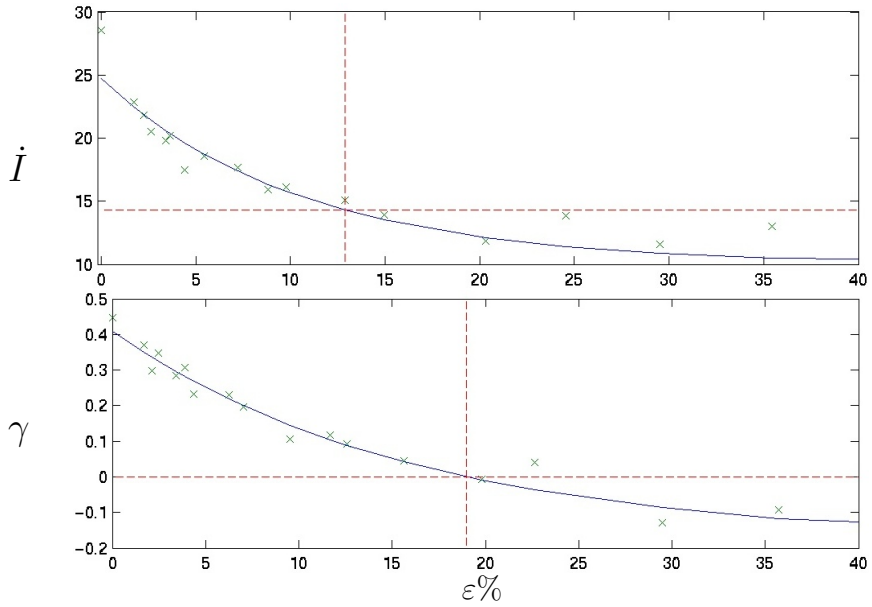


Figure 3.7: Effect of misclassification error on mutual information rate. Exponential curve (solid) fit to data to model $\dot{I}(\theta; \rho_1)$ (*top*) and coding fraction γ (*bottom*) as a function of percent spike sorting error. Dashed lines demarcate where error causes information rate to fall by half its information rate (*top*) and where coding fraction is 0 (*bottom*).

We recognize that the total spike error actually decomposes into two components: detection error and classification error, such that $\varepsilon = \varepsilon_{detect} + \varepsilon_{class}$. Thus, the effect of detection error was determined separately. The information content was again degraded in an exponential fashion. Fig. 3.8 shows the effect of average firing rate on the relationship between information rate and detection error. The average firing rate changed linearly in accord with the error type; i.e., decreasing with false negatives and increasing with false positives. Yet, the information rate changed non-linearly with error and furthermore, decreased in both cases whether false positive or negative. As seen in Fig. 3.8, the effect of spike error on information rate was indistinguishable when the average firing rate changed with the amount of error or was kept constant by adding as many false positives as false negatives, for $\epsilon_{\%} < 50\%$. Fig. 3.9 compares the effect of adding false positive or false negative error. The effects of false positive and false negative error did not significantly differ from each other. Thus, error no longer

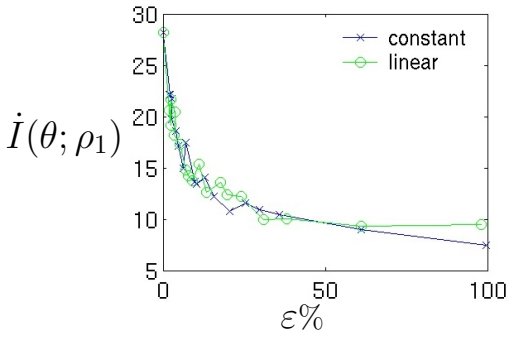


Figure 3.8: Comparing effect of error on information rate when average firing rate is kept constant (x) and when it is changing linearly (∘).

needed to be distinguished as FP or FN, and error was then added non-discriminately to $\rho_1^o(t)$.

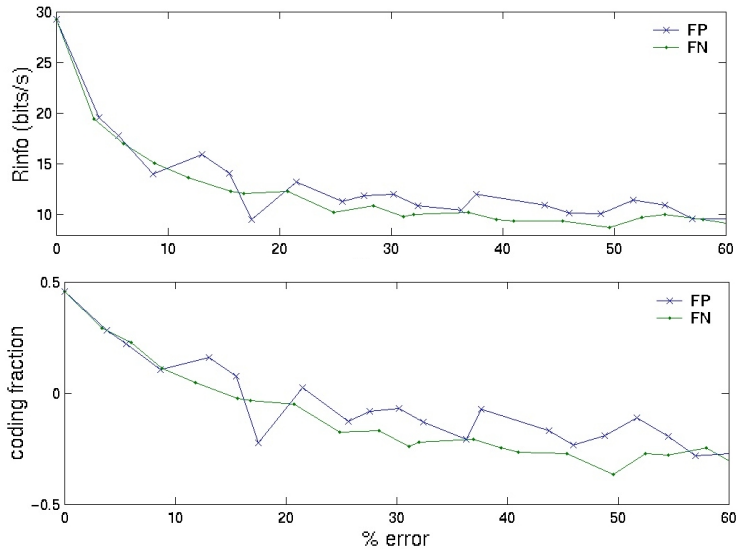


Figure 3.9: Effect of false positive (x) and false negative (·) misdetection error on information rate(top) and coding fraction(bottom).

Information rate as a function of misdetection error was also fit well by an exponential curve (Fig. 3.10). At percent errors greater than 17%, coding fraction γ dropped below 0. The information rate reached half the average maximum value at 10.4% detection error. Thus, the spike train should have no more than 3.3 spikes misdetected per second in order to retain half its original information.

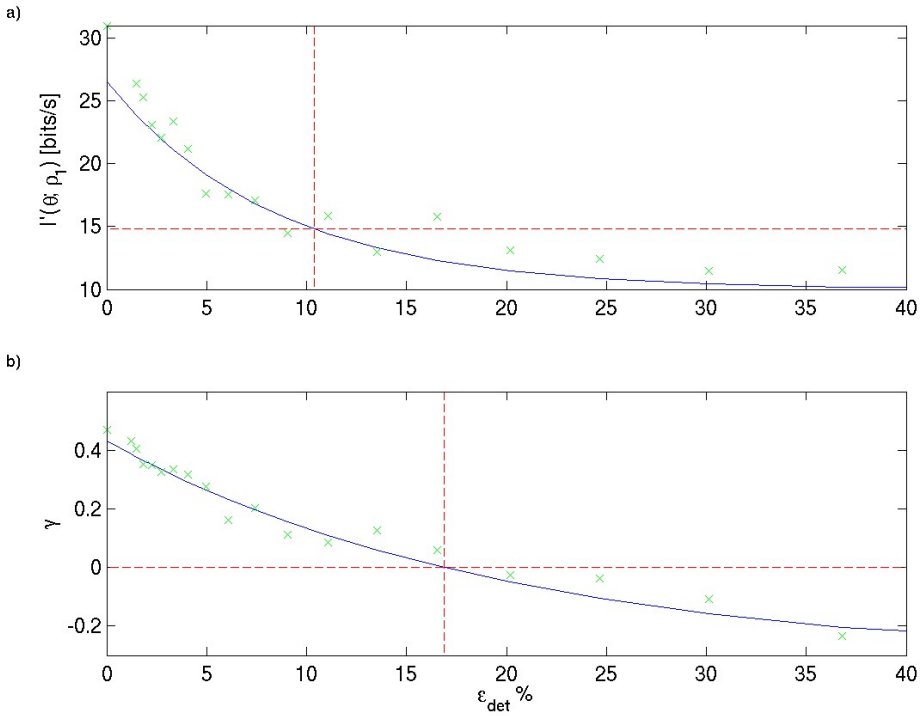


Figure 3.10: Effect of total misdetection error on information rate. Exponential curve (solid) fit to data to model $\dot{I}(\theta; \rho_1)$ as a function of percent misdetection. Dashed line demarcates where error causes information rate to fall by half its information rate (*top*).

In Chapter 5, we treat the effect of detection error and classification error to be approximately equivalent; e.g., a spike missing due to misdetection is considered equivalent to a spike missing due to classifying it in another unit's spike train despite the difference in effect shown here. Misclassification error was shown to have a less severe effect. Thus, the mutual information computed for signals with sorting error in Chap. 5 should be considered a lower bound for the information in sorted signals with error due to spike sorting. If we were to only include one type of error or the other, we would simply determine the proportion of the total error due to that type and consider the same curve as described here and shown in Fig. 3.7. Where we take an interest in the distinction between detection and classification error is in comparing information between sorted signals with both detection and classification error and

non-sorted signals with detection error only. We make this consideration when we compare MI in pooled signals and sorted signals with error later in Sec. 5.1.6 and 5.3.

For now, we note that no reported *in vivo* detection error rates have been found in the literature, and in order to make a conjecture on the range for detection error rates, we would need to consider factors that will be variable bewteen BCI systems, between subjects, and between particular recording sessions. Similar to the illustration given in [19], Fig. 3.11 illustrates the causes of detection error. A higher SNR recording and environment is likely to move the background noise amplitude distribution to the left or make it narrower, so that less of it interferes with the unit amplitudes' distributions. The placement of the electrodes relative to neurons will affect the neuron amplitude distributions. The firing rate will affect the relative probabilities of occurrence between the amplitude of a neural signal and other neural signals as well as the background noise. Thus, without carefully evaluating each of these variables and then making calculations based on these values, we here consider only the total spike error after spike sorting has been performed until we get to section 5.3, at which point we consider detection error to be within a range of percentages of the total spike error.

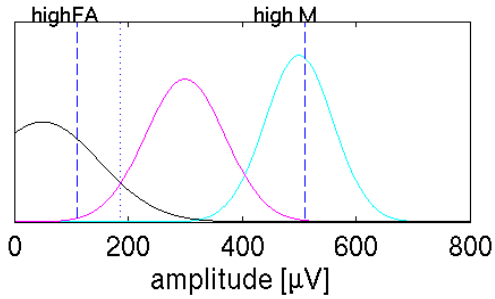


Figure 3.11: An illustration of a hypothetical decision-making process of setting detection thresholds. The probability of peak amplitudes for a large-amplitude neuron (cyan) and a smaller-amplitude neuron (magenta) are illustrated as normally-distributed. The background noise amplitude (black) is shown to be more variable. There are many possibilities for the amount of detection error, both false alarms and misses.

3.3.3 Summary

The aim of this investigation was to quantify the effects of incorrectly discriminated spikes on encoded information so as to determine potential information loss that would be present in a processing path with spike sorting but not without. To address how sorting error affects the information available to a BCI decoder, spike trains were simulated with classification error while changes in the Shannon information content were observed. The results gave insight into tolerable limits on sorting error.

Information rate ($\dot{I}$) decreased as a function of percent spike sorting error in an approximately exponential fashion, both with misclassification error and with misdetection error. Above 2% error, the average decay constant over ten trials was roughly 9.0 for misclassification and 11.9 for misdetection. $\dot{I}$ decayed even more quickly below 2% errors. $\dot{I}$ is expected to decrease to half its maximum value at 13.3% classification error and 10.4% detection error. False positives and false negatives had indistinguishable effects. The average firing rate did not affect the dependence of MI on error. There was no significant difference between $\dot{I}(\varepsilon)$ with constant firing rate and with linearly changing firing rate. Coding fraction γ reached 0 at 19% misclassification error; i.e., by 19% error, a computational decoder on a single-unit response could do no better than random guessing.

Mutual information, $I(\theta; \rho_1)$, between a stimulus signal and the resulting single-unit response was found to decrease exponentially with classification error and random misdetections. This dependence on error was regardless of error being false negative or false positive. On average, 13.3% classification error or 10.4% detection error could be tolerated before the information content dropped to half the information transmitted by the original spike train. Since spike sorters can reasonably be expected to perform with 10% error [20, 18], it is not unreasonable to expect spike sorting to degrade a neuron's response to half its original information content.

We have described the potential causes for spike sorting error and the challenge that implementing an automated spike sorter in a BCI would pose. We have quantified the effect spike error has on information in a neural spike train. We still do not know how to partition the total spike error into detection and classification error. However, these explorations have provided enough motivation to continue pursuing the question of the advantage that spike sorting provides for information processing in a BCI. In the next chapter, we look at the potential of non-sorted spike trains, which could have less spike error, to transmit information.

Chapter 4

A simulation study of information in multi-unit signals

The results in Sec. 3.3 demonstrated the information loss that spike sorting error incurs in a single neuron's spike train. We are aware that using both neurons as separate entities could provide at least as much information as is obtainable from the multi-unit response; however, we do not know how much of that information current decoding algorithms are able to extract from single-unit responses with sorting error. Additionally, we would like to answer the more fundamental question of how much of the information in each of the constituent single-unit signals is retrievable. Hence, we analyze the information content of multi-unit signals.

Simulations were performed to analyze the effects of pooling signals on mutual information. The framework within which these simulations were designed was that of extracellular recordings from a microelectrode of neural spike activity, similar to the information processing path through electrode A, for instance, in Fig. 2.2. In this analysis, we examine the information in ρ_{UA}, first with respect to the information in ρ_k^o, then in comparison to information in ρ_{LLA}. This gives us the picture in Fig. 4.1.

Each electrode detects a superposition of responses from more than one neuron, 2-3 units on average in various cortical and subcortical regions [15, 66]. Typically, these multi-unit responses are sorted into single-unit signals before being used in computational analysis. Two main questions arise from this processing of multi-unit recordings: (1) what information is available in the multi-unit signals and (2) what information can be extracted after a spike sorting process that introduces error? We have already seen the potential information loss spike error can cause in a single unit

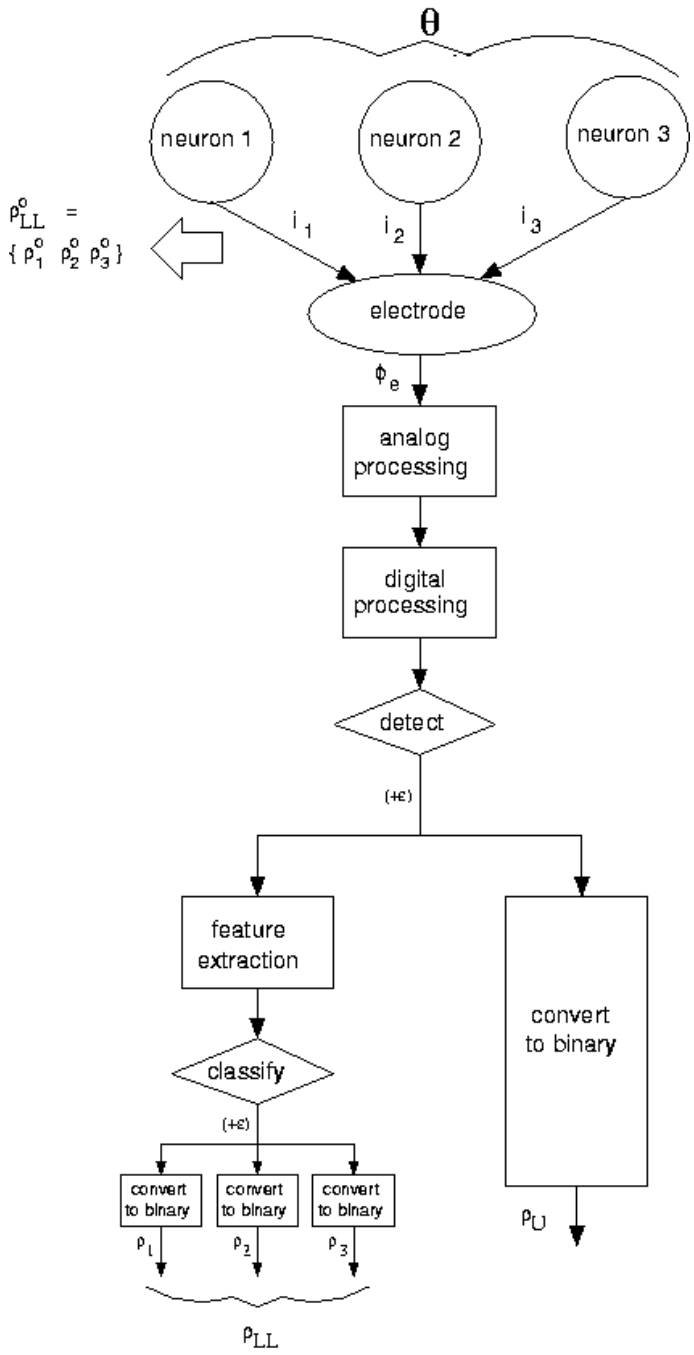


Figure 4.1: Information processing in an extracellular recording, from the target signal to the input of the decoder, for 1 electrode. The analysis in this chapter focuses on the information available in ρ_U for various relationships between ρ^o_1, ρ^o_2, and ρ^o_3. In Sec. 4.4, this information is compared to that in ρ_{LL}.

(section 3.3). Here, we focus on quantifying the available multi-unit information.

In this study, mutual information was calculated using an indirect method. In general, we seek to analyze the multi-unit signal's ability to encode stimuli (or control resultant behavior) and understand how this ability can be affected. In the process, we test the theoretical necessity and benefits of spike sorting a single channel. The responses of individual neurons have been observed to carry information about external stimuli. Neurons output responses that may covary with the stimulus, or may have characteristics that depend on the input stimulus. A summation of these responses would also have characteristic properties that depend on the stimulus. However, details would be lost in the summation which might prevent the multi-unit signals from carrying as much information as the single-unit signals. One question we would like to eventually address is how much information would be lost and whether that loss is worth the savings that would be achieved by eliminating spike sorting. The long-term goal of the studies described in this dissertation is to develop our understanding of the information available to real-time decoders of neural population activity. Here, we attempt to answer two questions: 1) How much attribute-specific information in individual spike trains is lost by pooling the responses? 2) How much of that information is recovered by maintaining the individual responses' identities in labelled line respones? To do so, we calculate Shannon information in simulated multi-unit responses and observe the effects of basic encoding properties of the combined spike trains.

4.1 A model of extracellularly-recorded spike trains from one electrode

We studied the effects of indiscriminately using clusters of two and three units, comparing the indirectly computed mutual information in multi-unit signals to that in

single-unit signals. In doing so, we build up an understanding of the information available to current computational decoding methods on multi-neuron recordings depending on the basic properties of the constituent spike trains.

Multi-unit spike trains were created by superimposing two or three simulated single-unit spike trains. The transmission rate of Shannon information about a particular stimulus, also referred to as attribute-specific information [57, 66], was analyzed. An integrate-and-fire neuron generated a stochastic single-unit neural response $\rho(t)$, as performed in the spike sorting error analysis (Section 3.3.1). Average firing rate was adjusted by varying the mean threshold voltage. A Gaussian signal filtered at 20Hz simulated the stimulus $\theta(t)$ into the neuron. To obtain a lower bound on the rate at which Shannon information about the stimulus was encoded in $\rho(t)$, an estimate of the stimulus, $\theta(t)$, was reconstructed to produce $\hat{\theta}$, and the mutual information rate between θ and ρ_1, $\dot{I}(\theta;\rho_1)$, was calculated [51, 49, 50]. The optimal linear estimate $\hat{\theta}$ of the stimulus was computed by applying the Wiener-Hopf filter to $\rho(t)$ [67]. From the error between θ and $\hat{\theta}$, the signal-to-noise ratio (SNR) was computed, and $\dot{I}(\theta;\rho_1)$ was calculated according to equation 2.13. Each trial was defined by the creation of a unique 100-second duration stimulus signal. Ten trials were performed, with each trial consisting of multiple presentations of the stimulus. Every presentation of the stimulus generated a unique spike train.

Four different cases (A-D) were considered in which the activity of two and three neurons are detected on the same electrode. The union of the individual units' responses was represented by the multi-unit signal ρ_{UX}, where X = A, B, C, or D indicates to which of the four cases the multi-unit signal belongs. Each of the individual spike trains were binary signals, making ρ_{UX} a binary signal which masked the overlap of spikes occurring within the same millisecond. The four combinations are shown schematically in Fig. 4.2 and described as follows:

A) Unit 2 independent of the stimulus. Two spike trains responding to different stimuli were generated, ρ_1 responding to θ_1 and ρ_2 responding to an independent stimulus θ_2 (Fig. 4.2A). ρ_1 and ρ_2 had mean spike rates within 5% of each other. The union ρ_{UA} simulated a multi-unit recording of two unrelated, uncorrelated neurons. For each trial, a unique θ_2 was presented to unit 2 ten times to create ten different unit 2 responses.

B) Unit 2 completely dependent on the stimulus. ρ_2 was generated from a second presentation of θ_1; i.e, ρ_2 was a different observation of the same random point process as unit 1 encoding θ_1. The output ρ_2 was combined with ρ_1 to produce multi-unit signal ρ_{UB} (Fig. 1B). As in case A, 10 different unit 2 responses were generated for each trial.

C) Unit 2 partially dependent on the stimulus. To look at the intermediate case

between complete dependence and independence of ρ_2 on θ_1, ρ_1 was combined with ρ_{2C}, a corrupted version of ρ_2 from case B; thus, ρ_{2C} responds to θ_1 but carries less information about the stimulus than ρ_{2B}. We have observed that random addition and deletion of spikes will decrease the mutual dependence of the spike train on the stimulus (Sec. 3.3); therefore, the spike train ρ_{2C} was created by adding random spikes to and deleting original spikes from ρ_2 (Fig. 4.2C) while maintaining a constant average firing rate. Error was added until the attribute-specific information $\dot{I}(\theta_1; \rho_{2C})$ reached a minimum beyond which adding random error did not decrease the information; we later define this point as the baseline level.

D) Both neurons partially dependent on the stimulus. The final case looked at a multi-unit signal comprised of two spike trains both responding to θ_1 but

decreasing similarly in their dependence on θ_1 (Fig. 4.2D).

Signal ρ_1 is present in ρ_{UA}, ρ_{UB}, and ρ_{UC} but is corrupted to create ρ_{1D} for case D. This signal, ρ_1, is the ideal single-unit spike train with maximal information content about stimulus θ_1. The relation of unit 2 to the stimulus θ_1 and to unit 1 varies in each of the four cases, as described. One time segment of the spike trains from each case is displayed in Fig. 4.2E. Corresponding 3-unit cases were simulated by adding a third unit which behaved like unit 2. We considered cluster sizes of two and three only because the number of neurons discriminated on the same electrode is most often two or three [15, 66].

Information about θ_1 in ρ_2 and ρ_{UX}, was calculated relative to information in ρ_1. The ratio of $\dot{I}(\theta_1;\rho_2)$ to $\dot{I}(\theta_1;\rho_1)$, is denoted by $\dot{I}_2/\dot{I}_1$. The relative change in information in ρ_{UX} compared to ρ_σ was calculated as

$$\Delta\dot{I}_r = \frac{\dot{I}(\theta_1;\rho_{UX}) - \dot{I}(\theta_1;\rho_\sigma)}{\dot{I}(\theta_1;\rho_\sigma)} \tag{4.1}$$

where σ here indicates the single-unit response to which the multi-unit response is being compared (1, 2A, 2B, 2C, 1D, or 2D). Although the stimulus reconstruction method has been accepted as a method to study information in neural spike trains [51, 49, 50], one caveat of the information estimate derived from a Gaussian channel assumption is that there is strictly a non-zero dependence of ρ on θ, unless $\theta = 0$ or the gain G of the system from the input θ to the output ρ equals 0. When ρ is independent of θ, mutual information should equal 0 as the length of the signals observed approaches infinity; hence, the true gain must be 0. The measured gain was estimated to be very small under this condition but not exactly equal to 0, due to random correlations between θ and ρ which results in a non-zero Wiener-Hopf filter. The consequence of $G \neq 0$ is an erroneously finite $P_{neff}(f)$, resulting in the calculated

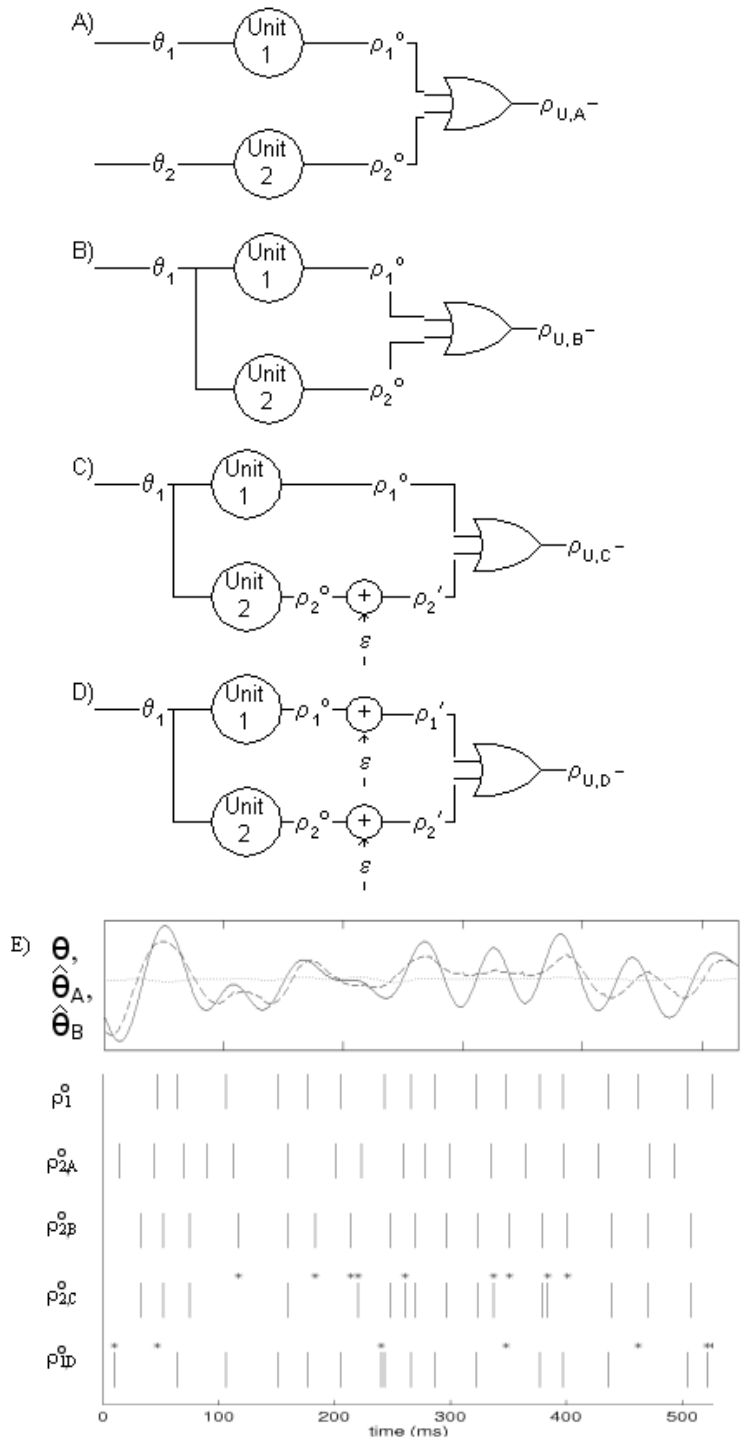


Figure 4.2: Generation of four types of multi-unit signals. A) A spike train independent of the stimulus is added. B) Added spike train is the same as ρ_1 in its encoding but uncorrelated. C) Spike train with less correlation to stimulus is added to ρ_1. D) Both units in multi-unit signal are similarly less correlated with stimulus. E) Examples of spike trains used in each of the four cases. Asterisks indicate spikes added or removed in cases C and D to decrease correlation between the two combined units. Upper panel shows original stimulus s, and reconstructions from ρ_{2A} (dotted) and ρ_{2B} (dashed).

SNR and information to be greater than 0 even in the case of unrelated θ and ρ. In order to be able to speak of information on a more intuitive scale, information rates were normalized to range from 0 at random guessing to 100% at the maximal single-unit information rate, $\dot{I}(\theta_1; \rho_1)$. For this purpose, the baseline level of information was defined to account for random correlations between finite-length signals and the non-negative nature of SNR. The baseline was defined as the average information rate for which the coding fraction = 0. Coding fraction γ is a measure of the variability in the stimulus which is encoded in the variability of the output and should equal 0 for a random-guess estimate; it is the fraction of the standard deviation of the stimulus which is not accounted for by the root mean square error of the stimulus estimate:

$$\gamma = 1 - \frac{\frac{1}{N}\sqrt{\Sigma(\hat{\theta} - \theta)^2}}{\sigma_\theta} \tag{4.2}$$

The subscript bl, for instance, in Fig. 4.5, denotes that information rates were adjusted by the baseline levels of information before calculations were carried out. Ten trials were performed, with each trial involving the generation of a unique signal for θ_1 and/or θ_2. Similar calculations were performed for all four cases.

The amount of attribute-specific information in a pair of responses was then calculated in a labelled-line approach. The stimulus was reconstructed from a multivariate optimal linear decoder operating on labelled-line 2-unit responses as opposed to one pooled multi-unit response; hence, one filter was constructed for the pair, but each unit was considered a separate entity. The optimal filter for the 2-unit cluster still

satisfied the Wiener-Hopf equations for a one-dimensional response [65]:

$$C^{\bar{\rho}\theta} = h \cdot C^{\bar{\rho}\bar{\rho}} \tag{4.3}$$

$$C^{\bar{\rho}\theta} = \bar{\rho}^T \theta \tag{4.4}$$

$$C^{\bar{\rho}\bar{\rho}} = \bar{\rho}^T \bar{\rho} \tag{4.5}$$

where C is the specified covariance matrix and $\bar{\rho}$ is the Toeplitz response matrix comprised of one row for each time lag in the filter. Thus, the j^{th} row of $\bar{\rho}$ holds the response $\rho(t)$, for $t = j, j+1, ...L+j$ for a filter with L time lags. To compute a filter for a labelled-line response instead of the pooled multi-unit response, the response matrix $\bar{\rho}$ was expanded for two units by concatenating the response matrix of each unit. The filter and the estimate of θ were created following the methods detailed in [55] with the exception that the filter used here had negative as well as positive time lags. The mutual information was then calculated from the labelled-line estimate using Eqn. 2.13 and compared to the information in pooled responses.

A note about notation: In this chapter, a capital letter A-D in the subscript indicates to which case the signal belongs. Thus, $I_{UD} \equiv I(\theta; \rho_{UD})$, for example, refers to the mutual information in the multi-unit signal of case D, which is the union of two signals, each of which are corrupted by noise. $I_{2A} \equiv I(\theta; \rho_{2A})$ would refer to MI in the corrupting signal in case A, which is independent of θ_1.

4.2 Mutual information a pooled signal

Here, we analyze the mutual information in the pooled response in the 4 different scenarios outlined in section 4.1. Results from the two-unit scenarios are described first, followed by an extension of the 2-unit cases to 3-unit cases.

4.2.1 Independent unit

A spike train was derived directly from the stimulus θ_1, with $\dot{I}(\theta_1;\rho_1) = 28.6$ bits/s. When ρ_2 was generated in response to a stimulus independent of θ_1, $\dot{I}_{bl}(\theta_1;\rho_{2A}) = 0$ by definition. When unit 1's response was contaminated by a response ρ_2 which was unrelated to θ_1, the multi- unit response ρ_{UA} transmitted 28% of the $\dot{I}_{bl}(\theta_1,\rho_1)$. The outcome of a presentation of θ_2 in a typical trial is shown in Fig. 4.2. Results showed that differences among trials at a 5% significance level were not significant. All significant testing between three or more groups was done using one-way ANOVA; for two groups, a two-tailed paired t-test was used unless otherwise specified.

Unit 2 by itself transmitted approximately 20% of the information about θ_1 transmitted by unit 1 (Fig. 4.3A). When unit 1's response to stimulus θ_1 was contaminated by a response $\rho_{2,A}$ which was unrelated to θ_1, the multi-unit response ρ_{UA} transmitted only 43% of the information in ρ_1. The mean of the change in information from ρ_1 to ρ_{UA} was -0.43, indicating a 57% loss of information.

Fig. 4.3A shows the mean information rates and standard deviations across all trials and stimulus presentations. Adding the second independent neuron decreased the mean information rate by 57%; the third independent neuron decreased the information by another 13%.

4.2.2 Co-dependent units

Because of the stochastic nature of neurons, different neurons may not respond identically even if they are responding to the same stimulus with the same encoding scheme. Likewise, the same stimulus presented to the integrate-and-fire neuron used here will not yield an identical response on every stimulus presentation. For scenario B, the outputs of separate stimulus presentations simulated the outputs of neurons distinct from ρ_1 but co-dependent on θ_1 with ρ_1.

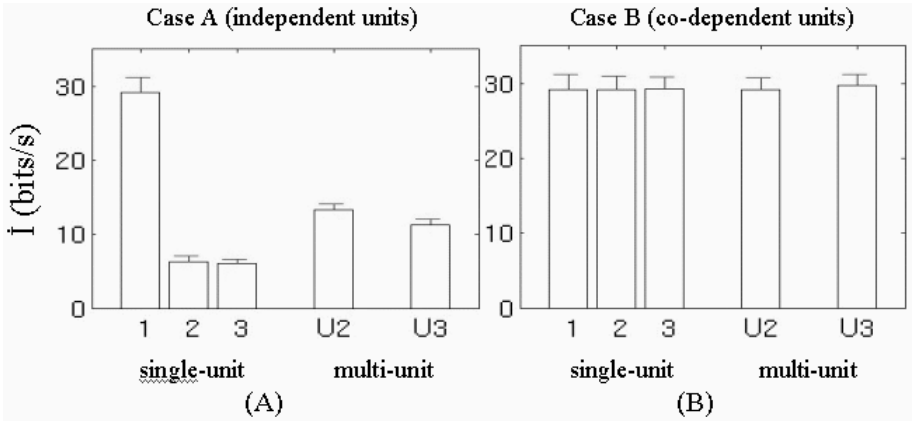


Figure 4.3: Average information rate and standard deviation of original single-unit spike train (1) and of the added spike trains (2, 3) are shown alongside that of the multi-unit signals comprised of 2 (U2) and 3 (U3) responses. A (left): Case A - Units 2 and 3 are independent of s, and therefore have low mutual information. Contaminating ρ_1 with the other spike trains that carry information independent of θ_1 decreases the mutual information about θ_1. B (right): Case B - When the spike train is combined with spike trains carrying insignificantly different information, the multi-unit spike train also maintains the same information rates. Average firing rate of all single units is 32 ± 0.1 spikes/s, of 2-unit signals is 64 ± 0.2 spikes/s, and of 3-unit signals is 95 ± 0.4.

Despite the variability in spike timing in case B (Fig. 4.2E), the information rates of ρ_1 and ρ_2, $\dot{I}(\theta_1; \rho_1)$ and $\dot{I}(\theta_1; \rho_2)$ showed non-significant differences (paired t-test, $p = .25$). It was found that combining these signals also caused a non-significant difference in information rate; the average $\Delta\dot{I}_r$ over the 10 trials was less than a 1% gain ($p = .80$, 95% confidence interval: $-.015 < \mu_{\Delta\dot{I}_r} < .031$). Fig. 4.3B shows the mean and standard deviation of the information rates of the multi-unit and constituent single-unit signals. The 2-unit response carried the same information as the mean constituent single-unit response (mean percent change $< 1\%$, $p = .80$). The information rate of the 3-unit response did increase significantly but only by 2.1% ($p = .01$).

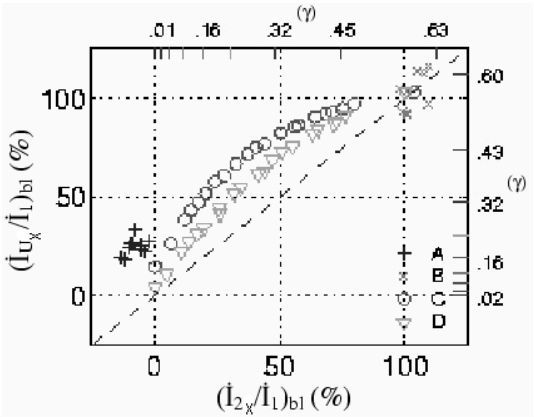


Figure 4.4: Comparison of relative information rates of multi-unit signals to that of corresponding single-unit signals ρ_2 for each case X = A, B, C, or D, as indicated in the legend. Normalization rate $\dot{I}(\theta_1;\rho_1) = 29$ bits/s. Normalized relative multi-unit information rate, $(\dot{I}_{UX}/\dot{I}_1)_{bl}$, on vertical axis plotted against normalized relative unit 2 information rate, $(\dot{I}_{2X}/\dot{I}_1)_{bl}$; X indicated in the legend. Corresponding coding fraction of ρ_U are given for reference on the right axis, and coding fraction of ρ_2 is shown on upper axis.

4.2.3 Varying dependence of one unit (sub-optimally encoded neuron)

Random spike error was added to ρ_2 to decrease the mutual information between ρ_2 and θ_1. The open circles in Fig. 4.4 show how much of the information in ρ_1 was retained when combined with $\rho_2\prime$ as it decreased in its dependence on θ_1. Information was normalized to $\dot{I}(\theta_1;\rho_1)$ because ρ_1 has maximal information. The dashed line indicates the values that would be obtained if the multi-unit signal, in this case ρ_{UC}, carried as much information as unit 2's response, in this case ρ_2'.

For case C, spike error was added to unit 2's response, which was transmitting as much information as unit 1, until $\rho_2\prime$ reached baseline information rates. As the relative information rate ρ_2/ρ_1 decreased from 100% to 0, the relative information rate of ρ_{UC} decreased in a logarithmic fashion from 100% to 20% (Fig. 4.4). Thus, information in ρ_{UC} decreased more slowly than in ρ_2. Multi-unit signal ρ_{UC} retained 23% of the information even when the contaminating spike train was reduced to

baseline information levels. ρ_{UC} was still at 83% when ρ_2 had half the information. If ρ_2 had been identical to ρ_1, then ρ_{UC} would exactly equal $\rho_2\prime$, and the multi-unit information would exactly follow the dashed line. However, when spike trains were combined whose spike timing showed slight variation, the information rate increased.

Fig. 4.5 (dotted curve) also shows the increase in information in the two-unit signal over the corrupted single-unit spike train. When the third unit was added (Fig. 4.5A, x's), the attribute-specific information did not change significantly even though the number of corrupting spikes had increased.

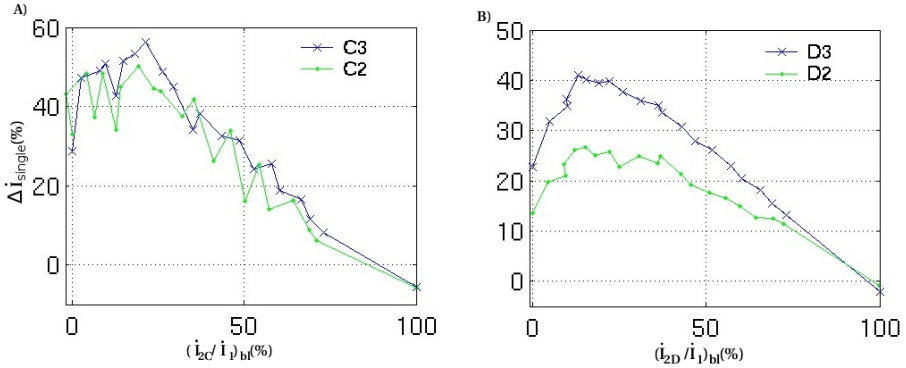


Figure 4.5: Effect of superimposing 2 and 3 neural responses on information content as a function of average normalized information rate of the corrupted single-unit responses ($(\dot{I}_2/\dot{I}_1)_{bl}$). Relative difference between multi-unit and single-unit information rates ($\Delta\dot{I}_{single}(\%)$) is plotted as a function of the normalized information rates of corrupted single-unit responses, relative to the maximum single-unit information rate **A)** Case C. Positive percent change in information rate with respect to $(\dot{I}_2/\dot{I}_1)_{bl}$ indicates an increase in information over the corrupted single-unit responses as 2 and 3 single unit responses are combined. This improvement increases as the single-unit responses become more corrupted until more improvement cannot be made because of the extent of corruption. Adding the third unit decreases the variability in information. **B)** Case D. Information improves over the average single-unit response as more units are combined.

4.2.4 Varying dependence of both units

In this scenario, neither neuron was completely dependent on θ_1. The relationship between $\dot{I}(\theta_1;\rho_{UD})$ and $\dot{I}(\theta_1;\rho_{2D})$ is shown in Figure 4. The mean difference between $\dot{I}(\theta_1;\rho_1\prime)$ and $\dot{I}(\theta_1;\rho_2\prime)$ was 0.30 bits/s ($p = .30$, two-tailed paired t-test). The pattern

of information decay was similar to that of case C, but as expected, ρ_{UD} had less information than ρ_{UC}, in which only one of the two neurons was sub-optimal. Fig. 4.6 plots the percent difference between information in ρ_U and unit 1's response against the percent difference between information in ρ_U and unit 2's response. This shows the change in information due to combining the two single-unit signals from the perspective of each of the individual units. The horizontal dashed line indicates where the multi-unit signal carries as much information as unit 1 and the vertical line where ρ_U carries as much as unit 2.

The combined signal in case D transmitted more information than either of the individual sub-optimally related neurons, $\rho_1\prime$ and $\rho_2\prime$, as shown by the location of $\rho_{UD}\prime$'s markers in the first quadrant. Data points are not shown for order-of-magnitude information gain. Such large gain occurs as $\rho_1\prime$ and $\rho_2\prime$ become less associated with θ_1 and approach baseline information levels. Quadrant IV holds the instances in which the multi-unit signal has greater information than unit 2 but less than unit 1. This was the situation for ρ_{UC} and ρ_{UA}. In case A, the multi-unit signal obviously carries less information than the ideal spike train. Adding ρ_2 can also be viewed as adding on average 97% error to ρ_1; yet, ρ_{UA} retains 20% of the information (mean $\Delta\dot{I}_{1A,bl}$ = -79.7%). Furthermore, it has a 200-300% increase in $\dot{I}_{bl}$ over unit 2. The cluster of x's at the origin for case B indicates negligible change in information with respect to either single unit which both have the optimal attribute-specific information.

When comparing the 3-unit information rates $\dot{I}_{UX,3}$ to the single-unit information rates of ρ_1 and ρ_3, the results are analogous to the 2-unit case. Although the details differ, the ordinal relationship between the 3-unit and single-unit information rates are the same as in the 2-unit cases. Thus, the 3-unit cases yield a plot similar to Fig. 4.6.

When the single-unit spike trains have similar rates of information about the

stimulus, the combined signal contains more information than either of the single-unit spike trains alone. Fig. 4.6B shows that as the average single-unit information $\dot{I}_{single,bl}$ decreases from 100% to about 20%, the two-unit information rate increases by more than 25%. As $\dot{I}_{single,bl}$ decreases further, the improvement in ρ_{UD}'s information rate over the single-unit information is reduced. The information rates improve to a greater degree in the 3-unit case, reaching a maximum increase of 41% when $\dot{I}_{single,bl}$ is 20%.

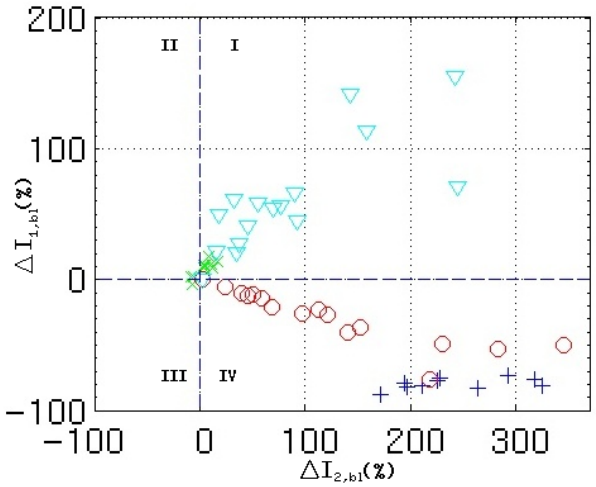


Figure 4.6: Percent change in information rate between the multi-unit signal and each of the constituent single-unit signals. The normalized percent difference between $\dot{I}_{bl}(\theta;\rho_U)$ and $\dot{I}_{bl}(\theta;\rho_1)$ ($\Delta\dot{I}_{1,bl}$) and the normalized percent difference between $\dot{I}_{bl}(\theta;\rho_U)$ and $\dot{I}_{bl}(\theta;\rho_2)$ for the 4 different cases, as indicated by the different cases (A = blue '+', B = green '×', C = red 'o', D = cyan '$\bigtriangledown$'). Data points are not shown for information gain greater than 10. Such large increases occur as $\dot{I}_{bl}(\theta_1;\rho_{2C})$, $\dot{I}_{bl}(\theta_1;\rho_{1D})$, and $\dot{I}_{bl}(\theta_1;\rho_{2D})$ approach 0.

4.3 Characterizing multi-unit information

Multi-unit recordings were simulated as the union of two and three units' outputs, and the mutual information between the responses and stimulus was analyzed. Four different types of simulations were carried out to model scenarios in which the information the individual neurons transmit about the stimulus differs. The 3-unit

responses were an extension of the 2-unit responses which can be summarized in the following:

A) One neuron's response was directly derived from the stimulus and had maximal information ($I_{bl}(\theta_1;\rho_1) = 100\%$). The second neuron's response, which was completely independent of θ_1, had no attribute-specific information ($\dot{I}_{bl}(\theta_1;\rho_2) = 0$).

B) Both single-unit spike trains were derived directly from the stimulus; i.e., both had complete mutual dependence on the stimulus. Hence, both neurons transmitted roughly the same information ($\dot{I}_{1,bl} \approx \dot{I}_{2,bl} \approx 100\%$).

C) Both neurons responded to θ_1 but the second was only indirectly related to it ($\dot{I}_{1,bl} \approx 100\%, 0 < \dot{I}_{2,bl} < 100\%$).

D) Both neurons had about the same information, as both were related to θ_1; however, neither had maximal information about θ_1 ($0 < \dot{I}_{1,bl} \approx \dot{I}_{2,bl} < 100\%$).

Four general outcomes are possible in the relation of the multi-unit information to the single-unit information:

1. $\dot{I}(\theta_1;\rho_U) < \dot{I}(\theta_1;\rho_1)$, $\dot{I}(\theta_1;\rho_U) < \dot{I}(\theta_1;\rho_2)$. Some information in both neurons was lost, leading to data points in quadrant III of Fig. 4.6. Adding another signal caused unit 1 to lose all its information about θ_1 because ρ_1 could no longer be distinguished from other spikes in ρ_U.

2. $\dot{I}(\theta_1;\rho_U) < \dot{I}(\theta_1;\rho_1)$, $\dot{I}(\theta_1;\rho_U) \approx \dot{I}(\theta_1;\rho_2)$ The multi-unit signal retained the information in one of the single-unit signals but lost information in the other. The data could lie along the horizontal axis between quadrants II and III, or along the vertical axis bordering III and IV. However, since the scenarios were designed such that unit 1 had at least as much information as unit 2, only the latter situation could occur.

3. $\dot{I}(\theta_1;\rho_U) \approx \dot{I}(\theta_1;\rho_1) \approx \dot{I}s_1;\rho_2)$ The multi-unit output carried as much information as both of the single-unit responses, which led to a cluster of data points at the origin.

4. $\dot{I}(\theta_1;\rho_2) < \dot{I}(\theta_1;\rho_U) < \dot{I}(\theta_1;\rho_1)$ The multi-unit response improved information compared to unit 2 but not to unit 1 (quadrant IV).

5. $\dot{I}(\theta_1;\rho_U) > \dot{I}(\theta_1;\rho_1), \dot{I}(\theta_1;\rho_U) > \dot{I}(\theta_1;\rho_2)$ The multi-unit signal transmitted more attribute-specific information than both single-unit signals (quadrant I).

Stimulus θ_1 was considered relevant to a particular task. Units 1 and 2 were considered as neighboring neurons whose extracellular potentials would be detected on the same microelectrode. In case A, even though the spike trains were completely uncorrelated and independent, the multi-unit signal did not lose all the information. It lost 80% of the information with respect to ρ_1 but gained information with respect to the independent neuron and remained above baseline levels.

Although the decoder in this scenario was able to extract a little over 20% of the information in ρ_1, we do not know how much better a decoder may with the multiple single-unit signals. ρ_2 would still be contaminating the population response even if discriminated single-unit responses were used, but perhaps a linear decoder can mitigate the effects of such a neuron if it is kept labelled. What needs to be tested is whether a population decoder could extract as much information from an ensemble of multi-unit signals as it could from an ensemble of spike-sorted responses if catered to multi-unit or single-unit input, respectively.

4.4 Comparing labelled-line and pooled information

At this point, we do not know how much information could be obtained by keeping the individual spike trains separate, but we know that using both neurons as separate entities without error could provide at least as much information as is obtainable from the multi-unit response. Although a pooled multi-unit response can provide more attribute-specific information than either of the two units alone, the results in this section show that even greater information is available in two perfectly discriminated signals considered jointly in a labelled-line response. Fig. 4.7 shows reconstructions of the stimulus from the union of two units' responses ($\hat{\theta}_U$) and from the labelled line response ($\hat{\theta}_{LL}$). The labelled line estimate tracked the stimulus (θ) much better than the pooled estimate when the units were independent (case A, Fig. 4.7A) and had less reconstruction error. When the units were co-dependent (case B), the two estimates tracked the stimulus with similar accuracy most of the time, but the pooled estimate occasionally had much larger error (Fig. 4.7B). In case A, the mean information in $\hat{\theta}_{LL}$ was greater than that in $\hat{\theta}_U$ by a factor of 2.2 (Fig. 4.8A). In case B, the information in $\hat{\theta}_{LL}$ was a factor of 1.2 times that in $\hat{\theta}_U$ (Fig. 4.8B).

Likewise, for case C, in which the two units had dissimilar information, the labelled line response transmitted significantly more information than the pooled response (Fig. 4.9A). The labelled line and pooled responses had much closer information rates in case D, for which the two constituent units also transmitted similar information (Fig. 4.9B). The labelled line information in case C decreased with relative linearity as the information in ρ_{2C} decreased; when the responses were pooled, the information decreased more rapidly. The pooled response could not achieve much more than 100% of the information in a single-unit response, whereas the labelled-line response was able to transmit as much as 30% more information than the single-unit

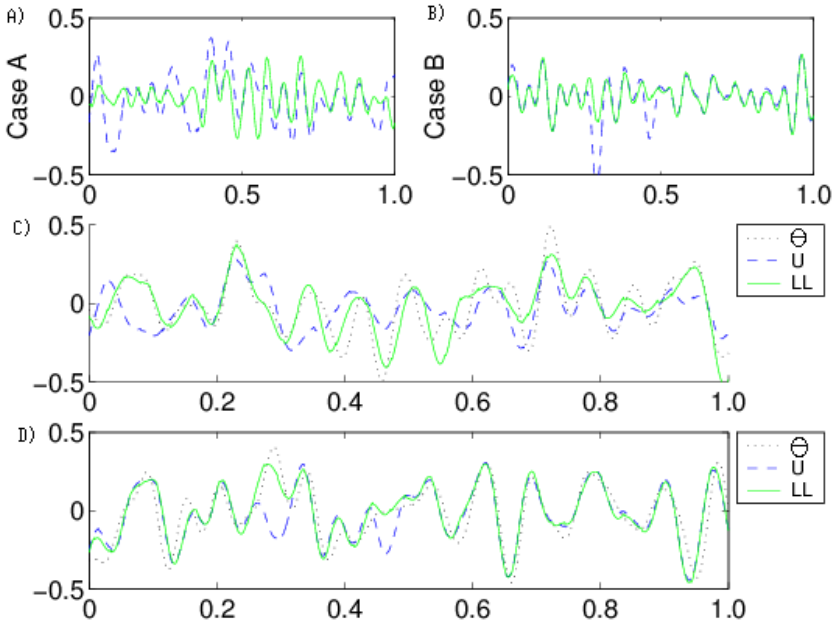


Figure 4.7: Reconstruction of the stimulus from two units' responses (subplots C and D), using the pooled and labelled line approach, when units are independent (C) and when they are co-dependent (D). θ (dotted): the original stimulus. U (dashed): $\hat{\theta}_U$ reconstructed from the union of the two responses. LL (solid): $\hat{\theta}_{LL}$ labelled line reconstruction of θ. A) reconstruction error from the estimate for case A (corresponding to subfigure C). B) reconstruction error from the estimate for case B (corresponding to subfigure D).

response ($\dot{I}(\theta_1; \rho_{U1})$) and transmitted 100% of the relative information when unit 2's information, $(\dot{I}_{2C}/\dot{I}_1)_{bl}$, was 0. In case D, $\dot{I}_U$ and $\dot{I}_{LL}$ increased together as the average single-unit information increased to 50%; beyond 50%, I_U increased with a lower slope.

This comparison of $\dot{I}_U$ to $\dot{I}_{LL}$ as the single-unit information increases demonstrates that, as expected, using both neurons as separate entities without discrimination error provides at least as much information as is obtainable from the pooled multi-unit response. However, the amount by which $\dot{I}_{LL}$ exceeds $\dot{I}_U$ varies from case C to case D, and neither $\dot{I}_U$ nor $\dot{I}_{LL}$ increase in the same manner in case C as in case D. Thus, we considered the factor that distinguishes case C from case D - the difference in the information rates of the constituent single-unit signals.

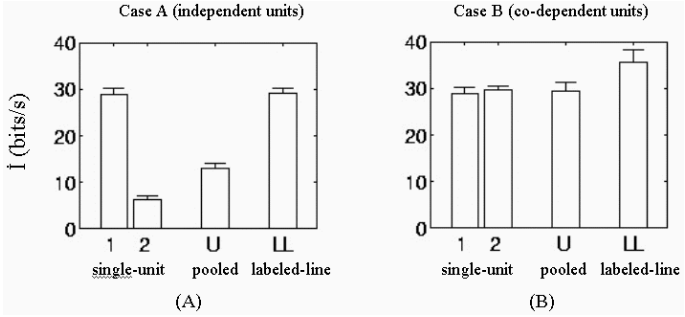


Figure 4.8: Average information rates and standard deviations of the two single-unit spike trains ρ_1 (1) and ρ_2 (2) are shown alongside those of the pooled signals (U) and labelled-line (LL) responses. A (left): Case A - Unit 2 is independent of θ. ρ_{LL} is able to maintain all the information in unit 1. Pooling the responses causes significant loss of that information. B (right): Case B - When the spike trains combined carry similar information, the pooled response can maintain that information. A labelled-line response can extract even more information about θ.

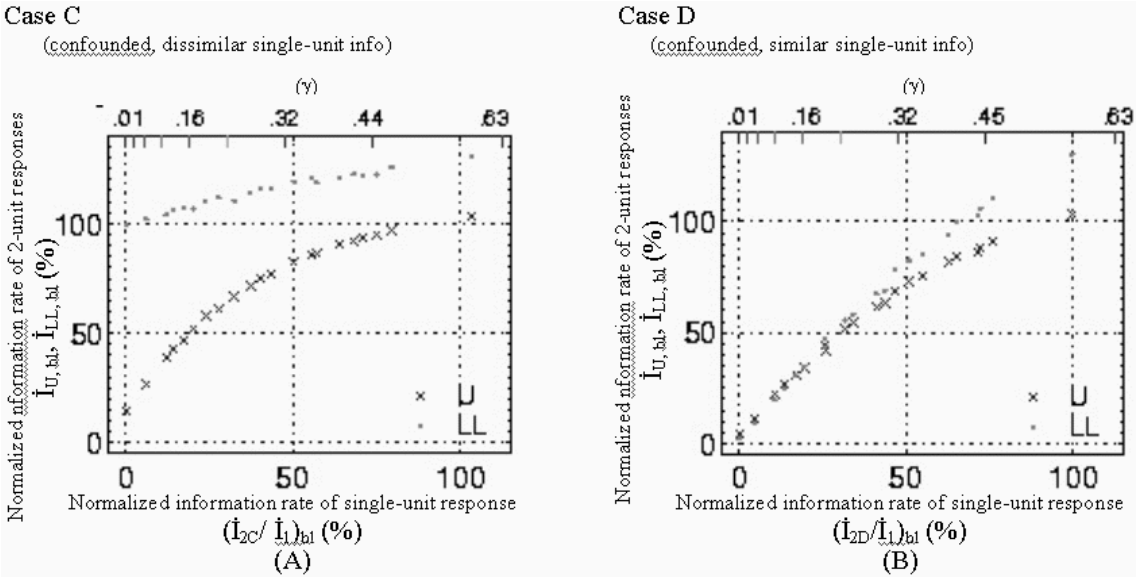


Figure 4.9: Normalized information rates of pooled responses ($\dot{I}_{U,bl}$) relative to maximum single-unit information and labelled-line ($\dot{I}_{LL,bl}$) responses versus normalized information rate of constituent single-unit responses. A (left): Information of the pooled and labelled-line responses in case C as unit 2 is increasingly corrupted. B (right): Information of the 2-unit responses in case D as unit 1 and unit 2 are increasingly corrupted. Corresponding coding fraction of ρ_2 is shown on upper axis.

The differences between $\dot{I}(\theta_1;\rho_{1D})$ and $\dot{I}(\theta_1;\rho_{2D})$ were randomly distributed about 0. The difference $\dot{I}_{LLD}$ -$\dot{I}_{UD}$ did not depend on these differences in information between the constituent units (Fig. 4.10) but rather on the average constituent single-unit information. From Fig. 4.9B, it can be seen that $\dot{I}_{LLD}$ -$\dot{I}_{UD}$ increased as $\dot{I}_2/\dot{I}_1$ increased. In contrast, the difference between $\dot{I}(\theta_1;\rho_{1C})$ and $\dot{I}(\theta_1;\rho_{2C})$ was systematically increased. As this difference increased, the difference between labelled line and pooled multi-unit signals increased monotonically in an exponential fashion (Fig. 4.10).

Information in a labelled line response ($\dot{I}(\theta_1;\rho_{LL})$) is in general greater than that in a pooled response ($\dot{I}(\theta_1;\rho_U)$). However, in our simulations, the information gained by using discriminated unit responses varied for the different cases. The gain in information due to using a labelled line instead of a pooled response was greater overall for cases A and C (27 - 84% increase) than in cases B and D (-2.6 - 27%). Fig. 4.9 shows that the reason the labelled line extracts even more information in case C than D is not related to the total amount of information available in the sum of the individual units' information. The gain $\dot{I}(\theta_1;\rho_{LL}) - \dot{I}(\theta_1;\rho_U)$ increased as the sum of single-unit information in case D, $\dot{I}(\theta_1;\rho_{1D}) + \dot{I}(\theta_1;\rho_{2D})$, increased, but actually decreased as the sum in case C, $\dot{I}(\theta_1;\rho_1) + \dot{I}(\theta_1;\rho_{2C})$, increased.

Fig. 4.10 indicates that the labelled line gain over pooled responses is due to dissimilarity between the two constituent units. The difference in gain between cases C and D can be explained by the units having disparate information in case C and similar information in case D. The variability between spike trains in case D was due to random noise; thus, differences between unit 2 and 1 did not systematically encode information about θ. However, when the variability was due to systematic differences in information between the units, keeping the units separate allowed more attribute-specific information to be extracted from the cluster, and summing the responses was

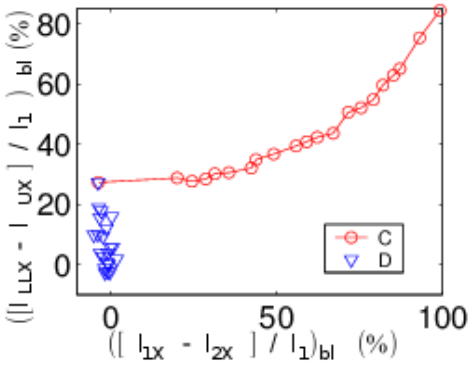


Figure 4.10: The difference in information rates of the labelled-line and pooled responses, $([\dot{I}_{LLX} - \dot{I}_{UX}]/\dot{I}_1)_{bl}$, as a function of relative difference between unit 2 and unit 1's information rates, $([\dot{I}_{1X} - \dot{I}_{2X}]/\dot{I}_1)_{bl}$. Case C (circles) - As differences between unit 1 and unit 2 information increase gradually, differences between 2-unit response information rates increase exponentially. Case D (triangles) - Differences between unit 1 and unit 2 information is clustered around 0. Differences in the 2-unit response information varies from 0 to 20%.

no longer comparable to the labelled line approach.

4.5 Summary and limitations

We have seen that more attribute-specific information can be extracted by pooling the responses than using any one of the responses alone. However, attribute-specific information in an individual unit's spike train can be degraded in a multi-unit signal that pools its response with a unit that carries disparate information. A labelled line response is more robust than a pooled response against such corruption. In general, information about the stimulus (or target signal) in the population as a whole will be lost by pooling the responses. If the differences in the responses are due to non-random differences in $\dot{I}(\theta_1; \rho_\sigma)$, those differences can actually encode information about θ, and the identities of the spikes would need to be maintained to avoid losing significant amounts of information. The amount of information that can be retained by pooling responses will depend on the degree of similarity in information of the

constituent single-unit responses.

As in case B, the information in all constituent signals of ρ_{UD} was non-significantly different. When two spike trains are found on the same electrode with $\dot{I}(\theta_1;\rho_{1D}) \approx \dot{I}(\theta_1;\rho_{2D}) = \dot{I}_2$, the dual-unit signal was found to have at least as much information as the constituent single-unit signals; i.e., $\dot{I}(\theta_1;\rho_U) \geq \dot{I}_2$. $\dot{I}(\theta_1;\rho_{UD})$ increased by as much as 7.3 bits/s over $\dot{I}_2$, or equivalently, the change in mutual information was 41% of $\dot{I}_2$ and 36% of $\dot{I}_{bl}(\theta_1;\rho_1)$ (95% CI $\dot{I}_{UD} - \dot{I}_2 > 3.0$ bits/s). Thus, it appears that when the individual units encode information about θ_1 but their responses are confounded by extraneous information, the combination actually has a synergistic effect in which the occurrence of spikes is more heavily weighted at times of spikes actually resulting from θ_1. This should reduce the conditional entropy, thus increasing the mutual information. The attribute-specific information increased by another 15% when the third unit was added.

The results of case C fall into outcome category 4 (Sec. 4.3). In other words, once a response was corrupted by confounding information, 100% of the attribute-specific information $\dot{I}(\theta_1;\rho_1)$ could not be retrieved, but more of this information could be retrieved by reinforcing the response with another response transmitting $\dot{I}(\theta_1;\rho_1)$. Our model of mutual information in pooled pairs and triplets of neurons would state the following:

- When information about a stimulus attribute is encoded in common in each of the individual neurons, all the information is retained in the combined response.
- Information independent of the stimulus attribute of interest will corrupt the information encoding that stimulus attribute.
- Corruption of this sort is reduced in the multi-unit signal.
- More information can be retrieved in the labelled-line signal.

- The amount of additional information retrievable in the labelled-line signal increases with the disparity between the constituent single-unit information.

Outcome categories 1 and 2 from Sec. 4.3 would argue that using multi-unit information would have deleterious effects in neural decoding. However, all of the multi-unit responses studied here fall into outcome categories 3-5. When two units are similar in mutual information, the information content in the multi-unit signal of $\rho_1\prime$ and $\rho_2\prime$ combined maintains the information in the single-unit response alone.

One way to use both neurons while maintaining the individual identity of the neurons is to average the reconstructions of the stimulus from the individual responses. If the error in the stimulus reconstructions is due to random noise, the SNR of these estimates will increase by $\sqrt{n}$, where n is the number of neurons. This in turn would increase the mutual information rate by roughly 5 bits/s for every doubling of n. We tried this averaging approach for one trial of case D and indeed observed a 5 bit/s increase for two neurons at all levels of corruption. This gain exceeded the information increase in the multi-unit signal. Thus, perfectly maintaining the separate identity of the neurons in this case would allow for a significantly greater increase in information than combining the responses.

These results make the mere beginnings of understanding MI in multi-unit signals. It would be an unsupported jump to interpret these theoretical results to make practical BCI decisions. For example, the generation of spike trains lacked biological detail, and only 1 cluster of units were simulated. The lack of non-linear dynamics in the integrate-and-fire neuron used may place limitations on how far we can extend our interpretations to true biology. However, it does not seem unreasonable to use simplified models to capture the essential properties of the neurophysiology. Linear systems have accurately modeled spike outputs in cat LGN neurons [68], to use an example. We are also less concerned with modeling the actual behavior of a neuron

or network of neurons than we are with understanding how the loss of individual neuron identity can affect the information available to a decoding algorithm. We are looking only at relative differences between single unit and multi-unit activity and think the general conclusions we have drawn are still valid in spite of the simplicity of our model. Variety of information content was limited to confounding information with spike error.

We have considered applying direct methods of calculating information as described in [69]. An indirect method does not take full advantage of information theory to bypass any need to assume an encoding/decoding model. Although the method used in this chapter is typically used to estimate total mutual information in neural spike trains, it can also be viewed as a way of determining the linearly decoded mutual information in them. The stimulus reconstruction paradigm closely addresses the question of how much information a linear computational decoding algorithm would be able to extract from spike trains. We also recognize that our information calculations may be affected by our choice of the stimulus reconstruction algorithm. Again, we believe that the use of an optimal linear filter is valid for gaining insight into the real-world analogy of a BCI system. Optimal linear filtering and linear regression are often used in current decoding algorithms and have performed as well as other nonlinear neural computational decoders [70, 11, 12, 71, 72]. An indirect method using an optimal stimulus-reconstruction algorithm was a suitable approach for our purposes because we are concerned more with the information that can be extracted by the computational decoder about the stimulus than with the total capacity of the multi-unit or single-unit signals to encode information.

Aside from the implications for BCI applications, these simulations begin to provide grounds for understanding what information from single-unit spike trains is or is not lost in the multi-unit signals. We observed the effects of combining pairs and

triplets of single-unit spike trains on encoding information about a stimulus of interest. When the single-unit spike trains had disparate information about the stimulus, some of that information was lost by combining them. However, information in the single-unit spike trains was retained in the multi-unit response when the neurons detected on the same electrode were mutually dependent on the stimulus. Mutual information is corrupted by spikes independent of the stimulus but is increasingly reinforced in the multi-unit signal as more spike trains encoding attribute-specific information are added. We have not tried using a different decoder for the multi-unit signals. It should also be noted that $\dot{I}(\theta_1; \rho_1)$ is the information transmitted by an ideal single-unit spike train, with no spike sorting error. In practice, the discriminated spike trains will contain error. From the results obtained in section 3.3, it can be expected that a spike train with a reasonable 15% discrimination error [73] will have as much information as an accurate non-discriminated two-unit signal with a completely independent neuron corrupting it (case A).

The multi-unit response cannot carry more information than a perfectly sorted population of the constituent single-unit responses. We showed MI in multi-unit responses was significantly lower than in labelled-line responses when the diversity in responses most aided the discriminability of a stimulus; i.e., when there were confounding spikes in a spike train.

The next chapter aims to address the question of the necessity of spike sorting also by looking at differences between MI in clusters of pooled and labelled-line spike trains, but produces results that are more directly translatable to BCI applications. This was done modeling the cosine-tuning property, expanding the problem to include a 2nd electrode, and comparing pooled cluster with labelled-line cluster including spike sorting error.

Chapter 5

Cosine-tuned information

In the previous analyses, simulations were performed without incorporating a model of neural encoding that is relevant to a BCI. Here, we define a model of neural activity to further contextualize the problem in terms of a BCI. One feature commonly observed in the spike activity of motor cortical neurons during arm reaching tasks [74] is that of cosine-tuning. For a cosine-tuned neuron, the average firing rate of the neuron is modulated as the cosine of the angular difference between the movement direction and the neuron's "preferred direction". The preferred direction is the direction of movement for which the neuron fires maximally. Work in primates has typically utilized the directional cosine-tuning property in motor cortical neurons or other linear decoding methods which may perform reasonably well as long as the neurons are cosine-tuned to the decoded motor variables [70, 11, 12, 71]. Other groups have simulated cosine-tuned motor responses to demonstrate the performance of a movement decoder [75] and to model the effects of cell death on motor impairment [76]. In this chapter, the cosine tuning property was modelled in simulated neurons during a center-out reaching task, which will be described in Sec. 5.1, to study the effects of processing neural spike activity on information in a BCI.

These simulations examined the effects on information available to a BMI decoder of combining units with and without spike error in a cosine-tuned neural system. Fig. 5.1 shows the information processing system considered in this analysis. $I(\theta;\rho^b_{UA})$ and $I(\theta;\rho^b_{UB})$ were compared with $I(\theta;\rho^b_{LLA})$ and $I(\theta;\rho^b_{LLB})$, respectively; $I(\theta;\rho^b_{U})$ was compared with $I(\theta;\rho^b_{LL})$ with varying amounts of spike error. Mutual information calculations were performed with the following aims: 1) to model neurons with cosine-

tuned responses, 2) to include spike error, 3) to compute information in two or more neurons, and 4) to calculate mutual information by direct computation without the bias correction needed for limited data size (see Sec. 2.5.4). We studied the influence of major defining characteristics of neural activity and processing, including mean firing rate, modulation depth, bin size, and spike error, on information available to a decoder from cosine-tuned populations, again asking: 1) How much of the information in a labelled-line population does the pooled population retain? and 2) How much of the information is retained in the labelled-line population when spike error is introduced?

5.1 Implementing the cosine-tuning model

Four units were simulated as two pairs of unit activity, analogous to spike activity detected and sorted from two implanted extracellular microelectrodes, each with 2 neurons. Units were pooled together as a way to model the spike activity from a local extracellular space detected on one microelectrode. This was done by taking the union of the two binary spike trains $\rho_i(t)$ from the same electrode X to create ρ_{UX}. $\rho_{UA}(t)$ refers to the pooled multiunit spike train found on electrode A, while ρ_{UB} refers to the pooled multiunit spike train found on electrode B. Both pooled signals taken together (i.e., the pair of pooled multiunit signals) comprise the pooled population signal $\rho_U(t)$. The 2 units considered on each electrode A and B jointly but separately are denoted by $\rho_{LLA}(t)$ and $\rho_{LLB}(t)$, respectively. The two pairs collectively comprise the full joint labelled-line population signal $\rho_{LL}(t)$. The responses were binned to create spike count signals in ρ_1^b, ρ_2^b, ρ_3^b, ρ_4^b, ρ_{LLA}^b, ρ_{LLB}^b, ρ_{LL}^b, ρ_{UA}^b, ρ_{UB}^b, and ρ_U^b. Mutual information was computed between the target signal and the neural responses as individual units, pairs of labelled-line and pooled units, a pair of pooled multi-unit signals, and a 4-unit labelled-line system. The relationship between the individual unit responses,

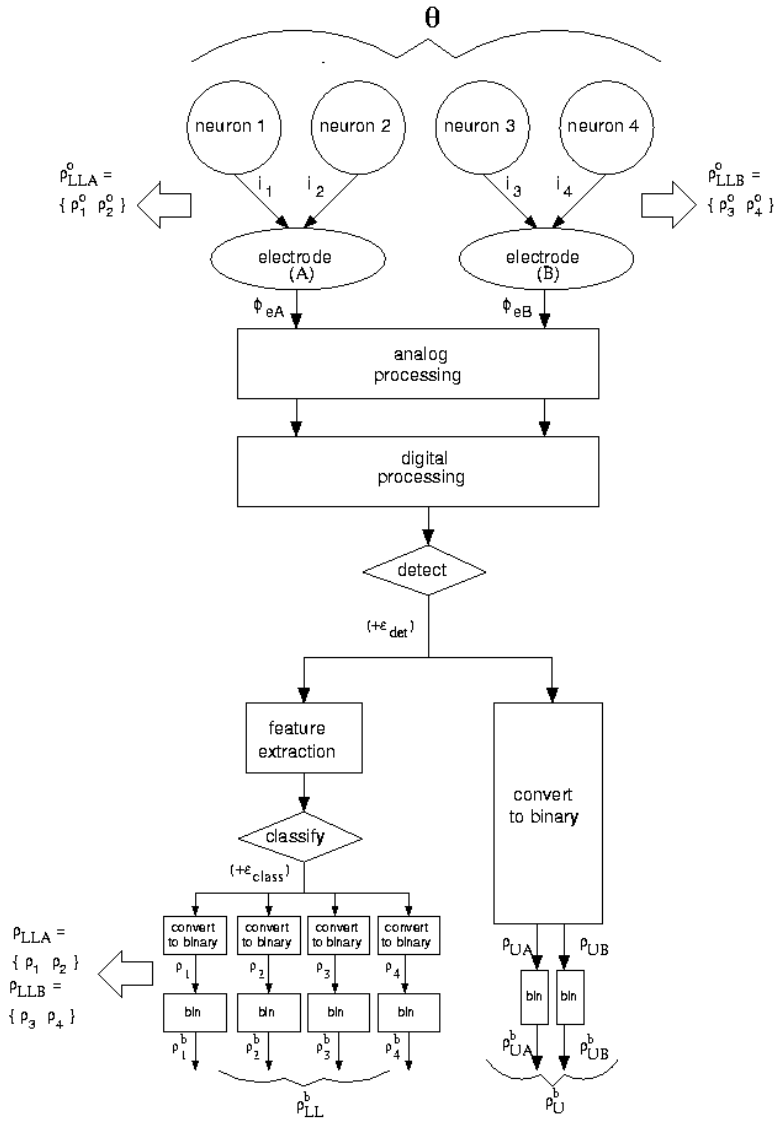


Figure 5.1: A 2-electrode system was simulated in which the extracellular signals from 2 cosine-tuned neurons contributed to the voltage detected on each electrode. This system is a subset of the information processing system shown in Fig. 2.2. In this analysis, the pooled signal on each electrode, $I(\theta; \rho^b_{UA})$ and $I(\theta; \rho^b_{UB})$, were compared with the corresponding labelled-line signals, $I(\theta; \rho^b_{LLA})$ and $I(\theta; \rho^b_{LLB})$, respectively; $I(\theta; \rho^b_U)$ was compared with $I(\theta; \rho^b_{LL})$ with varying amounts of spike error, ε.

the pooled responses, which represent non-sorted signals, and the joint labelled-line responses, which represent population activity sorted from signals detected on the electrodes, is schematically illustrated in Fig. 5.2. The naming convention is further described in Table 5.1. Since the analyses in this chapter examined signals after binning only, the supserscript ^{b} was omitted in figure labels, but all the reported information values can be assumed to be the MI between the target signal and binned neural signals.

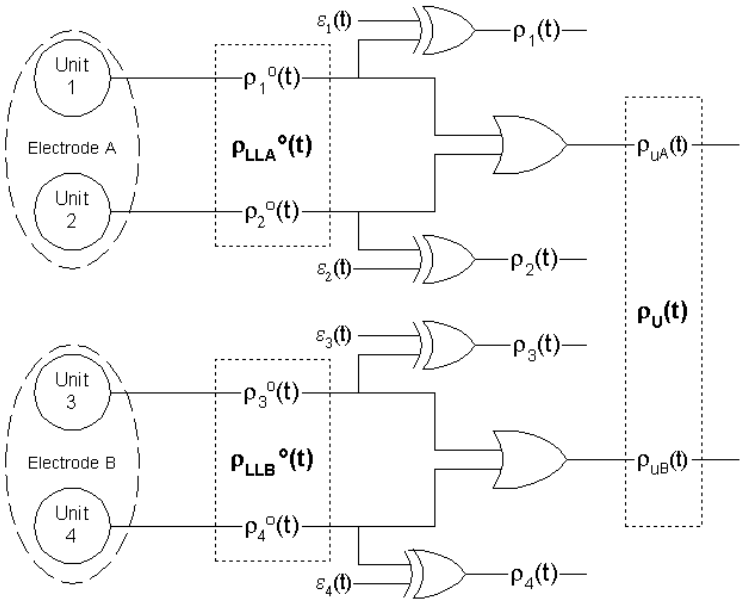


Figure 5.2: Schematic illustration of the analogy between the simulated signals and a hypothetical situation of recording spike trains from electrodes. Spike trains from unit 1 and unit 2 were detected by electrode A; from unit 3 and 4 by electrode B. Spike trains ρ_1^o and ρ_2^o jointly represented electrode A's labelled-line response (ρ_{LLA}). The union of $\rho_1^o(t)$ and $\rho_2^o(t)$ represented channel A's non-sorted pooled response (ρ_{UA}). Spike sorting error was also simulated; binary error signals $\epsilon(t)$ indicated times of spike error. If a spike existed in the original spike train, it was removed; if no spike existed, one was added. Spike trains ρ_1 and ρ_2 jointly represented electrode A's sorted labelled-line response (ρ_{LLA}), which without any error was considered equivalent to the original labelled-line response (ρ_{LLA}^o). Electrode B's spike trains similarly were either kept sorted (ρ_{LLB}) or were pooled (ρ_{UB}). Together, the pooled signals represented the population of non-sorted signals ρ_U.

5.1.1 Cosine-tuned responses

Neurons were assigned preferred directions (θ_{PD}), which were constructed as Cartesian-coordinate vectors (c_{PD}) for each neuron i:

$$c_{PD,i} = [\cos(\theta_{PD}) \quad \sin(\theta_{PD})] \tag{5.1}$$

A 2-dimensional velocity vector, $\mathbf{v}(t)$, for desired movement was generated for a center-out reaching task. Movements 2 seconds in duration were simulated in 8 directions $(0^o, 45^o, ..., 315^o)$ radially outward from the origin. The target signal θ was defined as the direction of movement. The results presented are from 8000 trials of the simulated center-out task.

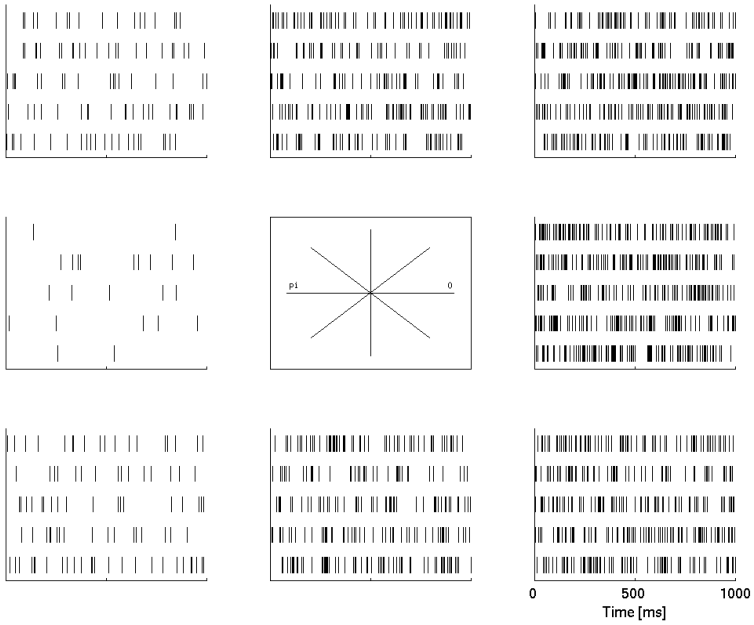


Figure 5.3: Raster plots showing spike times from a single neuron for 5 trials of the center-out reaching task in each of the 8 directions

Target instantaneous firing rates, $\lambda'(t)$, for this movement were calculated (Eqn. 5.2) and normalized to range between a minimum firing rate, to occur when the movement was in the $(\theta_{PD} + 180^o)$-direction, and a maximum firing rate, to occur

when the movement was in the θ_{PD}-direction (Eqn. 5.3).

$$\lambda' = c_{PD} \cdot \frac{\mathbf{v}}{\| v \|} \tag{5.2}$$

$$\lambda^* = \frac{\lambda' + 1}{2}(\lambda_{max} - \lambda_{min}) + \lambda_{min} \tag{5.3}$$

where magnitude of the velocity $\| \mathbf{v} \| = \sqrt{v_x^2 + v_y^2}$, and v_x and v_y are the two orthogonal components of $\mathbf{v}$.

Simulations were performed in both Matlab and C. The C code performed essentially equivalent simulations but was significantly faster and more powerful. One difference was that the movement was specified as purely movement directions, since speed was never incorporated to play a role in these simulations. Hence, the instantaneous firing rate for each unit was determined by

$$\lambda' = \frac{\lambda_{max}}{2}(cos(\theta_v - \theta_{PD}) + 1) + \lambda_{min} \tag{5.4}$$

Spike trains, ρ_i, were generated as a Poisson process with the instantaneous firing rates, $\lambda^*(t)$, similar to [75], at a sampling rate of 1kHz and were rectified into binary spike trains with no refractory period. The spike trains $\rho(t_i)$ were binned according to Eqn. 5.5, where n_Δ is the number of samples in a bin window of length Δ and the spike count in the j^{th} bin was stored in $\rho[j]$.

$$\rho[j] = \sum_{i=(j-1)n_\Delta}^{jn_\Delta - 1} \rho(t_i) \tag{5.5}$$

The simulation parameters were chosen based on a study in which cosine functions were fit to the tuning responses of motor cortical neurons in monkeys [77]. Using a cosine-tuning function of $\lambda(t) = b_0 + kcos(\theta(t) - \theta_{PD})$, with $b_0 = 10$ spikes/s and $k = 40$ spikes/s, the average firing rate ranged between approximately 2 spikes/s

Symbol	Description	Units
$\theta(t) \equiv \theta$	movement direction target signal	degrees
θ_{PD}	preferred direction	degrees
ϕ_{inter}	difference in preferred directions between electrodes	degrees
ϕ_{intra}	difference in preferred directions within electrodes	degrees
$\rho(t_i)$	binary spike train	spike event
$\rho[i] \equiv \rho$	binned spike count signal	number of spikes in bin i
$\rho'(t_i)$	binary spike train with spike error	spike event
ρ_i	response of unit i, $i \subset 1, 2, 3, 4$	number of spikes
ρ_{Ux}, $x \subset A, B$	response of union (pooled) electrode A or B	number of spikes
ρ_{LLx}, $x \subset A, B$	response of joint (labelled line) pair A or B	number of spikes
ρ_U	response of union population $\equiv \{\rho_{UA}\rho_{UB}\}$	number of spikes
ρ_{LL}	response of joint population $\equiv \{\rho_1\rho_2\rho_3\rho_4\}$	number of spikes
$\lambda(t) \equiv \lambda_t$	instantaneous firing rate	spikes/s
Δ	bin size	ms
$\lambda_{\Delta_t} \equiv \lambda$	average spike count	spikes/bin
λ_{peak}	peak average firing rate (λ at θ_{PD})	spikes/s
λ_{min}	minimum average firing rate (λ at $\theta_{PD} + 180^o$)	spikes/s
$\lambda_{avg} \equiv \mu$	grand mean firing rate across entire center-out reaching task	spikes/s
ϵ	error fraction	fraction of total number of spikes
$\epsilon_{\%}$	percent error	%
$\dot{\epsilon}$	error rate	spikes/s
T_θ	simulated time of movement in a given direction	seconds
L	total length of each simulated spike train in a complete center-out task	number of samples
L_{bin}	total length of each binned neural response in a complete center-out task	number of bins
$I(\theta; \rho_i) \equiv I_i$	mutual information between target signal θand	bits
$I(\theta; \rho_{LLX}) \equiv I_{LLX}$	MI between target and joint sorted pair on electrode X	
$I(\theta; \rho_{UB}) \equiv I_{UX}$	MI between target and union non-sorted pair on electrode X	
$I(\theta; \rho_{LL}) \equiv I_{LL}$	MI between target and full joint sorted population	
$I(\theta; \rho_U) \equiv I_U$	MI between target and non-sorted population	

Table 5.1: Summary table of parameters in this cosine-tuned neural system model. Notation for each parameter is given as well as a description.

during movements in the anti-preferred direction and 74 spikes/s during movements in the preferred direction. Other studies have used bin sizes ranging between 20ms and 100ms. Here we used $\Delta_t = 40ms$, $\lambda_{peak} = 74spikes/s$, and $\lambda_{min} = 2spikes/s$; if not otherwise stated, these parameter values may be assumed.

5.1.2 Information computation

Mutual information was computed according to Eqn. 5.6 and 5.7 [37, 78] (see also Sec. 2.2).

$$I(\boldsymbol{\theta};\boldsymbol{\rho}) = H(\boldsymbol{\rho}) - H(\boldsymbol{\rho}|\boldsymbol{\theta}) \tag{5.6}$$

where $H(\boldsymbol{\rho})$, the entropy of a signal $\boldsymbol{\rho}$, is defined as

$$H(\boldsymbol{\rho}) = -\sum_{\boldsymbol{\rho}} p(\rho)\log_2(p(\rho)) \tag{5.7}$$

and $H(\boldsymbol{\rho}|\boldsymbol{\theta})$, the conditional entropy of $\boldsymbol{\rho}$ conditioned on $\boldsymbol{\theta}$, is defined as

$$H(\boldsymbol{\rho}|\boldsymbol{\theta}) = -\sum_{\boldsymbol{\theta}} p(\theta)\sum_{\boldsymbol{\rho}} p(\rho|\theta)\log_2(p(\rho|\theta)) \tag{5.8}$$

Parameters and nomenclature are summarized in Table 5.1.

5.1.3 Numerical calculations

The definitions of mutual information are given by equations 2.6, 2.4, and 2.5. From these equations, it can be seen that the probabilities of spike counts for each direction of movement are required.

The conditional probability for a single unit can be determined from the Poisson probability distribution given for a generic signal $\mathbf{X}$ in Eqn. 5.9.

$$p(\mathbf{X} = x) = e^{-\lambda}\frac{\lambda^x}{x!} \tag{5.9}$$

In our model of cosine-tuned neurons, λ is a function of the movement direction θ. This is given by Eqn. 5.2 and 5.3. The conditional probability was obtained by simply substituting $\lambda(\theta)$ for the λ parameter. The conditional probability was generated in Matlab as a column array.

$$p(\boldsymbol{\rho} = r|\theta) = e^{-\lambda(\theta)}\frac{\lambda(\theta)^r}{r!} \tag{5.10}$$

Then, determining the total probability of $\boldsymbol{\rho}$ required only the dot product between the probability of target values $p(\theta)$ and the conditional response probability $p(\boldsymbol{\rho}|\theta)$. In order to determine the conditional probability of a joint pair of units, the $p(\boldsymbol{\rho}_i|\boldsymbol{\theta})$ was expanded and then array multiplied, in which each element in one matrix is multiplied with the corresponding element in the second matrix. The conditional probability of all four units was then determined by expanding the joint pair probabilities and array multiplying them (pairwise multiplication). Equivalently, the single-unit probabilities were expanded further and each of the four expanded probabilities array-multiplied. The pooled signals were approximated as $\lambda_{UA}(\theta) \approx \lambda_1(\theta) + \lambda_2(\theta)$ and $\lambda_{UB}(\theta) \approx \lambda_3(\theta) + \lambda_4(\theta)$, since we know the analytical distribution of the sum of Poisson signals but not the union. The probabilities for the pair of pooled signals were determined from ρ_{UA} and ρ_{UB}'s probabilities by expansion and multiplication.

When computing the entropy values, the conditional probabilities in equations 5.7 and 5.8, into which Eqn. 5.10 was substituted, were summed over a finite range of response values. To determine which response values to sum over, the maximum response value needed to reach 99.5% of the cumulative Poisson probability distribution was determined. This percentage was expressed as a proportion was termed $p_{capture}$.

5.1.4 Direct computation

It was necessary to estimate the mutual information directly from simulated data for two reasons. 1) The non-sorted signals were created in the former (numerical computation) case by treating them as the sum of single-unit signals, rather than a true union. 2) Spike error needed to be simulated, which changed the distribution for a given direction of movement from being Poisson to some unknown distribution. These direct computations were made as described in section 2.5.1.

As described in Sec. 2.5.4, the accuracy of the direct computation of MI is dependent on the amount of data that can be produced; more data allows a more accurate calculation of the probability distributions in Eqn. 5.7 and 5.8. In Appendix **??**, we describe the generations of code developed to perform the cosine-tuning simulations at the desired lengths.

5.1.5 Information in a population vector decoder output

In Sec. 5.6, we compare the output of a population vector decoder using sorted signals with that using non-sorted signals. The population vector is a commonly used decoder in the BCI community [70, 79, 80, 11]. It assumes that neurons are cosine-tuned. The direction of movement is then decoded from the weighted sum of the preferred directions in the ensemble of neurons. This weighted sum is called the population vector for a movement in direction θ and is defined for N neurons by Eqn. 5.11 [79],

$$\mathbf{P}(\theta) = \frac{1}{N}(\boldsymbol{\lambda}(\theta) - \boldsymbol{\lambda_{min}}) \cdot \mathbf{c_{PD}} \tag{5.11}$$

which can also be written for each component of the population vector as:

$$P_j(\theta) = \sum_i (\lambda_i(\theta) - \lambda_{min,i}) \cdot c_{PD,i,j} \tag{5.12}$$

where $c_{PD,i,j}$ is the j^{th} component of the preferred direction of unit i given by Eqn. 5.1, and $\lambda_i(\theta) - \lambda_{min,i}$ gives the discharge rate above the baseline firing rate for unit i. In Sec. 5.6, we compare the information extracted by the population vector decoder using the measured firing rate of sorted neurons and using that of non-sorted neurons, giving the following:

$$P_j(\theta) = \sum_i (\rho_i(\theta) - \rho_{min,i}) \cdot c_{PD,i,j} \tag{5.13}$$

The known preferred direction of each sorted neuron i was used for $c_{PD,i}$. For the non-sorted units, the preferred direction was the average of the preferred directions of the pooled neurons. Thus, for cluster i which pools neurons j and k, for instance, $c_{PD,i} = \frac{1}{2}(c_{PD,j} + c_{PD,k})$. The estimate of movement direction from the population vector is the direction of the population vector; i.e.,

$$\hat{\theta} = \arctan \frac{P_y}{P_x} \tag{5.14}$$

for $\mathbf{P} = \begin{bmatrix} P_x & P_y \end{bmatrix}$ [79].

A comparison was made between information decoded from non-sorted and from sorted signals. Mutual information was computed between $\boldsymbol{\theta}$ and $\hat{\boldsymbol{\theta}}_{LL}$, using the method described in Sec. 2.5.1 and compared to mutual information between $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\theta}}_U$. The population vector output was quantized, with $n_q = 32$ in Eqn. 2.12, before $H(\boldsymbol{\theta})$ and $H(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}})$ were computed. Mutual information in the decoded output using the population vector is denoted by a PV subscript; e.g., $I_{PV}(\theta; \hat{\theta}_{LL})$ is the MI between $\boldsymbol{\theta}$ and the estimate of the target signal decoded using a population vector of the labelled-line signals. MI was also computed between $\boldsymbol{\theta}$ and $\mathbf{P_{LL}}$, the population vector computed from ρ_{LL} (refer to Fig. 5.4 and Eqn. 5.13), and compared to that between $\boldsymbol{\theta}$ and $\mathbf{P_U}$, the population vector computed from ρ_U.

5.1.6 Spike error

Extraneous spikes were randomly added (false positives) and randomly selected spikes were omitted (false negatives) from $\boldsymbol{\rho}$ to simulate spike sorting error. The amount of error added was specified as a proportion of the total number of spikes in the original spike train $\boldsymbol{\rho}$. Error was defined as the ratio of the total number of spike errors in $\boldsymbol{\rho}'(t_i)$ to the total number of spikes in the original spike train without error $\boldsymbol{\rho}(t_i)$ (Eqn. 5.15).

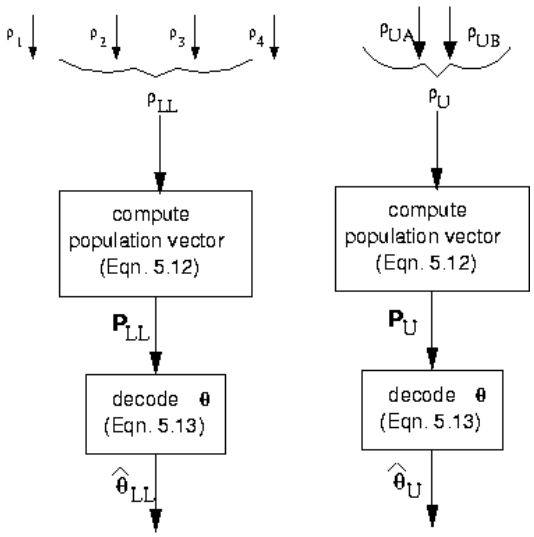


Figure 5.4: A population vector method was used to decode the movement direction from the binned labelled line population and the binned union population. In both cases, the population vector was computed from the binned spike activity according to Eqn. 5.13. The movement direction is then estimated to be the direction of the population vector using Eqn. 5.14.

$$\varepsilon = \frac{\sum_i \rho'(t_i) - \rho(t_i)}{\sum_i \rho(t_i)} \tag{5.15}$$

Percent error is denoted by $\varepsilon_{\%}$ and defined as $\varepsilon_{\%} = 100\varepsilon$. These spike additions and omissions simulated spike sorting error. An exponential fit of mutual information to percent spike error was obtained by linearly regressing the natural *log* of MI on spike error. The fit was in the form $ln(I) = B\varepsilon + M$ with a coefficient of determination given by R^2.

Different values of I_{bl} were attempted while regressing $ln(I - I_{bl})$ onto ε. The I_{bl} which yielded the maximum R^2 was selected for the fit. It was also found that when the data across the whole range of ε values were used, the curve fit the data points well overall but at 0% error did not reach as high as the data point and did not have as steep a slope initially as the data points. As further detailed in Sec. 5.3, the curve was fit to the relevant ranges of error more closely.

The final form of the fit was given by

$$\hat{I} = \begin{cases} e^{B_0\varepsilon + M_0} + I_{bl} & \text{for } 0 \leq \varepsilon_{\%} \leq 50\% \\ e^{B_1\varepsilon + M_1} + I_{bl} & \text{for } 50 < \varepsilon_{\%} \leq 200\% \end{cases} \tag{5.16}$$

We compared sorted and non-sorted signals in the presence of spike error. We also considered detection error in non-sorted signals. However, we did not make a distinction between misdetection and misclassification error when simulating this error, even though results from Sec. 3.3.2 indicate that misdetection error causes slightly more severe degradation of MI than misclassifcation error (information rate decays to half its maximum value at 9.5% misdetection error and 13.3% misclassification error). Therefore, the values we obtained for MI in sorted signals with error should be treated as a lower bound, and MI loss due to pooling could be more severe than what is shown. We simulated non-sorted signals without any error, and then with a proportion of the error that was in the sorted signals. Detection error and the problem of determining a fixed amount to add in the case of non-sorted signals was described in chapter 3 (sections 3.1 and 3.3.2).

There have been a few studies that have estimated detection error at different SNR values. Because of the infinite number of combinations of detection error with classification error and other parameters, we chose to look at a couple of particular cases within a reasonable range of detection error. From [81] and [82], the SNR of *in vivo* recordings averages between 3 and 4. We hesitate to use the detection error values reported therein because they were referenced against the output of another signal acquisition and detection system that could have its own false alarms and missed spikes, and the values were highly variable even for similar SNRs.

Choi *et al.* [61] did show correct detection and false alarm rates for low SNR environments. We do not know precisely how the false alarm ratio in [61] compares

with [81] and [82] because they do not explicitly define the false alarm ratio. However, if the conventional definition is interpreted for the action potential case to be r_{FA}/f_s, where r_{FA} is the number of false alarms per second as defined in [81] and f_s is the sampling rate, then the false alarm rates reported in [61] appear to be much higher than that reported in [81] and [82].

We do note that Choi *et al.* stated that "above [an SNR of] 4.0, most existing detectors show invariably good performance", and the aforementioned typical *in vivo* SNRs of 3-4 are roughly the square root of the SNR used in Choi *et al.* because of the difference in defining SNR. We will call SNR from Obeid *et al.* SNR_{sqrt}, so that the typical *in vivo* SNR reported in [81, 82] actually ranges between 9-16. The correct detection percentage, which is a more straightforward quantity to define, reported by both groups are in a similar range. Choi's values, based on known spike times, reached 65% and 85% at SNR=4.0. Thus, for typical SNR values, we expect the correct detection to be much higher, and consider 10% detection error to be a maximum for average recording environments. For the comparisons in Sec. 5.5, we considered pooled signals with varying amounts of detection error, as defined in Eqn. 5.15, that ranged from 0 to 10%, as well as detection error defined as a percentage of the total spike error (from $\varepsilon_{detection}/\varepsilon = 0$ to $\varepsilon_{detection}/\varepsilon = 0.1$).

5.1.7 Verification of Poisson-distributed spike counts and cosine-tuning

Results demonstrated that the cosine-tuning behavior observed experimentally in M1 neurons was captured in simulation. The spike activity of neuron 1 ($\theta_{PD,1} = 0$) during outward movement in each of the eight center-out reaching task directions is shown in Fig. 5.3 at each of the respective positions corresponding to the movement direction. Raster plots are shown for 5 trials in each direction.

The center-out reaching task was simulated by making θ vary at regular intervals and span $0 \rightarrow 360^o$ uniformly, as shown in Fig. 5.5. In this case, unit 1 (ρ_1) had a preferred direction of $\theta_{PD,1} = 180^o$, and unit 3 had a preferred direction of $\theta_{PD,3} = 0$.

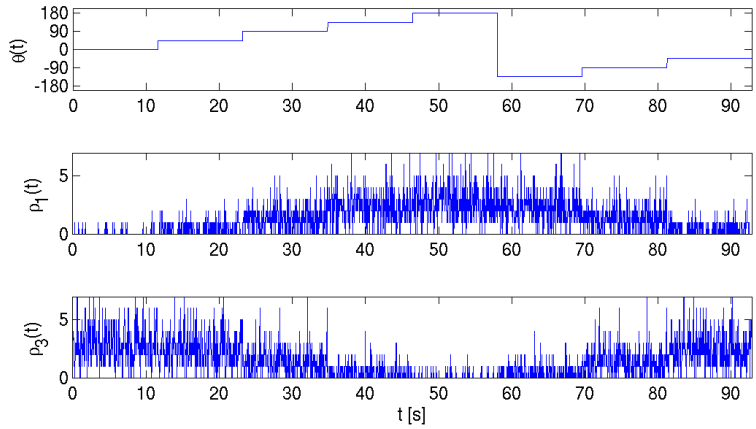


Figure 5.5: Movement direction signal (top) and the cosine-tuned response signals (bottom) across time in seconds. Unit 1's response ρ_1 shows a preferred direction of 180^o and unit 3 a preferred direction of 0.

On a given trial, there was significant variation in the instantaneous firing rate of a neuron, but the average firing rate was well-fit by a cosine function $\lambda \sim \cos(\theta - \theta_{PD})$ (Fig. 5.6). The instantaneous firing rates averaged across 1000 trials are shown for unit 1 (Fig. 5.6a) and unit 2 (Fig. 5.6b), whose preferred directions are 180^o and 90^o, respectively.

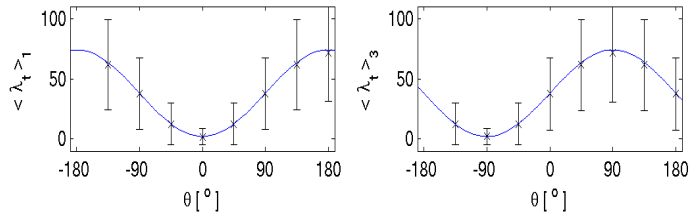


Figure 5.6: Cosine-tuning is displayed in the firing rates of the simulated neurons during outward movement in each of the directions during the center-out reaching task. Instantaneous firing rates in spikes/s ('x') of unit 1 and unit 3 as a function of movement direction θ averaged across 1000 trials of the center-out reaching task. One standard deviation away from mean indicated by bars. **a)** [left] Unit 1 $\theta_{PD,1} = 180^o$. **b)** [right] Unit 3 $\theta_{PD,3} = 90^o$

.

The generated signals were verified to be Poisson both by looking at the distribution of spike counts for each movement direction and also by comparing the mean and variance of the spike counts during each movement in the same direction. The mean and variance is shown in Table 5.2. As expected for a Poisson distribution, the mean and variance were roughly equal.

Table 5.2: The mean 100ms-bin count and variance during each direction of movement.

	0	45^o	90^o	135^o	180^o	-135^o	-90^o	-45^o
mean	0	1.49	5.06	8.56	10.01	8.56	5.05	1.49
variance	0	1.45	4.91	8.62	10.0	8.601	4.91	1.45

Fig. 5.7 shows the Poisson distribution of a unit whose preferred direction was 90^o as well as the shifting of average firing rate across direction of movement. The mean firing rate peaked at the preferred direction of 90^o and fell to its minimum at 270^o (180^o from the preferred direction). The minimum firing rate could be considered a suppression of firing at the anti-preferred direction even though a background firing rate with no movement was not included in the simulation.

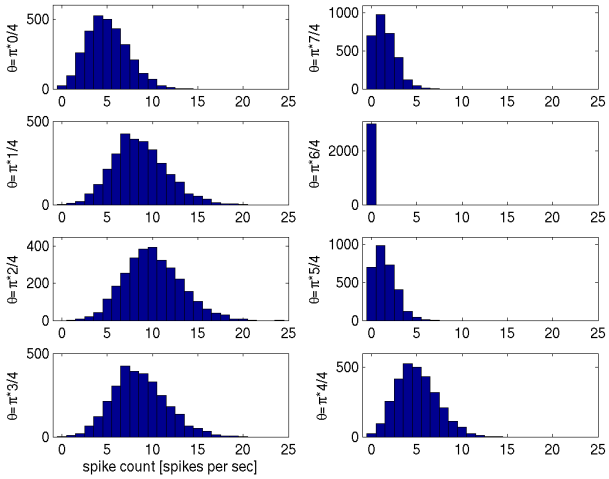


Figure 5.7: Histogram of spike counts of a unit with a preferred direction of 90^o during movements in each of the 8 radial directions for $\lambda_{peak} = 100$ spikes/s and $\lambda_{min} = 0$

5.1.8 Negligible difference in MI between summed spike counts and union signals

If a simple threshold detector was used to detect spikes without sorting of signals, then hypothetically, the acquired spike train would be a superposition of the spikes from the nearby source neurons. The binned spike counts from the pooled spike train would not be exactly equal to the sum of binned spike counts from each component unit. We simulated both the pooled union signals and the summed bin counts to determine how much of a difference in MI using summed signals, as opposed to union signals, would cause. A comparison of the information, when the intra-electrode angular tuning difference was $\phi_{intra} = -90^o$, revealed the percent differences to be $< 2\%$:

$$\Delta_{\Sigma-\cup}(\rho_U) = -1.7\%, \quad \Delta_{\Sigma-\cup}(\rho_{UA}) = -1.8\%, \quad \Delta_{\Sigma-\cup}(\rho_{UB}) = -1.8\%$$

where $\Delta_{\Sigma-\cup}(\rho_X) = 100\frac{I(\theta;\rho_X^{\Sigma}) - I(\theta;\rho_X^{\cup})}{I(\theta;\rho_X^{\cup})}$
and ρ_X^{Σ} = the sum of the binned signals
while $\rho_X^{\cup}$ = the union of the binned signals.

Thus, we confidently use the numerical computations, which treat the pooled signals as summed signals, when making MI computations for signals with no spike error.

5.2 Dependence on phase

5.2.1 Information loss due to pooling on one electrode

MI loss in pooled neurons with dissimilar tuning

We first examine the information from a pair of neurons with dissimilar directional tuning on Electrode A. Neuron 1's preferred direction was fixed at 180^o, while $\theta_{PD,2}$ was varied from $\theta_{PD,1}$ by an angular difference; i.e., $\theta_{PD,2} = \theta_{PD,1} + \phi_{intra}$. The

information as a function of the angular difference between units is shown in Fig. 5.8 a-c). The mutual information between movement direction and the spike count is shown for the responses of the individual units (a), the joint 2-neuron population (b), and the 2-unit union (c).

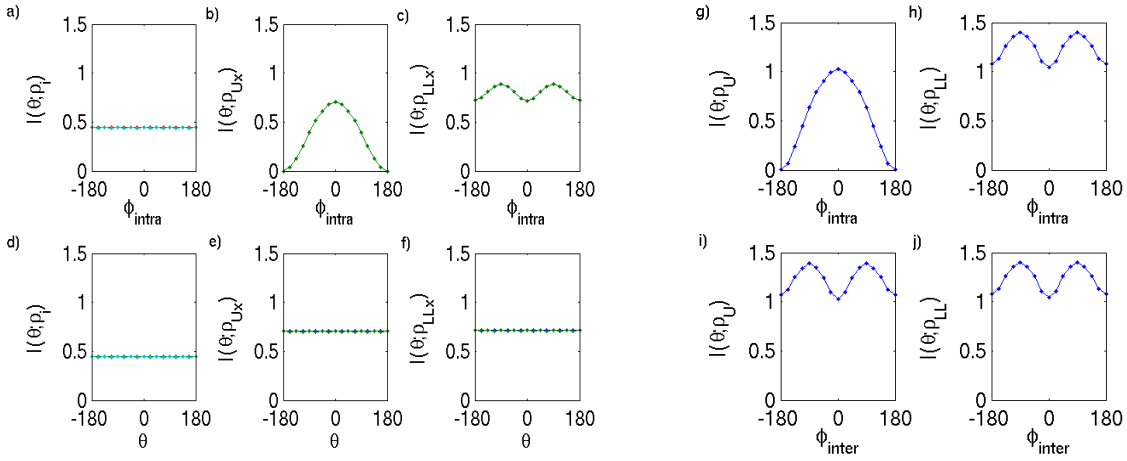


Figure 5.8: **a-f)** [Left panel] Mutual information between movement direction and neural responses in a 2-neuron system for unit 1 and unit 2 (a, d); the joint pairs ρ_{LLA} and ρ_{LLB} (b, e); and the combined unions ρ_{UA} and ρ_{UB} (c, f). **a-c)** $\theta_{PD,1} = 180^o$, $\theta_{PD,2} = 180^o + \phi_{intra}$. **d-f)** $\theta_{PD,1} = \theta_{PD,2} = 180^o$ **g-j)** [Right panel] 4-neuron system (2 units on each of 2 electrodes): **g, i)** union population **h, j)** joint population. **g, h)** Preferred directions of units from common electrode differ. **i, j)** Preferred directions of units from a common electrode are equal but differ from units on the other electrode.

The information contained in the response of each neuron individually was independent of the preferred direction and showed little variation around 0.45 bits, or equivalently 11.2 bits/s (Fig. 5.8a). When the two units were combined as a union and were completely out of phase, the union contained no information, but the information increased as the preferred directions became more similar to one another, peaking when there was no angular difference. We know that information in a pooled signal (e.g., ρ_{UA}) is necessarily less than or equal to the information in the corresponding labelled-line signal (ρ_{LLA}) which perfectly replicates the original spike trains ρ^o_{LLA} due to the information processing inequality [33]. These results demonstrate that information loss does occur and can be severe when unit activity is

not labelled, but this loss is directly dependent on the magnitude of the difference in preferred directions. Fig. 5.9 a) and b) depict how that information loss grows with dissimilarity in tuning.

Conservation of information with redundantly tuned units

Now, we take a closer look at the case of similar tuning between the units from a common electrode. The two neurons were assigned equal preferred directions ($\theta_{PD,1} = \theta_{PD,2}$). The information was computed as the preferred direction varied. Whether the two units were combined as 1 multi-unit union signal or kept as 2 separate single-unit signals, the information in the 2-unit population was approximately 0.72 bits (17.8 bits/s). The average information loss due to pooling the units within a pair was less than 1%, as illustrated in Fig. 5.9 c) and d); $\langle \Delta_{Ux-LLx} \rangle = -0.74 \pm 0.013\%$, mean and standard deviation across all ϕ values and trials. These results show that information in the labelled-line pair can be conserved when pooled, as long as the units are redundantly tuned.

As expected, information was symmetric with respect to angular difference; i.e., $I(\theta; \rho(\phi)) = I(\theta; \rho(-\phi))$. This was verified in simulation for the intra-electrode angular difference case for ϕ varied from -180^o to $+180^o$. Simulations hereafter were performed only for non-positive ϕ, and $I(\theta; \rho_x)|_{\phi=\phi'>0}$ was determined from $I(\theta; \rho_x)|_{\phi=-\phi'}$.

5.2.2 Information loss due to pooling on 2 electrodes

Information loss in the case of dissimilar tuning on each electrode

We expanded the scenario in Sec. 5.2.1 from 1 electrode to 2 electrodes. The labelled-line information and pooled information in two pairs of neurons varied with ϕ_{intra} similarly to the information in one pair of neurons, but the overall information was higher. The information increased as each pair of neurons became less redundant;

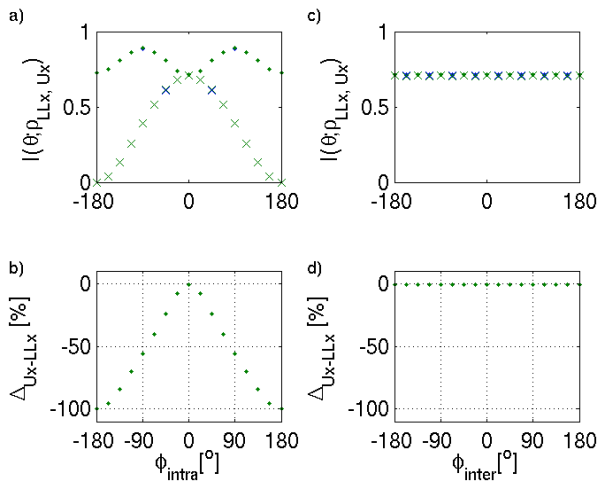


Figure 5.9: A comparison of information in the joint pair and union pair as ϕ, the angle between preferred directions, varies. **a-b)** The preferred directions of the units being pooled had an angular difference of ϕ_{intra}. **a)** 2-unit information in ρ_{LLA} and ρ_{LLB} ('·') and ρ_{UA} and ρ_{UB} ('x') as the angular difference in θ_{PD} within electrodes varied. **b)** Percent difference between information in ρ_{Ux} and ρ_{LLx}. **c-d)** Same quantities for the case in which the units being pooled had equal preferred directions. ϕ_{inter} represents the difference in preferred direction between electrodes.

i.e., as ϕ_{intra} grew from 0 to $\pm 90^o$ (Fig. 5.8g and h). The information loss due to pooling units within a pair with different preferred directions dropped from close to 0 when the units within a pair were completely in phase to 100% information loss when completely out of phase. This pattern occurred for both 2-unit information and 4-unit information (Fig. 5.10a and b).

Minimum information loss due to pooling in the case of similar tuning within electrodes

We demonstrated that information loss due to pooling a pair of units on a single electrode was minimized when the units had equal preferred directions (Sec. 5.2.1). We simulated a comparable situation for two electrodes in which the two units were tuned similarly within each electrode but the tuning of units found on Electrode A differed from that of units found on Electrode B. Fig. 5.8i and j) show results for the case in which electrode A's preferred directions were fixed at 180^o and θ_{PD3} and

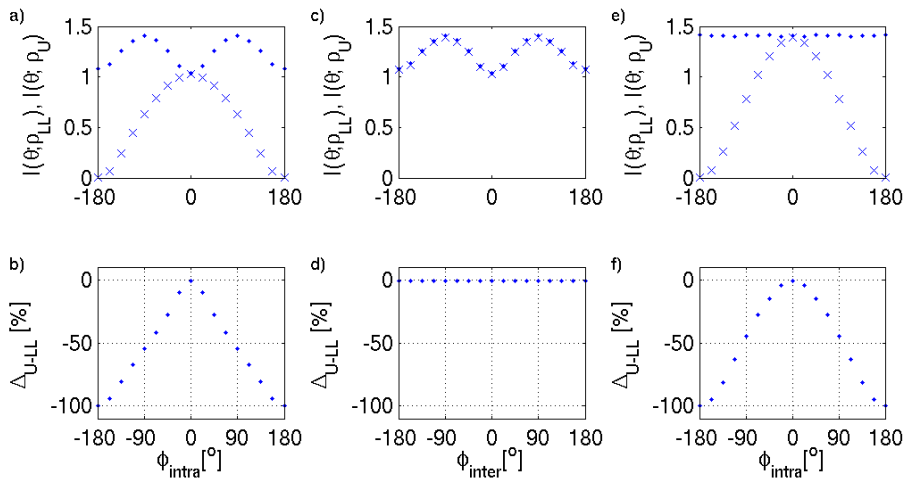


Figure 5.10: A comparison of information in the joint and union population as ϕ, the angle between preferred directions, varies. **a-b)** The preferred directions of the units being pooled differed by ϕ_{intra}. **a)** 4-unit information in ρ_{LL} ('·') and ρ_U ('x') as the angular difference within electrodes varied. **b)** Percent difference between information in ρ_{Ux} and ρ_{LLx}. **c-d)** The pooled units had equal preferred directions while angular difference in preferred directions between units on electrode A and units on electrode B are given by ϕ_{inter}. **e-f)** ϕ_{inter} was fixed at 90^o. Preferred directions of units from the same electrode also differed; that difference is given by ϕ_{intra}.

θ_{PD4} were incrementally varied from 0^o to 360^o. The information loss was less than 1% when both sets of units were pooled; $\langle \Delta_{U-LL} \rangle = -0.96 \pm 0.12\%$ (Fig. 5.10c and d). The information was indistinguishable between pairs, e.g. $I(\theta;\rho_{UA}) \approx I(\theta;\rho_{UB})$. Again, no information loss occurred for 2 electrodes when tuning within each electrode was equal but tuning on Electrode A differed from tuning on Electrode B.

Recoverable information in the case of maximal information on 2 electrodes

We showed that the information loss due to pooling was minimized when neurons on the same electrode were redundant, both when the information between electrodes was redundant or non-redundant. In those cases, the information in the sorted (labelled-line) population was potentially sub-maximal because more information might be gained by adding units on Electrode B with different tuning rather

than responses that reinforce those on Electrode A. We now look at the effect of pooling when the units from the same electrode were tuned differently *and* the units on Electrode B were tuned differently from Electrode A such that the information in the sorted population was maximized; this happened for an inter-electrode preferred-direction difference of $\phi_{inter} = 90^o$.

The 2-unit information ($I(\theta; \rho_{Ux})$, $I(\theta; \rho_{LLx})$) was equivalent to that in the case of intra-electrode differences only (Sec. 5.2.2). Fig. 5.10 e shows the comparison of $I(\theta; \rho_U)$ and $I(\theta; \rho_{LL})$. The information in the full sorted population signal is now constantly at its maximum level for all ϕ_{intra}. This information was fully recovered in the pooled signal when the preferred directions of the pooled units were equal. The information in the pooled signal dropped to 0 in a similar fashion as in the other cases examined when the units' responses were completely out of phase. The information loss was less sensitive to preferred direction differences when the differences were small compared to the case in which only intra-cluster differences existed. For example, there was a 28% information loss once ϕ_{intra} had reached 22.5^o when the information between electrodes was redundant (Fig. 5.10b), whereas the information loss was 15% for the same ϕ_{intra} when the information between electrodes was non-redundant (Fig. 5.10f).

Population MI was determined for simultaneously varying ϕ_{inter} and ϕ_{intra} (Fig. 5.11). The surface plot on the right side of Fig. 5.12 shows percent differences in pooled and labelled-line MI for different combinations of ϕ_{intra} and ϕ_{inter}. At all ϕ_{inter} values, the dependence of Δ_{U-LL} on ϕ_{intra} remained the same, decreasing linearly from 0 to -100% as ϕ_{intra} increased from 0 to 180^o. The contour plot on the left side shows the differences in preferred directions that can be tolerated for a given level of information loss. To maintain 10% loss, for instance, neurons with preferred directions within $23 - 24^o$ of each other could be pooled if one pair was completely

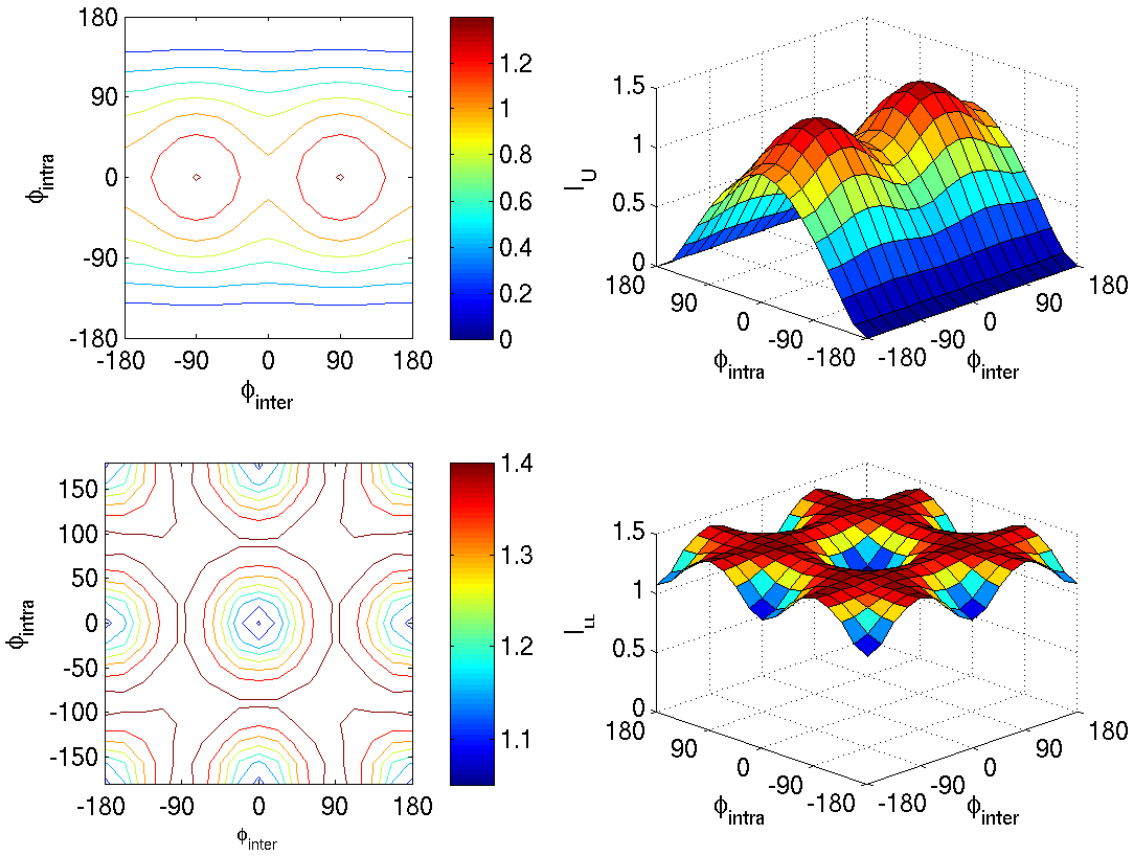


Figure 5.11: A 3-d view of MI in labelled-line and pooled population signals as a function of both ϕ_{inter} and ϕ_{intra}.

out of phase or completely redundant with the pair on the second electrode ($\phi_{inter} = \pm 180^o$ or 0^o). However, as the inter-electrode difference moves toward 90^o, a larger intra-electrode difference can be tolerated for 10% information loss; right at $\phi_{inter} = 90^o$, a ϕ_{intra} of 37^o can be tolerated. In other words, while pooling incurs information loss when neurons of different preffered directions are pooled, the diversity between electrodes compensates for that loss to some extent.

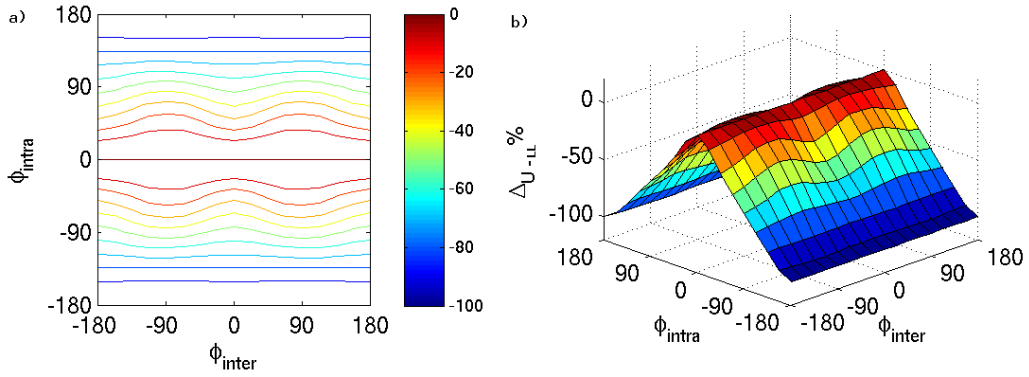


Figure 5.12: MI differences as a joint function of both intra- and inter-electrode angular tuning differences. **a)** Iso-contour lines showing what combinations of ϕ_{intra} and ϕ_{inter} give equivalent information losses. **b)** 3-dimensional surface plot of Δ_{U-LL}.

5.3 Information tradeoff: angular tuning differences versus spike error

Spike trains were generated, spike error added (Fig. 5.2 ρ_i'), and the union signals created before any signals were binned. Again, it is important to remember that random spike error was added, whereas true sorting error may originate from correlated spike trains. Thus, these results should be interpreted as the lower bound on MI of sorted signals with sorting error, and the percent differences between non-sorted and sorted signals are likely to be more negative than the values reported here. Simulations, with spike error added, were performed at $\phi_{intra} = 0^o$, at which information in the labelled line population is at a minimum, and $\phi_{intra} = -90^o$, at which labelled line information is at its peak. Information decayed exponentially (Fig. 5.13), with the linear regression of $ln(I)$ on ε yielding $R^2 = 1.00$, .99, 1.00, .98, and 1.00 for $I_i(\varepsilon)$, $I_{LLx}(\varepsilon)$, $I_{Ux}(\varepsilon)$, $I_{LL}(\varepsilon)$, and $I_U(\varepsilon)$, respectively, all with $p < .01$. In the 0–30% error range, the information in the labelled line pair and pooled pair decayed at similar rates, as indicated by the slopes of the regression lines: $B_{LLx} = -1.23$ vs $B_{Ux} = -1.26$; the information in the 4-unit labelled line signals decayed at a slightly higher rate than that in the corresponding union signals: $B_{LL} = -0.95$ vs $B_U = -0.89$.

Percent difference in information between the pooled and labelled line signals as spike error was added was calculated according to Eqn. 5.17 and is shown in Fig. 5.14.

$$\Delta_{U_0-LL_\varepsilon} = 100\frac{I_U(\varepsilon_\% = 0) - I_{LL}(\varepsilon_\% = \varepsilon)}{I_{LL}(\varepsilon_\% = \varepsilon)} \tag{5.17}$$

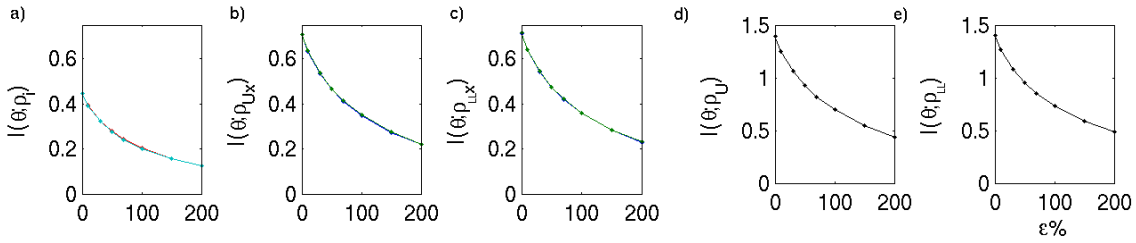


Figure 5.13: Information in **a)** single units, **b)** union pairs, **c)** labelled-line pairs, **d)** union population, and **e)** labelled-line population decreased in an exponential fashion as spike error was added to each neuron's response. Spike trains had equal preferred directions within electrodes and a difference of 90^o between electrodes.

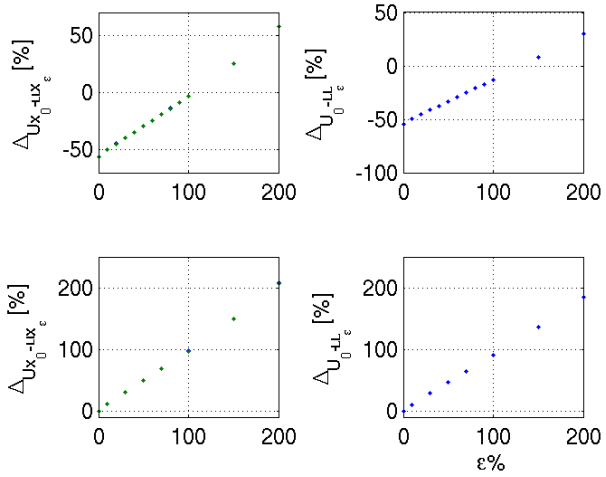


Figure 5.14: Percent difference between information in union and joint signals for the 2-unit system [left] and the 4-unit system [right] as spike error is added to single-unit spike trains. [Top] Preferred directions within pairs differ by 90^o. [Bottom] Preferred directions within pairs are equal.

Information decay was not truly exponential. By 1000% error I_i, I_{LLX}, I_{UX}, and I_U all decreased to 0. Also, the linear regression fits had high R^2 (.98-.99) when using points from different parts of the full range $0-500\%$; e.g., $0-100\%$, $0-150\%$,

$0-200\%$, and $0-500\%$, but the slopes decreased as more of the range was used. Choosing a different range resulted in different B and M values and determined whether the initial slope was best fit, or the asymptotic decline was better fit. Fig. 5.15 shows a comparison of fits to different parts of the range; choosing which points to fit depends on which part of the range we are more interested in. Since typical errors are between 0–30%, we are most interested in how the information behaves in the 0–30% error range. Initially, I_{LL} decays more quickly than I_U ($B_{LL} = -0.95$ vs. $B_U = -0.89$ for fit to 0–30% error) but then the union population information falls off more quickly after the initial decay ($B_{LL} = -0.54$, $B_U = -0.69$ for fit to $\varepsilon_{\%}$ =40–200%, Fig. 5.15).

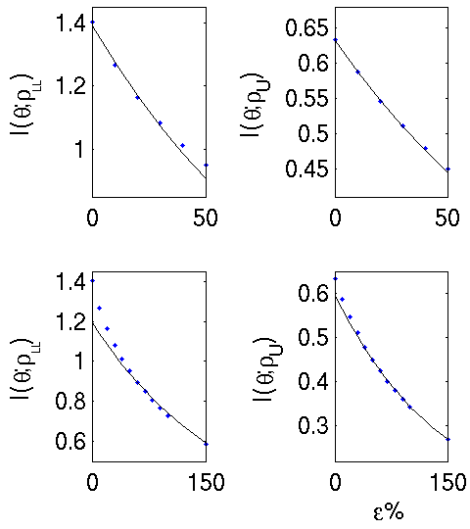


Figure 5.15: Fitting 2 different curves to different ranges of percent error for both I_{LL} [left] and I_U [right]. The upper plots show a fit to the range 0-30% error while the lower plots show a fit to the range 40-150% error.

The rate of decay was compared for different angular tuning differences. Exponential curves were fit to MI for $\phi_{inter} \subset \{0^o, 45^o, 90^o, 135^o, 180^o\}$ at a fixed ϕ_{intra} of 90^o. It was verified that coefficients of the regression were identical for the fits to I_i, I_{UX}, and I_{LLX}. We use these fits for the contour plots generated for section 5.5. A further investigation including detection error, using the fits shown here, is described in section 5.5.

5.4 Interdependent signal features of cosine-tuned Poisson

It is interesting to note that the mean firing rates, even with 100% error, were still cosine-tuned to the proper angle. Because of the way spike error was added to the spike trains, an equal proportion of spikes was added and removed from the spikes in each direction. Thus, the information contained in the mean firing rate should not be affected, indicating that the information loss was due, instead, to factors other than the mean. We explore now the influence that signal features such as the level of firing and the scaling of the tuning curve might have on the difference between MI in sorted and non-sorted signals.

In Sec. 5.2 and 5.3, we constrained our model with particular values for peak and minimum firing rates as well as bin size. We used common values of $\lambda_{peak} = 74$ and $\lambda_{min} = 2$ for all units. All the differences in signal characteristics between individual units thus far have been in their preferred directions; i.e., the units were tuned to different movement directions. We have observed the effect of differences in preferred directions between units on the changes in MI due to pooling. Now, we look at what happens if the units within a population have additional differences in response features, such as the mean firing rates, or the scale factor of the tuning curves. We measure the latter feature by the quantity we call tuning depth, and define it as $\delta = \lambda_{peak} - \lambda_{min}$.

5.4.1 Effect of mean firing rate on information

In both the inter- and intra-cluster difference cases, the individual units all had roughly the same information content independent of preferred direction. As seen by comparing Fig. 5.8**b** vs **a**, one way information is increased is by pooling units with the same preferred direction. Since one obvious difference between the signals of a

pooled pair (ρ_{UX}) and each individual unit (ρ_i) is the mean firing rate, we considered the role that an increase in average firing rate plays in increasing mutual information.

As a way to observe the dependence of information on mean firing rate, $I(\theta;\rho_i)$ was computed as the mean firing rate increased. The mean firing rate λ_{avg}, or μ, was varied from 26 spikes/s to unrealistically high rates of $\sim$ 1000 spikes/s while the firing rate during movements in the anti-preferred direction, λ_{min}, was fixed at 2 spikes/s. Simulations were performed with $\phi_{intra} = 90^o$. Information in single units was found to increase monotonically as their mean firing rates were increased, as shown in Fig. 5.16, for a fixed bin size of 100ms.

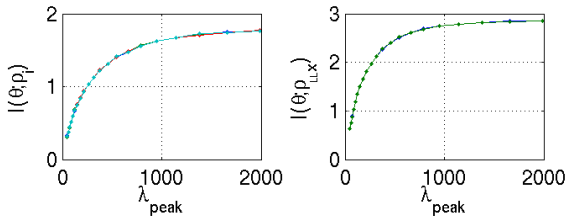


Figure 5.16: Peak average firing rates were increased while the base firing rate remained fixed at 2 spikes/s, thus increasing the variance and modulation depth simultaneously. Thus, λ_{avg} increases proportionally with λ_{peak}

.

However, we note an important caveat. Fixing the base firing rate while changing the mean causes the the peak average firing rate to increase proportionally with the mean rate from 50-2000spike/s ($\lambda_{peak} = 2\lambda_{avg} - 2$spikes/s). This effectively changes the mean, variance, and modulation depth simultaneously, since $\mu = \sigma^2 = \lambda_{peak}/2$ and $\delta = \lambda_{peak} - \lambda_{min}$. In order to isolate the effects of mean firing rate and modulation depth, we fixed the modulation depth while increasing the mean firing rate.

The results in the top row of Fig. 5.17 were generated for a fixed modulation depth of $\delta = 36$ spikes/s while the mean firing rate was varied. Instead of increasing, MI actually decreased with mean firing rate. Since MI no longer increases but actually decreases with increasing mean firing rate, the increase in information observed in

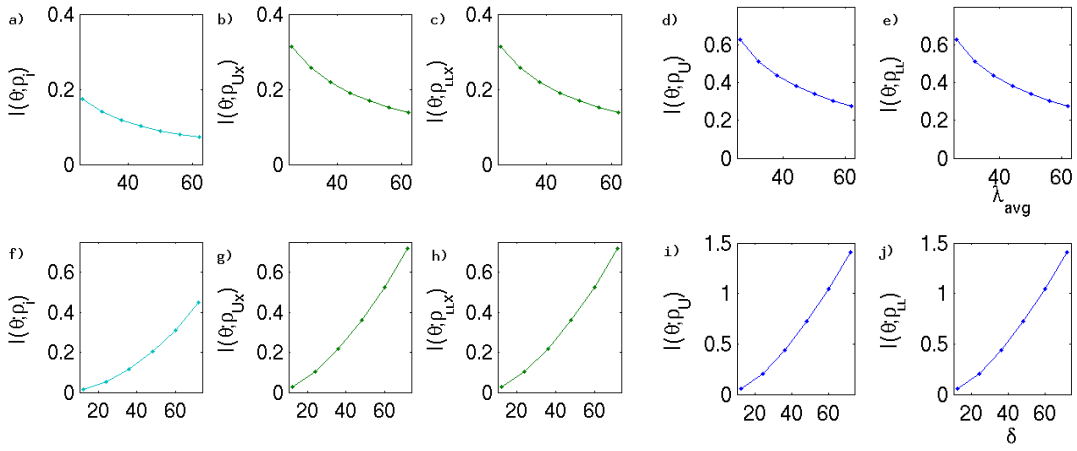


Figure 5.17: **a-e)** MI as a function of mean firing rate at a fixed modulation depth of 36. **f-i)** MI as a function of modulation depth for a fixed mean firing rate of 38. All results are shown for $\phi_{intra} = 0$, and $\phi_{inter} = 90^o$

Fig. 5.16 apparently was not primarily a function of increasing mean firing rate. We now observe the changes in MI for a fixed mean firing rate and changing modulation depth.

5.4.2 Effect of modulation depth

Modulation depth for a cosine-tuned neuron is a measure of the amplitude of modulation with movement direction and was defined as $\delta = \lambda_{peak} - \lambda_{min}$. Varying the modulation depth is equivalent to scaling of the tuning curve. When λ_{avg} was fixed at 38 spikes/s and δ varied, MI increased in an exponential fashion, with accelerating increases in MI as δ increased (bottom row of Fig. 5.17).

The surface plot in Fig. 5.18 displays the trend in 2-unit and 4-unit MI as both the modulation depth and mean firing rate vary. The graph appears to be truncated along one edge. That is because the spike rate was not allowed to decrease below 0. Thus, for a given mean firing rate, the depth could only be increased to a maximum of 2μ. The code running the computation iterated through different δ values for each μ value. To maintain a matrix structure when storing the data and an automated

process for computing the results, δ was allowed to increase beyond 2μ, and the true μ was shifted up slightly above the specified μ for that given outer iteration.

Aside from that truncation, the general trend is clear. Changes in MI occurred primarily along a diagonal axis. Information was maximized for the largest modulation depths with the lowest mean firing rate. MI peaked for the maximum δ and the lowest μ for that δ. For a given δ value, MI decreased slightly with increasing μ; it increased more dramatically for increasing δ at a fixed μ value.

This trend applied for all signals ρ_{UX}, ρ_{LLX}, ρ_U and ρ_{LL}, as shown by the similarity between the plots of Fig. 5.18. A comparison between plots on the left and right shows that MI in the pooled signals was affected similarly to MI in the sorted signals; a comparison between the upper and lower plots shows that MI in 2-unit signals was affected similary to MI in 4-unit signals. A similar trend was also seen when there were intra-cluster phase differences as opposed to inter-cluster differences (Fig. 5.19).

The trends in Δ_{U-LL} remained the same for different values of δ and μ. The 2-unit and 4-unit information loss due to pooling was affected very similarly by μ and δ, as indicated by the average percent difference between the 2-unit and 4-unit information loss: $\langle \frac{\Delta_{UX-LLX} - \Delta_{U-LL}}{\Delta_{U-LL}} \rangle = 0.5\%$. The average percent difference between $\Delta_{U-LL}^{\phi_{intra}=90^o} - \Delta_{U-LL}^{\phi_{inter}=90^o}$ was 100% while the average percent difference between Δ_{U-LL} at different values of μ was close to 0%. The information loss Δ_{U-LL} did not depend on μ or δ when the intra-cluster angular tuning differences were $= 0$ and inter-cluster difference was 90^o. The loss was 0 ($<< 0.001$) for all tested combinations of μ and δ. When $\phi_{inter} = 0$ and $\phi_{intra} = 90^o$, the loss did increase from -51% to -57% as δ and μ decreased (Fig. 5.20). Apparently, Δ_{U-LL} is more strongly affected by changes in angular tuning than by the values of δ or μ.

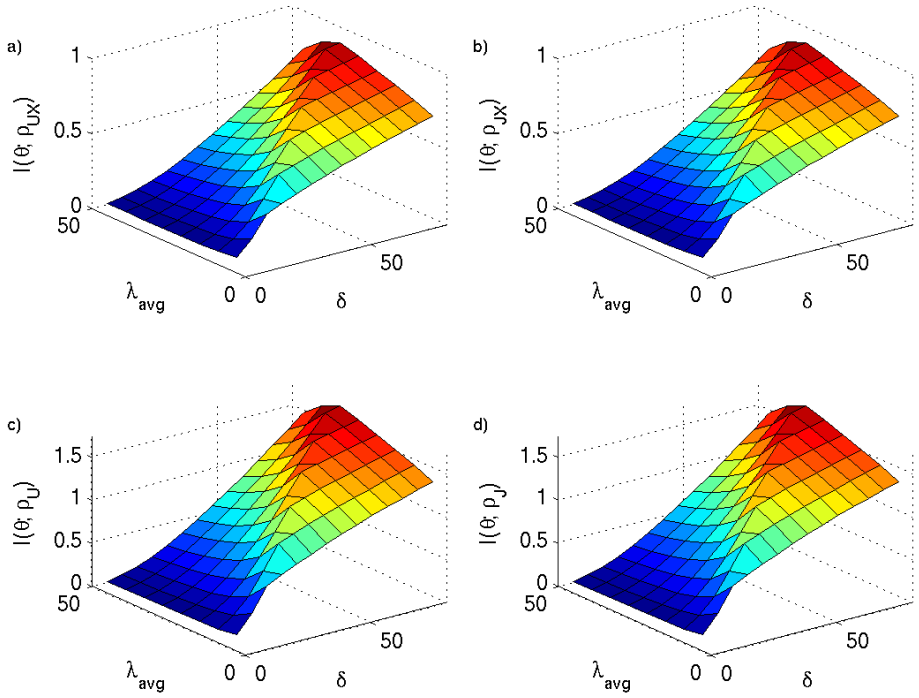


Figure 5.18: Mutual information in the 2-unit signals (**a-b)**) and 4-unit signals **c-d)** as a function of both mean firing rate λ_{avg} and modulation depth δ. $\phi_{inter} = 90^o$, $\phi_{intra} = 0^o$. Information in **a, c)** the pooled signals and **b, d)** in the sorted signals.

5.4.3 Dependence on bin size

Another way of affecting the spike counts aside from changing the mean firing rate is to change the bin size. We also wondered whether bin sizes affected the information gain of the labelled-line signals over the pooled signals. MI was numerically computed for bin sizes with $p_{capture} = .999$, yielding a maximum bin count for the multi-unit signal of 55 at a bin size of 50ms. At this range of response values, Matlab could only handle bin sizes up to 50ms before it ran out of memory. $p_{capture}$ was reduced to .995, .990, .950, and .900. The percent differences at the various $p_{capture}$ values were all $< 1\%$, and increased above 1% for the full joint information at $p_{capture} \approx .8$ The results are shown for $p_{capture} = .95$ (Fig. 5.21).

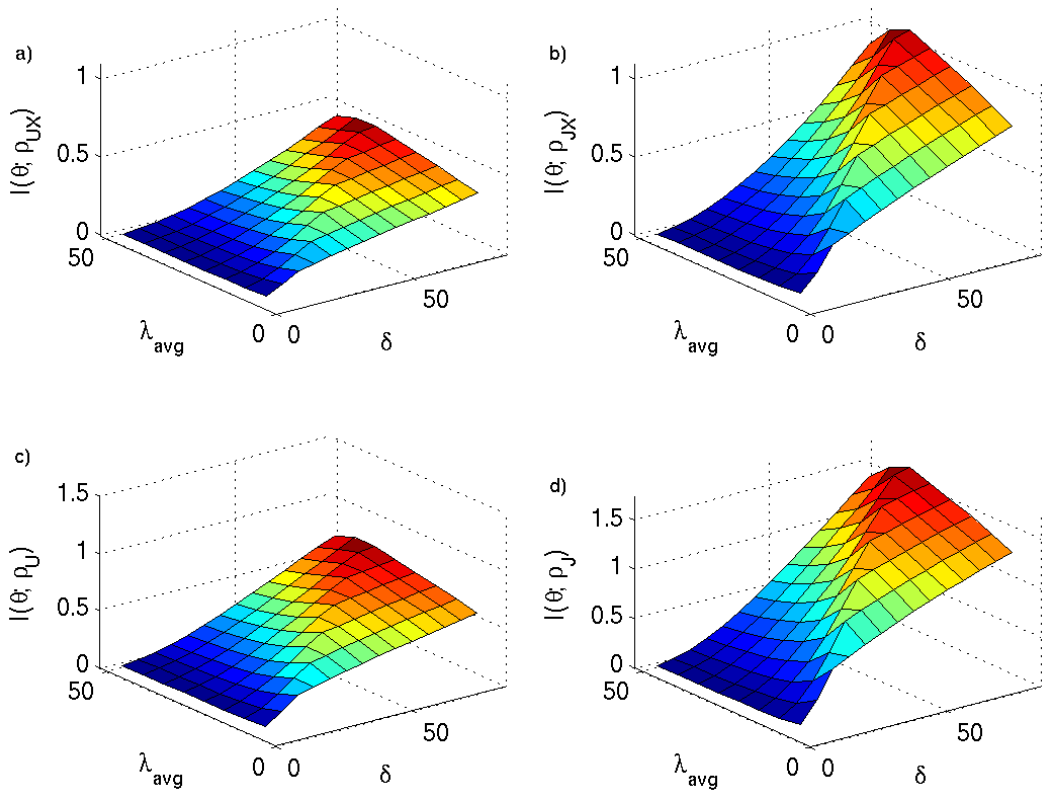


Figure 5.19: Mutual information in the 2-unit signals (**a-b)**) and 4-unit signals (**c-d)**) as a function of both mean firing rate λ_{avg} and modulation depth δ. $\phi_{intra} = 90^o$, $\phi_{inter} = 0$. Information in the pooled signals is shown in **a)** and **c)**, in the sorted signals in **b)** and **d)**.

The mutual information in the binned spike counts increased with bin size, but the information *rate* decreased. The monotonically increasing trend would be consistent with the relationship between the signal-to-noise ratio of a Poisson and its mean firing rate. This trend persists provided that the signal length in a given direction is long enough; i.e., that you could treat the signal when calculating information as a concatenation of homogenous Poisson signals. However, in biological signals, the signal is not a homogeneous Poisson process. We will address these issues and attempt to understand these trends in the following section.

MI computed from simulated signals generated similar results as numerical computations. Cosine-tuned spike trains were generated as before and binned in non-

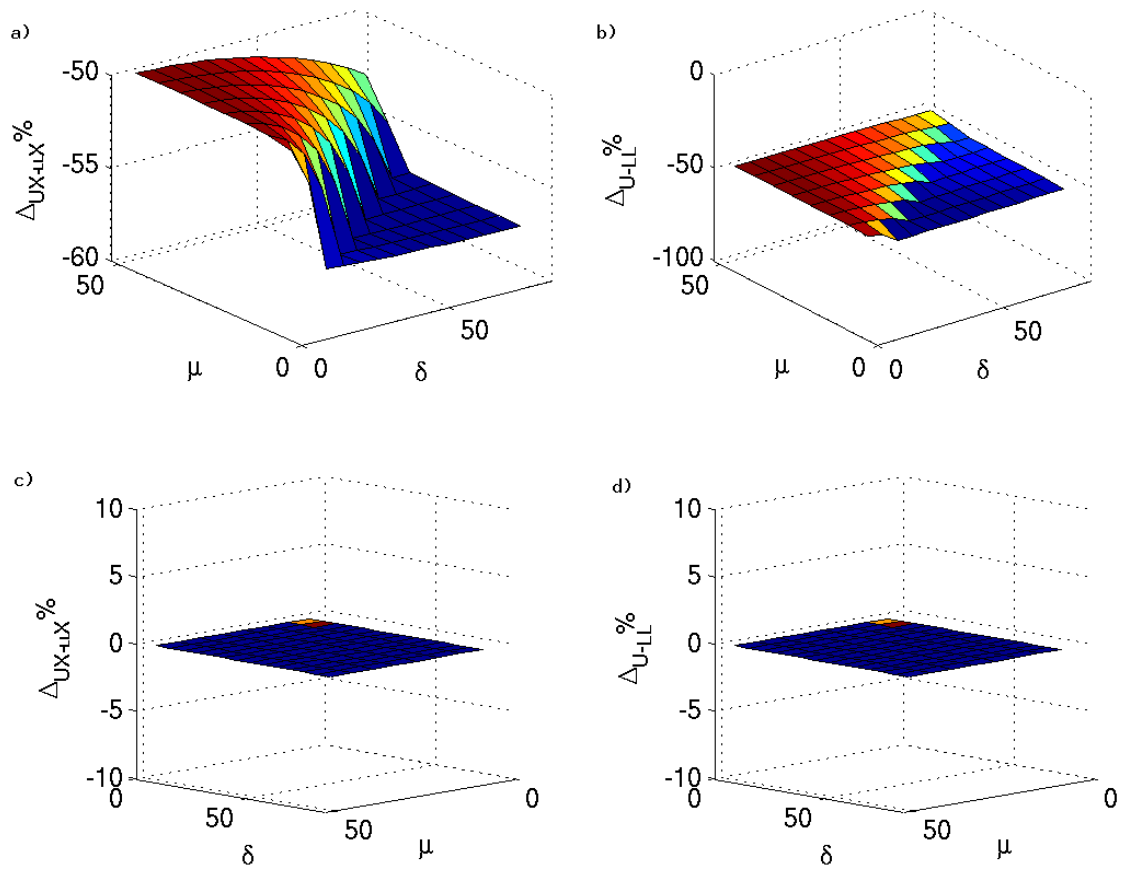


Figure 5.20: Percent difference in pooled and labelled-line mutual information as mean firing rate and modulation depth vary simultaneously. Shown for 2 cases: **a-b)** intra-cluster angular tuning difference is 90^o; 0 inter-cluster differences. (Note the difference in scale for the single-electrode MI loss Δ_{Ux-LLx}.) **c-d)** inter-cluster angular tuning difference is 90^o; 0 intra-cluster differences.

overlapping windows of different lengths. The results in Fig. 5.22 show virtually identical trends except that the information rate peaked at a bin size of 40ms. Other studies have indicated an optimal bin size for determining spike rates [83]. The curves in Fig. 5.21 do not show a decrease in information above a certain bin size. Perhaps, this is the difference between theory and practice. Theoretically, MI rates can increase for decreasing bin sizes until the bin size equals the sampling period; however, in practice, the spikes need to be summed over some reasonable period before information can be obtained. Regardless of these interpretations, the answer to the

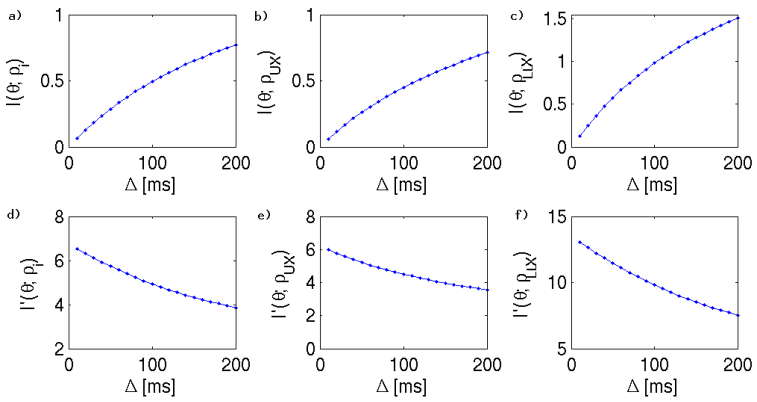


Figure 5.21: **a-c)** MI (bits) in binned 1- and 2-unit responses as the bin size was varied. **d-f)** The corresponding mutual information rate (bits per s) decreased with increasing bin size. All results shown for $\delta = 36$ and $\lambda_{min} = 2$ for 0 inter-cluster phase differences and $\phi_{intra} = 90^o$.

question raised in chapter 2 about the difference in information loss in sorted and non-sorted signals caused by binning remains the same. Binning affected sorted and non-sorted signals similarly, such that $\Delta_{UX-LLX} = -53.9 \pm 0.58\%$ across all bin sizes ranging from 10ms to 200ms, and only varied by 1.7% across this range.

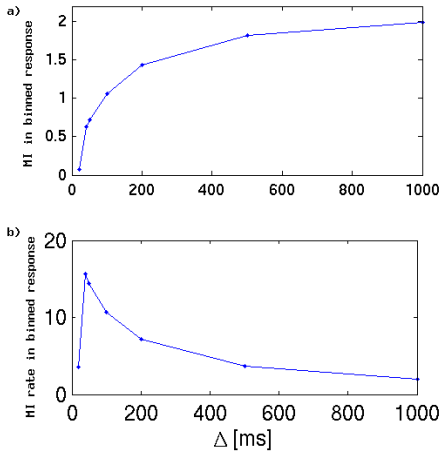


Figure 5.22: **a)** Mutual information in simulated binned single-unit responses ($I(\theta; \rho_i)$ in bits) as the bin size is varied. The modulation depth was $\delta = 72$. **b)** MI rate ($\dot{I}(\theta; \rho_i)$ in bits/s) for the same binned responses as in a).

5.4.4 Relation to SNR

We have observed that information represented in labelled-line pairs was lowest when the two neurons within an electrode carried redundant information (i.e., when the electrodes had equal preferred directions) and increased as the preferred directions of the two electrodes grew out of phase until $I(\theta; \rho_{LL})$ peaked at a within-electrode angular difference of $\pm 90^o$. This can be understood by the illustration in Fig. 5.23 of pairs of cosine curves as the phase is varied. Each pair of dots at a given θ represents the 2-unit response. As the disparity between these responses grows, the likelihood of overlap between response patterns diminishes. As Fig. 5.10 **e)** and **f)** revealed, the non-sorted signals from both electrodes transmit as much as the maximimum information for a 4-unit sorted population when the pair of pooled units on the same electrode reinforced one another while the two electrodes carried diversified tuning information.

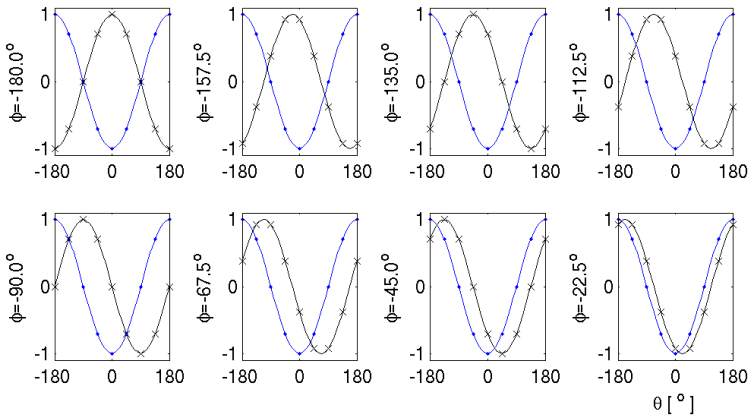


Figure 5.23: Pairs of cosine curves, representing ideal normalized averaging firing rates of pairs of neurons with a angular difference in their preferred directions. Viewing the pair of ideal responses as the phase changes helps visualize the relationship between redundancy and phase.

Thus, we can see how shifting the phase increases the separability of responses. When the cosine curves are 0 or 180^o apart, there are only 5 possible unique responses that the pair can create. As the phase between the cosine curves increases, the number

of unique responses jumps to 8. Shifting one cosine-tuned response out of phase with the other causes information capacity to increase.

However, MI in the modelled cosine-tuned responses did not jump from $\log_2(5)$ bits of MI, when a pair units are completely redundant or completely out of phase, to $\log_2(8)$, when the pair is slightly out of phase, as it would if they were pure cosine-tuned responses. MI instead grew gradually with shifting phase between the curves. Since MI should jump if the responses had no noise, but shift gradually with noise, the noise due to the Poisson nature of the signals must play a role in the gradual shift in information. The pairs of cosine curves in Fig. 5.23 show how shifting the phase between the curves allows for more noise to be present and maintain the ability to distinguish between different responses at different values of θ. We now view the trend in MI with varying δ and μ from sections 5.4.1 and 5.4.2 in light of these explanations.

In communications and signal processing, SNR is a measure of the quality of the signal and how well information can be transmitted across a channel. It thus seemed reasonable to attribute the increase in MI with λ_{peak} to an increase in SNR for a Poisson signal. The raw amplitude SNR was defined as the ratio of the mean of the spike count signal to the standard deviation of the signal as shown in Eqn. 5.18.

$$SNR_{raw} = \frac{\mu}{\sigma} \qquad \Rightarrow \qquad SNR_{dB} = 20 \log \frac{\mu}{\sigma} \tag{5.18}$$

We already know some properties about Poisson signals and their signal-to-noise ratios. For a Poisson signal, $SNR = \frac{\langle\lambda\rangle}{\sigma_\lambda} = \sqrt{\langle\lambda\rangle}$, since the variance is equal to the mean by definition. Thus, SNR would be expected to increase as the $\sqrt{\lambda_{avg}} = \sqrt{(\lambda_{peak} + \lambda_{min})/2}$. In Fig. 5.16, MI bore a visual resemblance to a square-root dependency on λ_{avg}. The dependency of information while mean firing rate and variance changed was suspected to be due to the direct relationship of signal-to-

noise ratio (SNR) on the λ parameter, or the mean, of a Poisson signal. It was also suspected that the SNR of summed Poisson signals could influence MI in a fashion reminscent to the effect noted in the integrate-and-fire neurons of Sec. 4.5.

MI in the cosine-tuned signals, however, began to level off as it approached 1.77 bits for the single-unit as shown in Fig. 5.16. When information was plotted against SNR, the relationship was nonlinear. To look at the connection between SNR and information in these cosine-tuned signals, hypothetical Poisson spike count signals were generated for different peak average firing rates (λ_{peak} in spikes/s) and at different bin sizes (Δ in ms). SNR was calculated and plotted as a function of the peak average firing rate for the Poisson signals at a bin size $\Delta = 40ms$ as the average binned count varied, as shown in Fig. 5.24**a**. These ratios were calculated for a range of firing rates, including unrealistically high values, to demonstrate the relationship between SNR and average firing rate. SNR_{raw} increased as the square root of the average firing rate, or equivalently $SNR_{dB} \propto \log \lambda_{peak}$. This relationship is demonstrated in Fig. 5.24.

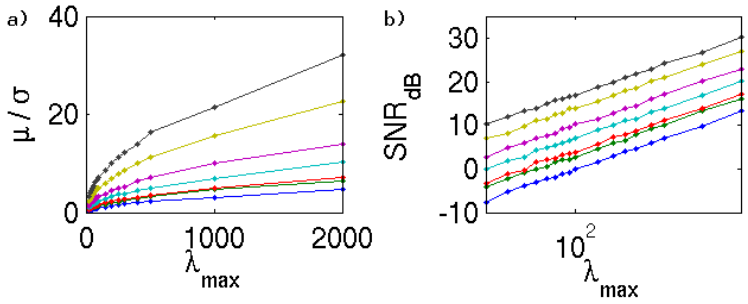


Figure 5.24: A graphical illustration of the known relationship between SNR of a homogeneous Poisson signal and its firing rate. **a)** The mean to standard deviation ratio is plotted against the peak average firing rate in spikes per second. **b)** SNR shown in dB.

In comparison, MI in simulated single-unit cosine-tuned spike trains, shown in Fig. 5.16, increased monotonically with a decreasing rate as λ_{peak} increased. However, it was not proportionally related to the $\sqrt{\lambda_{peak}}$, as shown by the nonlinear relationship

when MI was plotted against SNR. More importantly, MI *decreased* as mean firing rate increased for a fixed modulation depth (section 5.4.1).

One way in which the cosine-tuned responses in our model differ from these hypothetical Poisson responses is that their firing rate is modulated; i.e., the cosine-tuning makes the response inhomogenous. Although SNR for any given direction would increase with increased firing, the inhomogeneity of a cosine-tuned response seems to cancel any necessary relationship between SNR and increased firing based on theoretical Poisson properties.

We see that the cosine-tuning and Poisson nature of the signals creates an interdependency of the effects of modulation depth and mean firing rate on MI. If modulation depth was the sole determinant of information capacity, then we would expect MI not to be affected by varing the mean rate while modulation depth was kept constant. But this was not true.

When the modulation depth was kept constant at $\delta = 36$ and the mean firing rate μ was varied, the information decreased as the mean rate increased. As the mean rate increased, the variance (noise) at each value of the target signal (or direction of movement) increased, but since the modulation depth was the same, the difference in signal levels between target values was the same. Essentially, the overlap between responses at different signal levels increased as the average firing increased. This increasing overlap corresponded with the decrease in information as the average firing rate increased.

Mutual information displayed exponential growth with modulation depth, as long as the mean was high enough to allow as large a response range as specified by δ. In other words, when $\mu < \delta/2$, the bin counts would have to be negative in order to really have a modulation depth of δ. This is why the information did not continue to grow exponentially as μ was lowered (Fig. 5.18). When the mean firing rate was kept

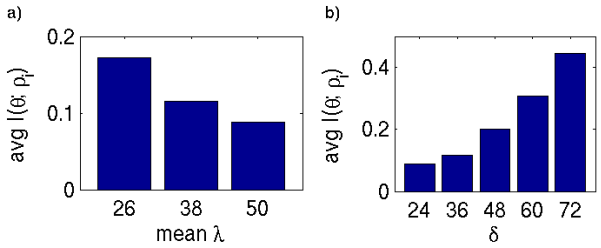


Figure 5.25: a) Mutual information in single units for 3 mean firing rates, at a constant modulation depth δ of 36. Bar height indicates average for all units at given mean firing rate and across all ϕ; standard deviations were all < 0.001. b) Mean firing rate is fixed at 38; average information in single units increases with δ

constant at $\lambda_{avg} = 38$ spikes/s, and the modulation depth δ was increased, the spread of spike counts between responses increased. The increasing spread corresponded to the increase in information.

We can now understand how the cosine tuning and the Poisson nature interact to cause MI's dependency on δ and μ. We still note other differences in behavior between SNR and MI with μ. SNR continues to increase without limit, while MI reaches an asympotic limit. For a uniformly regularized output space, there are only 5 unique values a single unit can take, since the cosine-tuned response is not a 1:1 function. The information did not go beyond 1.77 bits even though with a single evenly-sampled cosine curve with no noise contains $log_2(5) = 2.32$ bits. This is because the spike train from a single unit could not fire more than 1 spike per ms. Thus, the spike counts even during a movement in the preferred direction never exceeded Δ_{ms}; as λ_{peak} increased, the bin counts saturated as the largest noise margin was reached, and the information reached a peak. If λ_{peak} was increased much further, eventually the spike counts during movement in the other directions would also saturate and information should decrease.

Also, the information *rate* is limited by the sampling rate. The number of bits per sample cannot exceed 3, so if the sampling rate is 1000Hz, even with a noiseless system with maximum firing rates of 2000Hz, the information rate can be at most

3Kbits per second; i.e., it doesn't matter how great the information content in a neural spike response is if the acquired response has a limited information-carrying capacity. Although the information rate could increase without limit if the sampling period shrank to an infinitesimally small duration, the overall average information could never exceed 3 bits, the entropy of the target signal.

5.4.5 Average firing rate and modulation depth differences within clusters

In the previous analysis, each of the four neurons were assigned equal grand mean firing rates. This is not a likely situtation *in vivo*. Here, we examine the effect on information in a population with different mean firing rates among neurons.

One unit within each pair was assigned a mean firing rate of 38 spikes/s. The second unit within the pair was assigned a firing rate of $\lambda_1 + \Delta_\mu$. MI did change when the difference in average firing rates within pairs was increased from 0 to 12. However, MI did not seem to be primarily dependent on average firing rate differences within pairs. Even when the difference in average firing rate within pairs was unchanged, the information in pairs and in the 4-unit population varied, as shown in Table 5.3.

λ_1^{avg}	λ_2^{avg}	Δ_μ	I_{LL}	I_{LLA}	I_{LLB}	I_1	I_2	I_3	I_4	I_U	I_{UA}	I_{UB}
38	38	0	0.44	0.23	0.23	0.12	0.12	0.12	0.12	0.21	0.11	0.11
50	38	12	0.39	0.21	0.20	0.088	0.12	0.088	0.12	0.18	0.097	0.097
38	26	12	0.53	0.29	0.29	0.12	0.17	0.12	0.17	0.24	0.13	0.13

Table 5.3: Information in pairs and 4 units as the mean firing rates as well as the difference mean firing rates of units on the same electrode varied. The mean firing rates of units on electrode A were equal to those on electrode B ($\lambda_1^{avg} = \lambda_3^{avg}$, $\lambda_2^{avg} = \lambda_4^{avg}$). The modulation depth for all units was fixed at 36.

Modulation depth was then varied within clusters, so that the modulation depth difference was defined as $\Delta_\delta = |\delta_1 - \delta_2|$. Mean firing rates of all units were fixed at 38 spikes/s. Fig. 5.26 reveals that the different Δ_δ values had little effect on MI loss

due to pooling. Both MI in pooled and sorted signals increased with decreasing Δ_δ but proportionally with Δ_δ. Since δ_1 was fixed at 72 and $\delta_2 = \delta_1 - \Delta_\delta$, increasing Δ_δ equated to lowering the modulation depth of one of the units in each pair, thus explaining the increase in MI with decreasing Δ_δ.

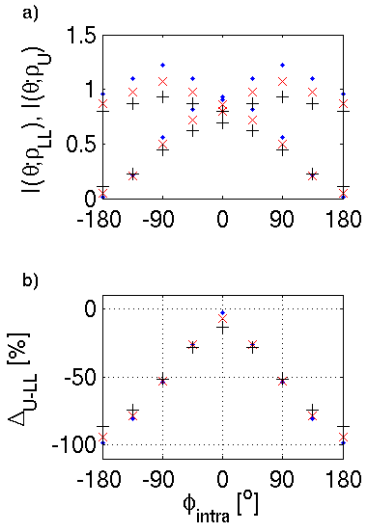


Figure 5.26: Relative effect of modulation depth differences within electrodes on pooled and labelled-line MI. **a)** I_{LL} and I_U when differences in modulation depth are 12 ('·', blue), 24 ('x', red), and 36 ('+', black). **b)** Δ_{U-LL} at each of the modulation depth differences; points are overlayed on each other, revealing the different Δ_δs cause indistinguishable differences in MI loss.

The intra-cluster phase difference case was again simulated with the peak firing rates differing across units. Mean firing rates and modulation depths were chosen so that every neuron had a different μ and δ. One population of 4 neurons was simulated with a higher level of firing than a second; and the same difference in δ was used for the corresponding electrode in each population, such that the Δ_δ^As were equal across the two populations as well as the Δ_δ^Bs. The base firing rate was fixed at 2 spikes/s. The first set of average firing rates and modulation depths was made to be $\{\lambda_{avg,1} = 35, \lambda_{avg,2} = 5, \lambda_{avg,3} = 25, \lambda_{avg,4} = 15$ spikes/s $\}$ and $\{\delta_1 = 66, \delta_2 = 6, \delta_3 = 46, \delta_4 = 26\}$ spikes/s. A second population of units was simulated with a set of higher λ_{avg} and δ: $\{\lambda_{avg,1} = 40, \lambda_{avg,2} = 10, \lambda_{avg,3} = 30, \lambda_{avg,4} = 20$ spikes/s $\}$, $\{\delta_1 = 76, \delta_2 = 16, \delta_3 = 56, \delta_4 = 36\}$ spikes/s. These values are shown in Table 5.4 for reference.

The results of these experiments followed a similar trend as the earlier results:

	λ_1^{avg}	δ	λ_2^{avg}	δ	Δ_δ^A	λ_3^{avg}	δ	λ_4^{avg}	δ	Δ_δ^B
low λ	35	66	5	6	60	25	46	15	26	20
high λ	40	76	10	16	60	30	56	20	36	20

Table 5.4: Combinations of mean firing rates and modulation depths for an example constructed to demonstrate the effects of differences in μ and δ among neurons within a population. The *high* λ set has higher μ and δ than the *low* λ set. In both sets, one unit on electrode A has higher firing rates than one on electrode B while the other unit on electrode A has lower firing rates than the other unit on electrode B. The results from these populations are shown in Fig. 5.27.

joint pair information increased as the directional tuning became less redundant, peaking when $\phi_{intra} = 90^o$; the union pair information had no information when the unit responses within a cluster were completely out of phase but peaked as the unit responses became redundant and converged with the joint pair information.

In both sets, cluster A's union mutual information ($I(\theta; \rho_{UA})$) was always higher than cluster B's ($I(\theta; \rho_{UB})$), (see Fig. 5.27 blue versus green); but the increased absolute modulation depths did not cause the union MI to be consistently greater than the lower modulation depth set (Fig. 5.27, middle column, 'x' vs '$\cdot$'). Conversely, the joint pair MI was higher for the set of higher modulation depths, but cluster A joint information ($I(\theta; \rho_{LLA})$) was not necessarily higher than cluster B's joint MI ($I(\theta; \rho_{LLB})$).

For every unit in the *low* λ_{avg} set, there was a unit in the *high* λ set with higher mean and modulation depth, thus resulting in the former set having lower joint MI than the latter set. Within a set, one unit in cluster A had a higher λ_{avg} and δ than a unit in cluster B, while the other unit in cluster A had a lower λ_{avg} and δ than the other unit in cluster B, thus making the difference in labelled-line MI between cluster A and B variable.

The converse relationship was true for MI in pooled responses. We interpret this result in light of the results from Sec. 5.2. As the tuning curves of individual responses shifted out of phase, pooling caused increasing cancellation of the responses, resulting

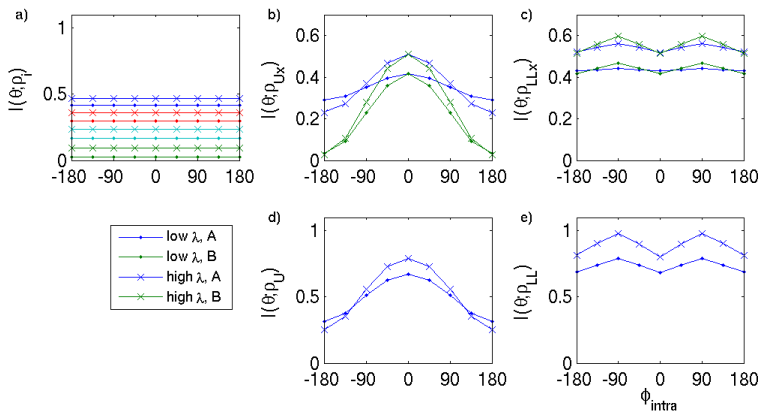


Figure 5.27: MI in 2-unit and 4-unit signals when the mean firing rates among units are different. Two sets of the 4-unit populations were simulated. One set (the "low λ" set) had mean rates of 35 and 5 spikes/s on electrode A, and 25 and 15 spikes/s on electrode B, as shown in Table 5.4. The other set ("high λ") had mean rates of 40 and 10 spikes/s on electrode A, and 30 and 20 spikes/s. MI as a function of intra-electrode preferred direction differences is shown for **a)** the individual units, **b)** the pooled signals on each electrode, **c)** the labelled-line signals on each electrode, **d)** the labelled-line signals on both electrodes, and **e)** the pooled signals on both electrodes. The dots ('$\cdot$') demarcate MI in the *low* λ set; the '$\times$'s demarcate MI in the *high* λ set. For the 2-unit information, green markers distinguish electrode B from electrode A, shown by the blue markers.

in greater MI loss. In the current analysis, the pooled MI did not decrease as much with increases in phase difference when there was greater disparity in tuning depth between the pooled responses. We interpret these results as showing that when the tuning depths are more disparate, there is less cancellation as tuning of responses grow out of phase, resulting in less MI loss due to pooling. The comparison between the *high* λ set and the *low* λ set was also a demonstration of how greater modulation depths increased sensitivity of pooled responses to phase differences.

Since the differences in modulation depths affect pooled and labelled-line MI conversely, as described above, we can understand the differences in MI loss due to pooling, as shown in Fig. 5.28. The larger differences in modulation depths caused smaller cancellation effects when pooling units of differing preferred directions (blue markers show lower MI losses for electrode A than for electrode B, shown in green, in the Δ_{UX-LLX} plots). When differences in modulation depth were equal, the larger

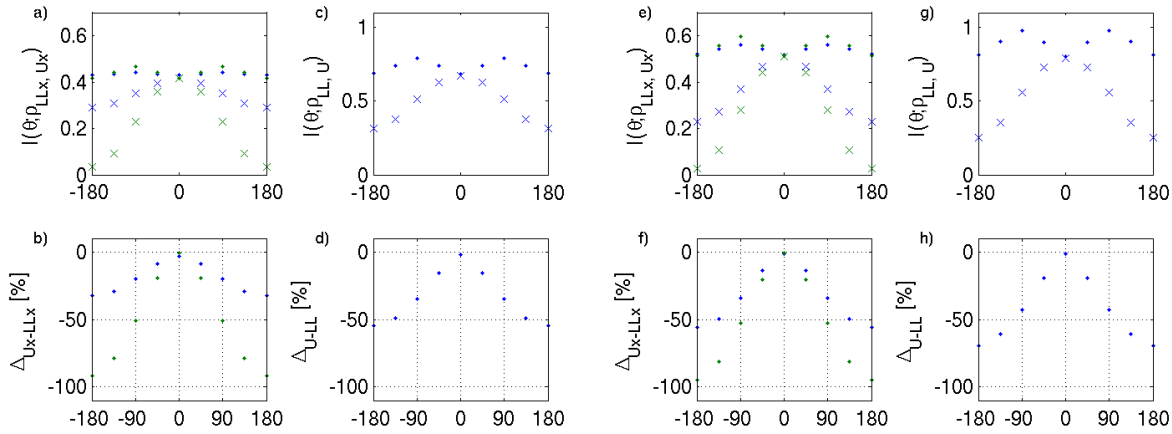


Figure 5.28: A comparison of information in the joint and union population as ϕ_{intra} varies for differing sets of λ_{avg} and δ, whose values are given in Table 5.4. **a-d)** *low* λ set. **e-h)** *high* λ set. In the top row (a, c, g, e), 'x' distinguishes labelled-line MI from pooled MI, shown as '$\cdot$'. Green distinguishes MI on electrode B from electrode A, which is shown in blue.

proportional difference in modulation depth caused less MI loss due to pooling (left panel showing the *low* λ set vs. right panel showing the *high* λ set).

5.4.6 Networking of noisy neurons

The total information available to the BCI system (Fig. 2.2) is equal to the entropy of the target signal, which in these simulations was 3 bits. The maximum information of a single perfectly cosine-tuned unit with no noise would be $log_2(5)$ bits. The information of two non-redundant units does not reach the maximum available information of 3 bits due to Poisson noise, but how close the information in these signals gets to the capacity of 3 bits depends on the angular tuning differences, average firing rates, bin sizes, and scaling of the tuning curve.

In Sec. 5.2.1, the information of the two-neuron system was computed as the difference in preferred direction was varied. $I(\theta;\rho_{LLx})$ vs. ϕ_{intra} showed the effects of adding a second neuron to the system but keeping the neurons labelled. There was approximately a 60% gain in information when the two neurons were exactly in phase

(preferred directions equal) or exactly out of phase (preferred directions 180^o apart). There was an additional gain in information as the separation between preferred directions moved toward 90^o, peaking when $\theta_{PD,2} = \theta_{PD,1} \pm 90^o$. This increase and peaking of information is consistent with the redundancy in a pair of cosine curves out of phase (see Fig. 5.23). A pair of these perfectly cosine-tuned responses taken jointly (Fig. 5.23) could fully represent the maximum 3 bits of mutual information available in this center-out task. The neural responses in these simulations did not reach this maximum due to Poisson noise.

Even with a perfect response, (i.e. using the lambda parameter of the Poisson signal as the neural response signal), the MI between the labelled-line response from a pair of units and the movement direction is 2.25 bits. A pair of units can encode the full information in the target signal if their preferred directions are slightly out of phase but not completely in phase or completely out of phase, i.e. for all ϕ except 0 in the range (-180^o 180^o), endpoints non-inclusive. The results illustrate the extent to which population codes can create reliable information sources even from very noisy components. The single-unit information reveals that a single neuron is a noisy information source when its response is Poisson at firing rates and integration times that can realistically be found in biological neurons: the MI in a unit with an average firing rate ranging from 2-74 bits/s and a bin size of 40ms is only 0.45 bits, 15% of the information capacity. Redundancy can increase information by 56%: $I_{UX} = 0.7$bits for $\phi_{intra} = 0$. Information amongst pooled clusters with diverse tuning can attain more than a 200% increase over a single unit: $100\frac{I_U - I_i}{I_i} > 200\%$ for $\phi_{inter} = 90^o$ and $\phi_{intra} = 0^o$. This is the same information also attained by 4 separate but diversely tuned units. Because the sum of Poisson signals with lambda parameters for each signal i given by λ_i is a Poisson signal with $\lambda = \sum_i \lambda_i$, the MI in a pooled signal of 2 units with the same tuning can be closely estimated by the MI in a single unit with

the sum of the two lambda parameters.

If the information increases as tuning curve is scaled up, then information should also increase as the number of pooled units increases. We examined the requirements of a population of the modelled cosine-tuned signals for them to reach more of the full 3 bits of information. For example, we are interested in how many neurons, and with what modulation depths, could attain the 3-bit capacity.

To estimate the modulation depth that would be required to reach a specified level of MI, e.g. 95% of capacity $\equiv$ 2.85bits, MI in pairs of units was numerically computed for increasing values of δ. Figure 5.17 showed how MI increased with δ for fixed μ. δ was set to equal the maximum modulation depth for a given μ, as the level was shifted. It took a $\delta = 1200$ to reach the 95% capacity level of MI.

When the preferred directions of units were spaced uniformly, the labelled-line population MI seemed to increase linearly with n, the number of neurons in the population, with a linear regression yielding an $R^2 = 1.00$. However, we must be careful about using this linear fit to determine the number of neurons needed to achieve a certain level of MI by extrapolation. After all, even a logarithmic curve can appear linear in certain small regions of the curve. However, for 1-6 neurons, the computed MI values were well fit by a linear regression. The linear regression for MI in n neurons with $\lambda_{min} = 2$ and $\lambda_{max} = 72$ yielded $\hat{I}(n) = 0.30n + 0.15$, $R^2 = 1.00$, $p \ll 0$. Six neurons with these parameters could attain an MI of 1.93 bits. If this linear trend continued, MI=3bits at $n = 9.6$. We know this linear trend could not continue past this point, since MI cannot exceed 3 bits in these center-out reaching tasks. We also would not expect most units acquired to have such large modulation depths. With the typical mix of neurons in biological recordings, we might expect on the order of tens of channels of cosine-tuned spike activity but likely within a range of 20-30 neurons.

5.5 Tolerances of error, directional tuning differences, tuning depths

We have examined the effects on MI of pooling units within a cluster for changes in MI along a single dimension. Those results give us confidence in our intuitive understanding of how MI should change with each of the individual parameters we have considered thus far: ϕ_{inter}, ϕ_{intra}, ε, λ_{avg}, and δ. Now, we consider the effects of pooling for different combinations of these parameters. We can view the changes in MI and Δ_{U-LL} for simultaneous changes in pairs of variables using parametric plots.

Two-dimensional contour maps were created which showed the iso-curves of percent difference between MI in multi-unit non-sorted and sorted signals ($\Delta_{U-LL}\%$) for pairs of parameter values, including ϕ_{intra}, ϕ_{inter}, $\varepsilon_{\%}$, λ_{avg}, and δ. These contour plots tell us what differences in pooled and labelled-line information we can maintain for various combinations of parameter values. In order to create contour maps for information loss as a function of percent spike error and angular tuning differences, MI was calculated in simulated signals at several values of ϕ and ε. At each ϕ value, exponential curves were fit to MI in ρ_{UX}, ρ_{LLX}, ρ_{U} and ρ_{LL}. Δ'_{U-LL} and Δ'_{UX-LLX} were calculated according to the fit for $\varepsilon_{\%}$ ranging from 0-50% at each simulated ϕ value, as described in sec. 5.1.6 and as demonstrated in Sec. 5.3. To give a finer mesh of independent variables, Δ'_{U-LL} (and Δ'_{UX-LLX}) were linearly interpolated between the ϕ values.

We observed in chapter 4 that MI loss due to pooling was directly related to the similarity in MI between the units that were pooled. We would like to know how much difference between units can be allowed to sustain specified amounts of MI loss. Thus, we consider the parameters which could cause differences between pooled units and thereby affect MI differences in sorted and non-sorted signals. As an example, we consider how the difference in the MI of sorted and non-sorted pairs varies with

differences in preferred direction and differences in modulation depth. The contour plot in Fig. 5.29 reveals that differences in MI for a sorted and non-sorted pair of units (Δ_{UX-LLX}) grew as $|\phi_{intra}|$ grew from 0^o to 180^o, as shown already in sections 5.2.1 and 5.2.2. On the other hand, it changed little for varying differences in modulation depth (Δ_δ) especially for smaller ϕ_{intra}. A notable characteristic in these plots is the change in the curvature of the contour lines between $|\phi_{intra}| < 90^o$ and $|\phi_{intra}| > 90^o$. Consistent with the results of sec. 5.4.5 and Fig. 5.28, as the difference in modulation depth increased, pooled signals could withstand larger angular tuning differences for the same level MI loss when the pooled neurons were non-redundant ($|\phi_{intra}| > 90^o$). The converse was true for more redundant neurons ($|\phi_{intra}| < 90^o$).

As noted in Sec. 5.4.2 (compare also Fig. 5.8d and e to a and b), differences in preferred direction have a larger impact on MI than differences in average firing rate and modulation depth. An intuitive explanation of this is that information in cosine-tuned neurons is primarily encoded through the preferred direction of the neuron. If we were told the preferred direction of a neuron with no noise, we could know based on its response, 1 of 2 directions in which the movement was made. If we were told the scale factor of the neuron, seeing its response would not tell us the direction of movement. We focus now on the factors that caused the most significant changes in MI: angular tuning differences and the amount of spike error.

MI was computed for the case of only inter-electrode differences in preferred directions and not intra-electrode differences, as in Sec. 5.2.2 (Fig. 5.10c and d), but with sorting error added. MI in the non-sorted (I_U) population exhibited the same behavior as in the sorted population (I_{LL}) as a function of inter-electrode phase differences and percent spike error, as long as there are no differences in preferred directions between pooled units, as shown in the surface plots of figure 5.30. MI exhibits the same peaks at $\phi_{inter} = \pm 90^o$ and valleys at $\phi_{inter} = \pm 180^o$, and this same curve

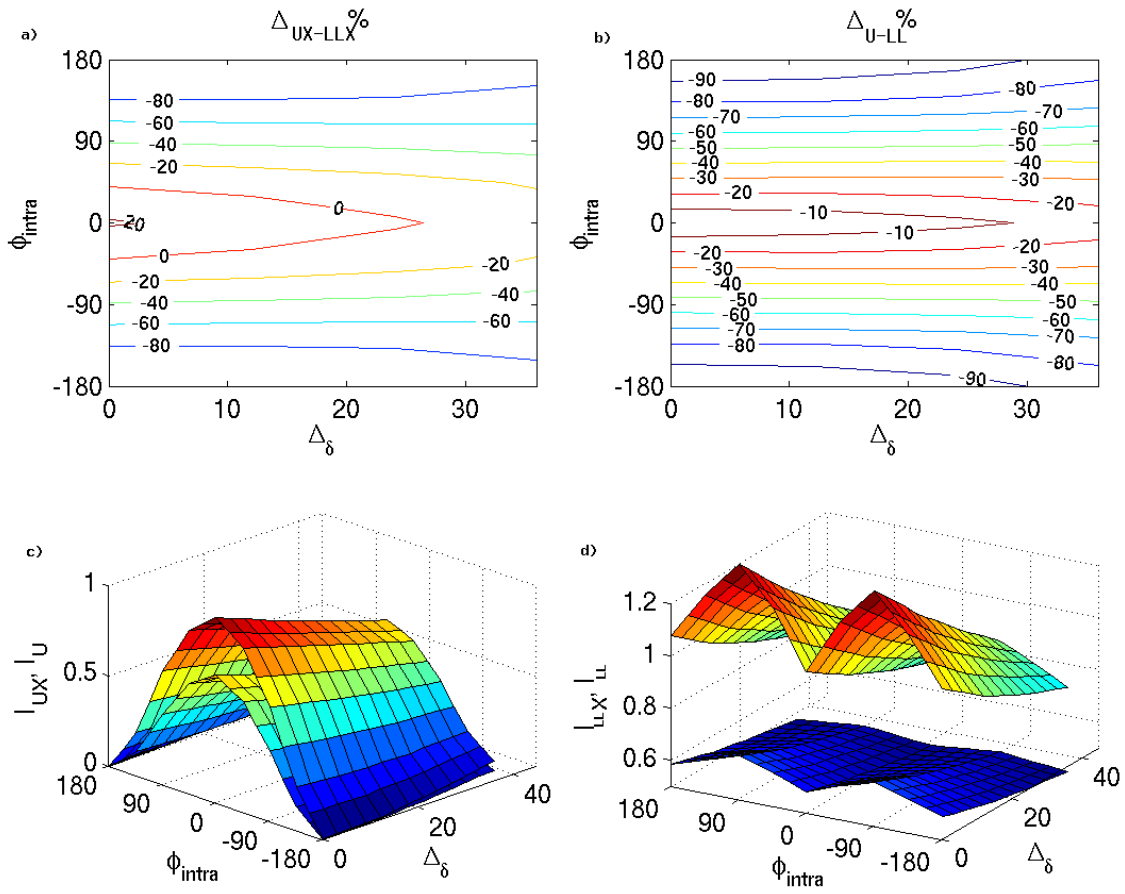


Figure 5.29: A comparison of information in pooled and labelled-line signals as differences within pooled pairs are varied. In these plots, the two parameters varied simultaneously are differences in preferred direction and differences in modulation depth. **a)** Contour plot of 2-unit signals, showing what differences in modulation depth and preferred direction can be sustained for a given percent information difference between a pooled pair and labelled-line pair ($\Delta_{UX-LLX}\%$). **b)** Similarly, for 4-unit signals. **c, d)** The effects of these modulation depth and preferred direction differences have the same relative effect on 2-unit MI as on 4-unit MI.

decays exponentially with $\varepsilon_{\%}$. In this case, MI is actually increased by leaving out spike sorting if sorting were to introduce spike error, because of the rapid exponential decay in MI with sorting error, as revealed in the contour plots at the bottom of the figure. Between 2 and 4% gain is possible for any inter-electrode phase differences when sorting creates 5% spike error. This is true for the perhaps unrealistic case of 0 difference in angular tuning for neurons on the same electrode and 0 detection error. It was possible for a non-sorted signal without any spike error to maintain the MI

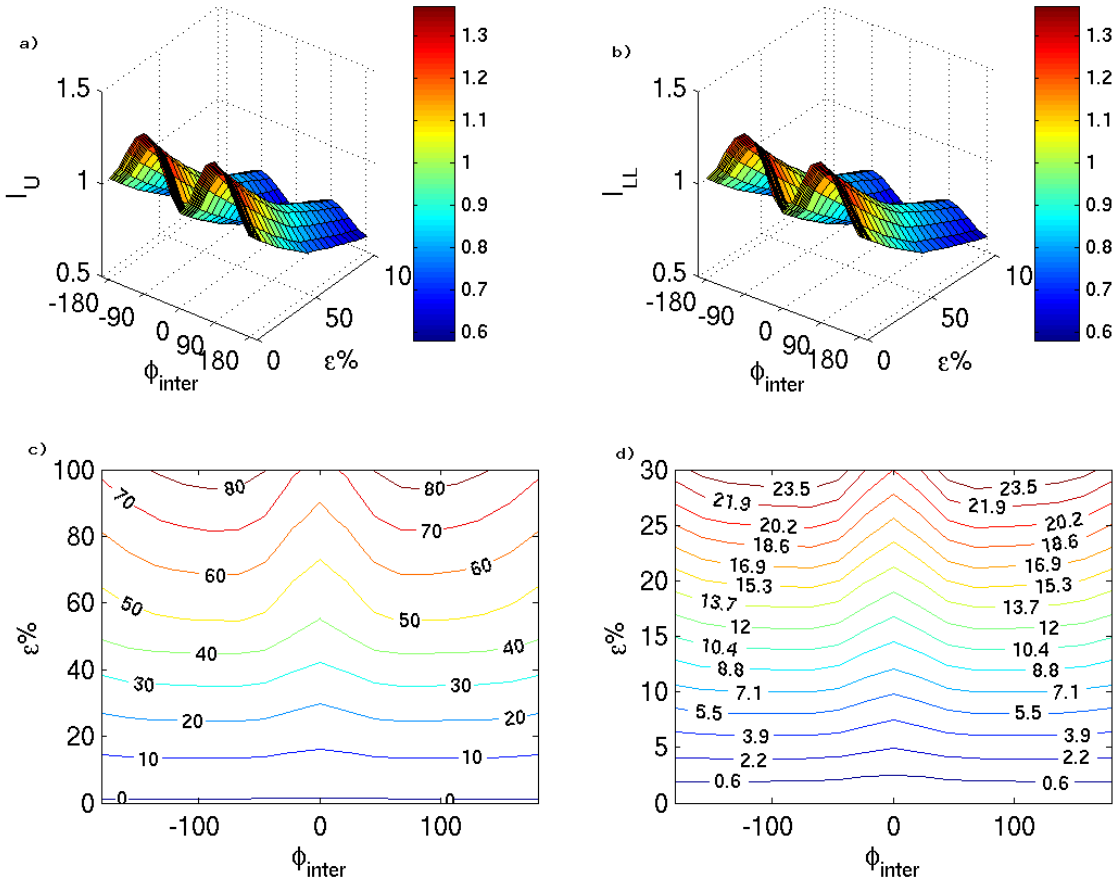


Figure 5.30: Mutual information as a function of inter-electrode preferred direction differences (ϕ_{inter}) and percent spike error ($\varepsilon_{\%}$) at a fixed ϕ_{intra} of 0^o for the non-sorted (**a**) and sorted population signals (**b**). Contour plots of the MI difference between sorted and non-sorted signals is shown on the bottom. The iso-contour lines for $\Delta_{U-LL}\%$ is shown for percent errors ranging from 0 to 100% (**c**) and from 0 to 30% (**d**).

in a corresponding sorted signal and to even have a 5.5% gain over a corresponding sorted signal with 7-10% sorting error depending on the inter-electrode tuning difference (Fig. 5.30).

The surface plots in Fig. 5.31 show that the same pattern of MI with changing ϕ_{intra}, as seen in Sec. 5.2.2 and Fig. 5.8g and h, remains but decreases exponentially in amplitude with increasing percent spike error. The contour maps at the bottom of the figure show the percent MI difference between non-sorted signals with 0 error and

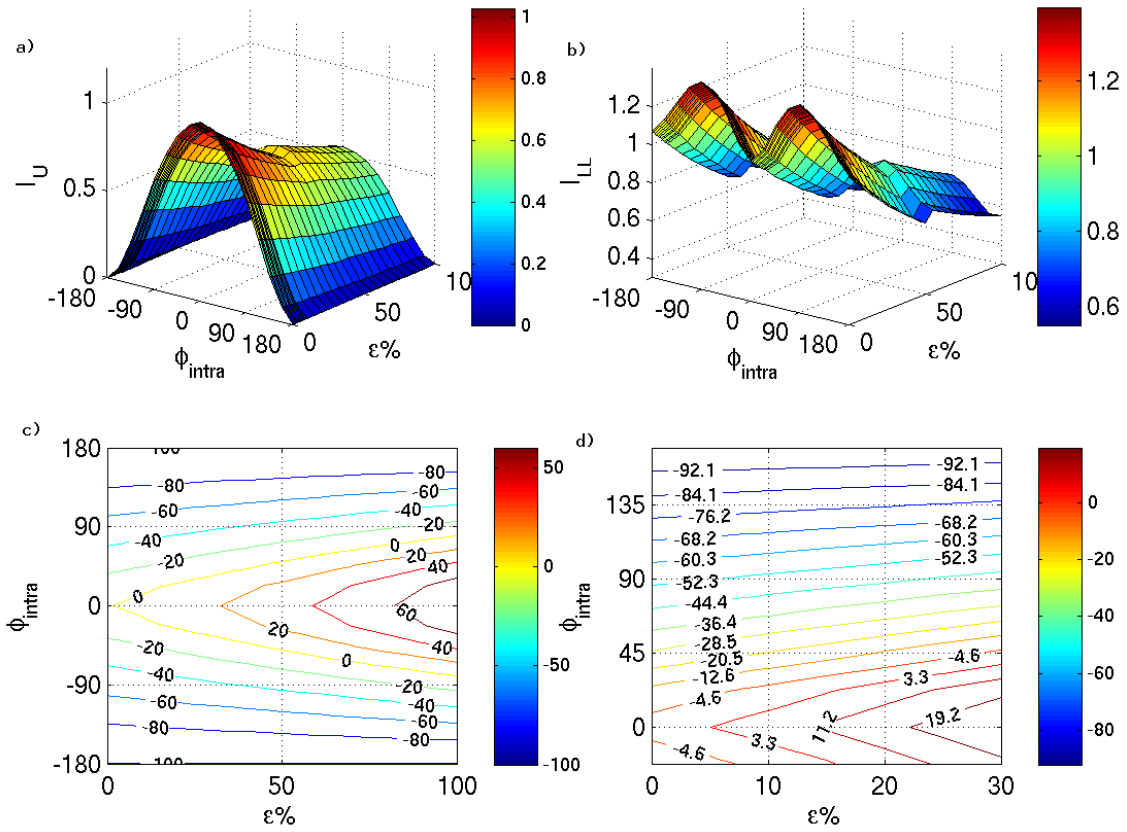


Figure 5.31: Mutual information as a function of intra-electrode preferred direction differences (ϕ_{intra}) and percent spike error ($\varepsilon_{\%}$) at a fixed ϕ_{inter} of 0^o for the non-sorted [top left] and sorted population signals [top right]. Contour plots of the MI difference between sorted and non-sorted signals is shown on the bottom. The iso-contour lines for $\Delta_{U-LL}\%$ is shown for percent errors ranging from 0 to 100% [bottom left] and from 0 to 30% [bottom right].

sorted signals with varying amounts of sorting error ($\varepsilon_{\%}$) at varying ϕ_{intra}. Without sorting, the pooled population suffers a 5% loss of information at an intra-electrode angular tuning difference of $\approx 10^o$. This 5% MI loss can be maintained for larger ϕ_{intra} only if more spike sorting error is present in the sorted signals. For example, when sorting error is at 10%, MI loss stays within 5% for angular tuning differences under 30^o; at the upper range of typical sorting errors of 30%, the units within an electrode can have preferred directions as much as 48^o apart for the 5% information loss.

As described in Sec. 3.3.2, spike error is not completely absent from non-sorted signals, owing to imperfect detection of action potentials. While the amount of detection error in any extracellular recording is dependent upon many variables of the recording system as well as the neural activity which is recorded (Sec. 3.3.2), we can compare MI for specific values of detection error within a range deemed reasonable according to values found in literature (Sec. 5.1.6). Fig. 5.32 shows contour lines for percent differences in MI between non-sorted signals with 5 and 10% detection error and sorted signals with sorting error given by $\varepsilon_{\%}$. As described in Sec. 5.1.6, the amount of detection error expected to be found in a BMI system is highly variable. Based on other work, we believe 10% to be on the high end of typical detection errors.

Fig. 5.31 shows the comparison between sorted and non-sorted MI with 0 detection error. Fig. 5.32 shows the comparison for the high end of detection error ($\varepsilon_{det} = 10\%$, left column) and an intermediate level of detection error ($\varepsilon_{det} = 5\%$, right column). If the detector made consistent errors, for instance due to common electrical noise in the system, and the spike sorter was unable to classify those artifacts as noise spikes, then we might expect the amount of detection error to be some proportion of the total spike sorting error. This is how detection error was defined in Fig. 5.32a and c. However, we believe that detection error will more likely be independent of total spike sorting error; hence, the comparison was made for detection error defined as the ratio of the total number of misdetections to the total number of spikes in the pooled signal, as defined in Eqn. 3.1. Those comparisons are shown in Fig. 5.32b and d. If spike sorting introduced 10% classification error, while 10% detection error remained in the absence of spike sorting, neglecting to spike sort would cause at least 3% information loss even if the preferred directions of the pooled neurons were exactly equal (Fig. 5.32b). This is possible if the spike sorter is able to identify misdetected spikes as "noise" spikes, as mentioned in Sec. 3.1. On the other hand, if total spike

sorting error was 10% while detection error was only 5%, almost a 15^o difference in preferred directions could be tolerated. These contour plots reveal the restrictions on the amount of detection and sorting error as well as angular tuning differences for the cases in which spike sorting can be neglected. If detection error of 10% is present, no more than 35^o of intra-electrode preferred direction differences could be tolerated before pooling incurred a 5% information loss over the labelled-line population, even at the high end of the range of sorting errors (30%).

These results can also be viewed as the percent difference in MI of the non-sorted population with respect to the sorted population at specific values of ε and ϕ_{intra}. The values for the case of maximal MI loss, Δ_{U-J} ($\phi_{inter} = 0^o$) and 10% detection error are summarized in Table 5.5. For example, when non-sorted signals contained 10% detection error and intra-electrode preferred direction differences were set to 45^o, the MI loss due to pooling was -27%, -18%, and -10% with respect to MI in sorted signals which contained a total of 10, 20, and 30% spike sorting error, respectively. A study by Ben-Shaul *et al.* indicated that the average difference in preferred directions between neurons found on the same electrode was 45^o and the overall average difference across neurons in the implant site was 76^o[73]. We, therefore, looked at the iso-contour lines as a function of total spike error in the sorted signals (ε) and intra-electrode angular tuning differences (ϕ_{intra}) at an inter-electrode angular tuning difference of 76^o 5.33.

5.6 Population vector decoding

One question raised earlier (Sec. 1.2, 2.3) was whether a BCI takes advantage of the higher information spike sorted signals provide over non-sorted signals. Because it is still debatable how best to decode motor output from neural activity, it is difficult to determine what aspects of neural activity are affecting the performance of a BCI. It is

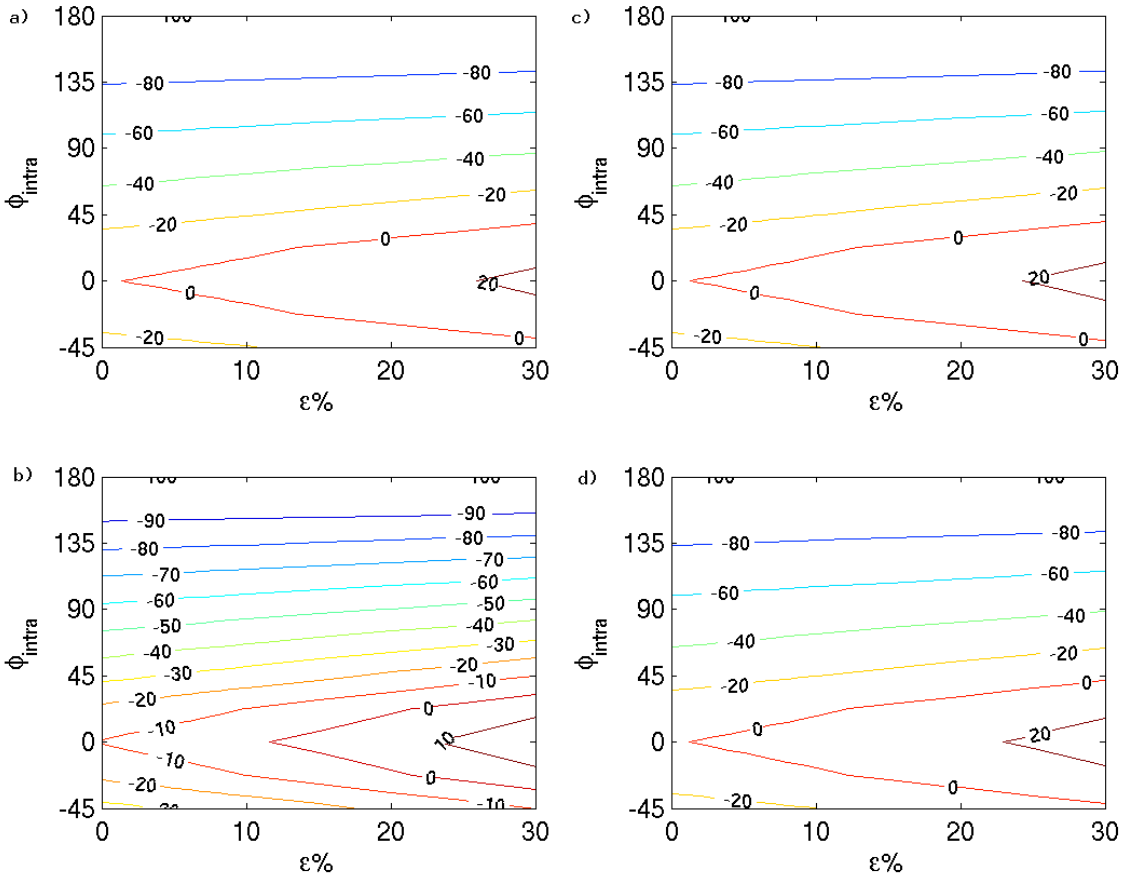


Figure 5.32: Contour plots showing percent MI differences between non-sorted signals with detection error and sorted signals as a function of intra-electrode preferred direction differences (ϕ_{intra}) and total spike sorting error ($\varepsilon_{\%}$). Inter-electrode angular tuning difference was fixed at $\phi_{inter} = 0^o$. Panels **a)** and **b)** are for 10% detection error. Panels **c)** and **d)** are for 5% detection error. **a, c)** Detection error defined relative to amount of sorting error ($\varepsilon_{detect,rel} = (n_{err}/\Sigma\rho_{UX})/\varepsilon_{class}$, the ratio of the proportion of spikes in the pooled signal with error to the proportion of spikes in the constituent spike trains with error) **b, d)** Detection error defined as absolute amount ($\varepsilon_{detect,fix} = n_{err}/\Sigma\rho_{UX}$, the number of spike errors, as a proportion of the total number of spikes in the pooled signal).

		ϕ_{intra}		
	$\varepsilon_{\%}$	0^o	45^o	90^o
$\phi_{inter} = 0^o$	10%	-3%	-27%	-54%
	20%	5%	-18%	-50%
	30%	13%	-10%	-47%
$\phi_{inter} = 76^o$	10%	-1%	-13%	-45%
	20%	7%	-4%	-40%
	30%	16%	2%	-34%

Table 5.5: MI loss due to pooling at 10% detection error for specific combinations of values of total spike error in sorted signals ($\varepsilon_{\%}$) and intra-electrode angular tuning differences (ϕ_{intra}). ϕ_{inter} was fixed at 0^o (top section) and 76^o (bottom section). Each element is the percent difference in MI (Δ_{U-LL}) of non-sorted signals with 10% detection error with respect to sorted signals with $\varepsilon_{\%}$ total error; therefore, negative values indicate MI loss due to pooling; positive values indicate that MI is not lost but actually gained with respect to the spike sorted signals.

possible, however, to compare the information in the outputs of a decoder when sorted and non-sorted signals are used for input. Here, we make such comparisons using a population vector decoder, which is one method used in BCI studies [70, 80, 11].

The MI loss due to pooling was compared before and after decoding via computation of the population vector. We determined the MI loss between $\mathbf{P}_{LL}$ and $\mathbf{P}_U$ (see Fig. 5.4) and compared this loss with Δ_{U-LL}, the MI loss between ρ^b_{LL} and ρ^b_U. The results in Fig. 5.34a, c, and e) reveal that when the preferred direction of each neuron is known exactly, the population vector is able to recover all the information at its input. Fig. 5.34b, d, and f) show there is no significant difference in the MI loss due to pooling after the population vector transformation. Thus, for cosine-tuned neurons, a population vector retains the full information advantage that spike sorting provides.

However, when the movement direction is decoded from the population vector, information is lost. MI in $\mathbf{P_{LL}}$ and $\mathbf{P_U}$, as indicated in Fig. 5.4, was computed and compared to that in $\hat{\theta}_{LL}$ and $\hat{\theta}_U$. The comparison between MI in the population vector and MI in the decoder output is shown in Fig. 5.35. When $\phi_{inter} = 0^o$, there was

significantly more MI loss due to pooling after decoding $\hat{\theta}$ from the population vector. As ϕ_{inter} increased, the MI loss before and after estimating $\hat{\theta}$ from the population vector became more similar. Thus, for cosine-tuned neurons, a population vector decoder at least retained, if not increased, the full information advantage that spike sorting provided.

5.7 Summary

Mutual information was directly computed from the simulated spike activity of a population of 4 cosine-tuned neurons as the relationship between tuning of the neurons was varied. Comparisons were made between the information in sorted activity and union activity.

Simulations demonstrated that pooling activity from pairs of neurons carrying motor information in cosine-tuned responses results in 56% information loss when the directional tuning of the neurons differs by 90^o and no information loss when the tuning of the neurons is equal. An additional pair of neurons will add information to the sorted population; this information is fully retrievable in the corresponding pooled population only when the units that are pooled have equal preferred directions. The amount of information lost by pooling directly depended on the difference between the preferred directions of pooled units. The information loss due to pooling was 0 when the preferred directions within pairs were equal; it was less than 5% when the preferred directions differed by no more than 23^o in the case of maximized information in the sorted population; but it exceeded 15% when the preferred directions differed by 45^o or more. All the information was retained in the population vector, and spike sorting was found to be more important when the population vector decoder was used on cosine-tuned neurons, as indicated by greater information loss due to pooling in the decoder output.

Information in single-unit and population signals decayed in an exponential fashion with spike error, as might be added by either detection or classification of spikes. MI loss in the pooled population with respect to MI in the sorted population decreased linearly with percent spike error. Information in a system of up to 4 non-sorted cosine-tuned neurons is expected to have less information than corresponding spike sorted cosine-tuned signals even when they include 10% spike error for the case in which neurons sorted from the same electrode have differences of 45^o in preferred directions. If the preferred directions differ by 90^o, the information in non-sorted signals will be 50% less than that in sorted signals with 10% error and only surpasses information in sorted signals once spike error exceeds 100%; i.e., in the unlikely scenario in which there is a false positive or false negative error for every spike actually emitted by the neuron. In the presence of 10% detection error, the only possibility for a 10% MI loss or less to be maintained without spike sorting is in the case of small intra-electrode differences in preferred directions ($0 \leq \phi_{intra} \leq 45^o$) and/or large amounts of total error in the sorted population ($\varepsilon \geq 30\%$).

We observed the effect of the absolute mean firing rate on each unit's information, as well as differences in mean firing rates and modulation depths on the information loss due to pooling, but the most significant effects were due to intra-electrode differences in preferred directions and percent spike error. Contour plots showed iso-curves of information loss due to pooling for combinations of differences in preferred directions and percent sorting error, and indicated the range of conditions under which information losses without spike sorting might be tolerated. We found that for typical differences in preferred directions between cosine-tuned units detected on the same electrode, according to [73], and detection errors that might be typical based on [81, 61] (namely, for 45^o of intra-electrode angular tuning differences and 10% detection error), neglecting to spike sort would cause a 10% information loss even if

spike sorting were to introduce 30% error (Fig. 5.32, bottom left). If detection error was small relative to total spike sorting error, then leaving out spike sorting would only be tolerable if spike sorting performed poorly with about 30% total error.

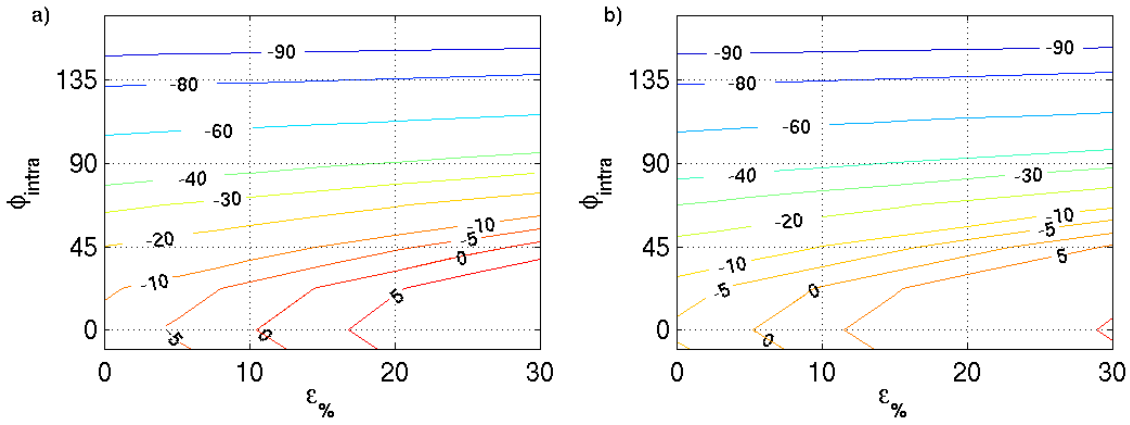


Figure 5.33: Contour plots showing percent MI differences between non-sorted signals with detection error and sorted signals as a function of intra-electrode preferred direction differences (ϕ_{intra}) and total spike sorting error ($\varepsilon_{\%}$). Inter-electrode angular tuning difference was fixed at $\phi_{inter} = 76^o$. Panel **a)** is for a fixed detection error rate of $\varepsilon_{det,fix} = 10\%$, as defined in Fig. 5.32. Panel **b)** is for a fixed detection error of $\varepsilon_{det,fix} = 5\%$.

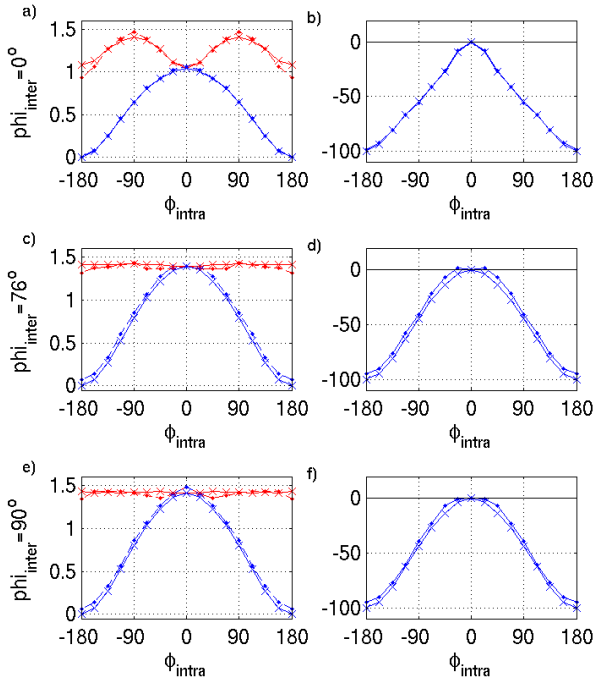


Figure 5.34: A comparison of mutual information in ρ^b_{LL} and ρ^b_U with that in the population vector derived from these signals. The left column shows $I(\theta;\rho^b_{LL})$ (red 'x'), $I_{PV}(\theta;\mathbf{P}_{LL})$ (red '$\cdot$'), $I(\theta;\rho^b_U)$ (blue 'x'), and $I(\theta;\mathbf{P}_U)$ (blue '$\cdot$'). The right column shows the percent difference in MI between signals in the processing path that includes and excludes spike sorting before ('x') and after ('$\cdot$') population vector decoding; i.e., Δ_{U-LL} and $\Delta_{PVU-PVLL}$.

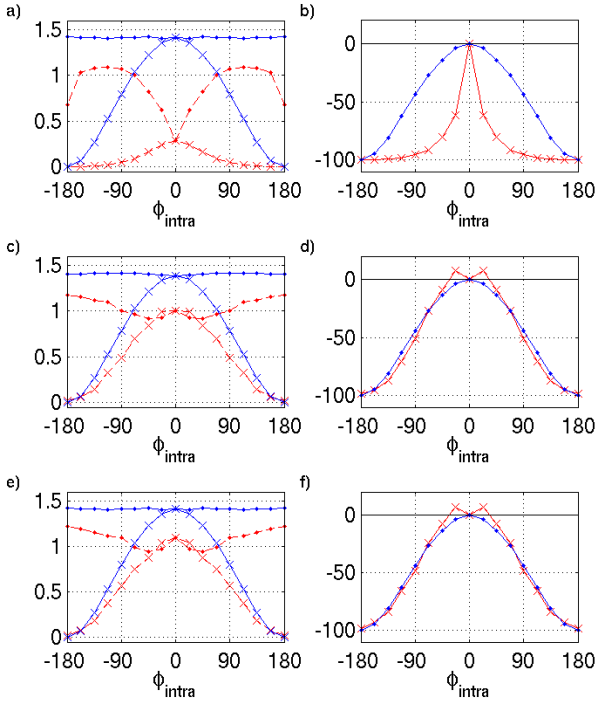


Figure 5.35: A comparison of mutual information at the input and the output of the population vector decoder. The estimates of $\hat{\theta}$ were derived from ρ_{LL} and ρ_U. The left column shows $I(\theta;\rho_{LL})$ (red 'x'), $I_{PV}(\theta;\hat{\theta}_{LL})$ (red '$\cdot$'), $I(\theta;\rho_U)$ (blue 'x'), and $I_{PV}(\theta;\hat{\theta}_U)$ (blue '$\cdot$'. The right column shows the percent difference in MI between signals in the processing path that includes and excludes spike sorting before ('x') and after ($\cdot$) population vector decoding; i.e., Δ_{U-LL} and $\Delta_{\hat{\theta}U-\hat{\theta}LL}$.

Chapter 6

Discussion

In multi-unit recording applications, the goal is to retain as much of the relevant information as possible in the original spike trains. If the spike output from the neurons could be reproduced exactly at the output of the acquisition system, the information transmitted to the computational decoders remains unchanged. Thus, for the purposes of decoding multi-neuron spike activity for a neuromotor prosthesis, it is desirable to extract the spike activity of many individual motor-related neurons to maximize the information available to the decoder. However, spike activity from more than one neuron is detected on an electrode implanted in the brain. Despite the lack of understanding of the effects of spike sorting, multi-unit signals have been discriminated into single-unit spike trains which are used in computational analysis to decode the neural response. The question remains whether it is necessary in a BCI to keep the identity of the spike trains separate and how much information is lost when neurons' responses are used in the combined form. Because of the burden spike sorting places on high-density recordings in BCIs, it was worth questioning the necessity of this task.

If spike sorting could be done easily and accurately, it would appear to be an obvious and logical step to take. However, spike sorting hinders the progress of miniaturizing neural prosthetic electronics into fully implantable hardware. The high fidelity signals needed for spike discrimination require analog filtering and high data transmission bandwidth, which are costly tasks in VLSI (very large-scale integrated) hardware. Despite all the research devoted to the problem of spike discrimination, an unsupervised real-time spike sorter that performs with consistent accuracy is yet

to be developed, even without the constraint of implementing spike sorting in an implantable device [64, 18, 19, 60, 29, 30].

Since the dawn of multi-unit recordings, spike sorting was deemed necessary because the original goal of such recordings was to observe the individual waveforms of single neurons and the interactions between them [18, 17]. There may be useful information in such interactions or in uniquely timed patterns of single-unit spike trains, but it has not been proven whether that information does exist in the population activity recorded by a BCI or if it is exploited by a BCI decoder. Yet, the necessity of spike sorting has been taken for granted without studying its effects. As BCI development advances and reaches the stage at which every opportunity for improving efficiency needs to be taken, it was important to question this assumption analytically. We began the process of analyzing spike sorting and its role in a BCI using information-theoretic analysis.

6.1 Summary of results

This dissertation investigated the role of spike discrimination in multi-neuron recordings that are used for computational decoding in a BCI. First, we explored the assumptions upon which the efforts to implement spike sorting have been based. The problem was framed from the perspective of information theory. We proposed to investigate two main assumptions: (a) that signficant loss of mutual information is caused by pooling single-unit signals, and (b) the information loss due to spike sorting error is not greater than this information loss due to pooling. We then described mutual information, the basic tool we used to tackle the problem. We proceeded to perform information-theoretic analysis on simulated neural data to: (1) determine the effect of error resulting from spike processing, (2) to analyze the information available in a non-sorted cluster of neurons and compare it to that available in the

corresponding sorted neurons, and (3) to examine information about movement direction that non-sorted spike activity of units could carry if they were cosine-tuned. It was shown that (1) typical amounts of spike sorting error (from misdetection and misclassification) can be detrimental to each unit's information content, (2) information in a pooled cluster of unit spike activity was not necessarily less than that in corresponding sorted units, but the information loss depended on the disparity between unit information, and (3) mutual information measured by direct computation in cosine-tuned units was not affected significantly by pooling for similarly-tuned units; but it was significantly decreased when the preferred directions differed by 45^o or more, even when typical amounts of spike error was present.

In order to address the question of how sorting error affects the ability to decode information from neural spike trains, Shannon information theory was applied to spike trains simulated with sorting error. The estimated mutual information rate was found to decrease exponentially with classification error, expressed as a proportion of the total number of spikes in the original spike train. Less than 13% classification error could be tolerated before the information content dropped to half its maximum value with no error. Detection error, as well as classification error, plays a major role in the comparison of mutual information in sorted and non-sorted populations. Random detection error was also simulated and found to have a have a similar but more detrimental effect than classification error from a second neuron; only 10% detection error would cause 50% information loss. Yet, because of the small differences in effect, random spike error was used in Chap. 5 when simulating both sorted populations with total spike sorting error and non-sorted signals with detection error only. The process of spike sorting was shown to be a potential source of significant information degradation through the introduction of spike error. However, detection error, simulated as random spike error, was found to be an equally

detrimental source of information degradation.

In order to analyze the information content available in linearly decoded multi-unit signals, cases of two and three superimposed neural responses to a stimulus were simulated. The multi-unit Shannon information rates estimated from the output of the optimal linear decoder were compared with those of the constituent single-unit spike trains and also with those of corresponding sorted signals. The mutual information in the multi-unit response was found to depend on the degree of similarity in the information of the single-unit spike trains. Summed responses could be used at low informational cost in redundantly encoding clusters of neurons. Higher information losses were incurred as the encoding became more independent.

Similar results were demonstrated for mutual information in cosine-tuned neural spike activity, independent of the type of decoder that would be used to extract information from the responses. Cosine-tuned neurons, as found in the motor cortex, were modelled with varying differences in their preferred directions. Mutual information was directly computed in simulated spike activity from a population of 4 cosine-tuned neurons as the tuning amongst the neurons was varied. Comparisons between the information in sorted activity and union activity showed that the information loss due to pooling 2 pairs of units was under 1% when the preferred directions within pairs were equal and less than 10% when the preferred directions differed by no more than 22^o. When spike error in the sorted population exceeded that in the non-sorted population, larger intra-electrode angular tuning differences were tolerable for a given level of a MI loss. MI loss due to pooling was lessened as spike error was added to the labelled-line signals, increasing linearly with percent spike sorting error.

6.2 Implications for a neuromotor prosthesis

We analyzed the benefit of spike sorting by answering basic questions about the sensitivity of information to spike error, as would be introduced by spike sorting, and the potential of pooled multi-unit signals to convey information. We compared information in sorted and non-sorted populations within a particular realization of a BCI, in which the primary source of information is in the cosine-tuning properties of neural activity and the signals simulated a center-out reaching task. In this scenario, we were able to make specific applications to BCI technology.

BCIs (both intracortical and extracortical) that have been tested in humans allow the user to make binary decisions (e.g., up/down) or decisions with up to 8 choices [13, 84, 85]. The maximum information obtainable about direction of movement in our simulated center-out task was 3 bits. The available information about movement direction would be higher for a finer sampling of the directional space. To obtain all the information available would require additional neurons at varied angular differences to best extract that information. Increasing the modulation depth of the units would also cause the information to increase. We do not know in general how much information is required for a BCI to operate "effectively". However, we can imagine a hypothetical situation. Suppose a BCI controller executes the user's commands perfectly using spike-sorted signals with no error. Assuming the BCI uses all the information available in the neural signals it acquires, an information loss of 10% means that same controller would execute the user's commands correctly 9 out of 10 times using non-sorted signals. Depending on the task the BCI is meant to execute, an information loss of 10% may be acceptable, especially if this also translates to a more portable, less costly device with smaller execution latencies and longer battery-life.

We have seen that spike sorting is generally necessary to preserve the desired information in neural spike activity. However, we have also found that small sacrifices

in information are theoretically possible under particular conditions and may be accepted for the sake of other advances in BCI technology. From the information-theoretic analysis performed, we have a better idea of the scenarios in which the benefits of spike sorting ultimately do not outweigh its costs. We describe these exceptional scenarios in sections 6.2.1 through 6.2.4.

6.2.1 Exception 1: Classification error exceeding detection error

Spike sorters make errors, but the effects of this error on the ability to decode the neural responses had not been studied. As a step towards determining whether the benefits of spike sorting in high-density neural recording applications, such as neural prostheses, outweigh the costs, information-theoretic analysis was conducted on stimulus estimates reconstructed from spike trains with randomly added and deleted spikes. Without knowing what of the external stimuli the brain considers to be relevant and important, or how the brain encodes that information, it is difficult to ascertain whether spike sorting is worth doing in spite of sorting error. However, Shannon information-theoretic measures applied to neural activity allowed us to quantify how increasing sorting error changes the ability to reconstruct the stimulus that generated the activity.

According to the results presented in Sec. 3.3.2, individual spike trains must have less than 2% classification error to maintain 80% of its original information content; and less than 14% error to maintain 50% of its information content. This tells us the degree to which spike sorting would allow or hinder information in single-unit responses to be retained. The results also revealed that false negatives and false positives had the same effect, and that the effect was not directly related to average firing rate. Rather, information rate falls off exponentially and approaches

its asymptotic value at approximately 17% error, where the coding fraction also indicates the spike train is no longer correlated to the stimulus (Sec. 3.3). Since the omission and insertion of an erroneous spike are equally detrimental to information content, a spike sorter should aim to optimize for both sensitivity and selectivity.

As mentioned, detection error can be present in non-sorted spike trains as well as sorted spike trains, and non-sorted spike trains may have higher occurrences of detection error than corresponding sorted trains. Sec. 3.3.2 revealed a similar degradation of MI due to random detection error. Spike error in populations of cosine-tuned neurons similarly caused exponential decay of mutual information.

The amount of detection error in the sorted and non-sorted spike trains would depend on the quality of the recordings, the behavior of the neurons that are recorded (firing rates, burstiness, similarity in waveforms), as well as the specific methods of detection and sorting that are used. Known studies that have either reported spike sorting error have reported only classification error or do not distinguish between the two types [20, 19, 18]. It is very difficult to obtain *in vivo* error rates because in order to determine the true spike trains, intracellular and extracellular recordings would need to be made simultaneously. One way that detection and classification error rates could be estimated is through the use of simulated spike activity that can be directly input to a spike sorter[1].

We assumed in Sec. 5.3 that the detection error is a small proportion of the total spike error in extracted spike trains from extracellular recordings. However, results from [86, 82] indicate that detection error could vary widely and be high enough to offset any savings in information that might have been gained through elimination of classification error. For classification error rates much higher than 30%, non-sorted signals without error could feasibly exceed sorted signals in information content.

[1]Nate Smith has developed a neural simulator. Use of this simulator for measurements of detection and classification error is being discussed via personal correspondence with C.S. Henriquez.

If a spike sorter can not meet the desired error tolerance, alternative schemes for computational analysis on neural population responses should be considered; e.g., using multi-unit signals of 2-3 neurons, which can be more accurately detected from microelectrodes. The method for acquiring multi-unit signals on a single channel is more robust than acquiring single-unit responses because the former requires only spike detection and not discrimination, which introduces the more common type of error. If the difference in total sorting error and detection error without sorting is less than 20%, however, the tolerance for intra-electrode tuning differences becomes much more stringent, as seen in Fig. 5.32. These results, in combination with the following considerations, make it unlikely that spike sorting could be neglected and result in acceptable information losses.

6.2.2 Exception 2: Similar tuning within pooled units

A second condition required to tolerate pooling is that the difference between the information conveyed by the pooled units is limited. For cosine-tuned units, this means that the differences in preferred directions should be within a given tolerance. From the simulation results in Figs. 4.6 and 4.9B, we saw that linearly decoded information could be maintained when neurons were pooled, but large information losses could also be incurred. The results were directly dependent on the similarity in MI between neurons. It is possible that local neurons encode information similarly in a way that allows them to be summed, but there is no definitive proof of what feature of the cortical response is necessary to extract for decoding. However, there are examples of neural encoding based on a body-centered map or spatial maps of stimulus features. Auditory neurons are spatially arranged tonotopically; retinal neurons map out the visual field; neurons in the visual cortex are spatially arranged according to stimulus orientation; columns in the rat barrel cortex map out the

vibrissa locations [87, 88, 89, 90]. These examples indicate some spatial coherence in cortical structures, suggesting neurons that are physically close may be contextually close as well.

Panzeri *et al.* [91] showed that negligible information about discriminability of whisker stimulation was lost by pooling spike activity from pairs of neurons from the same column of the rat barrel cortex [91]. The simulations of linearly decoded information from integrate-and-fire neurons (Chap. 4) corroborated results from [92]: pooled electrodes of two and three neurons encoding Gaussian stimuli had much less information than corresponding electrodes with sorted spike trains, but this gap diminished as the disparity in information between pooled neurons diminished. These results provide possible explanations for the linear prediction results from [92]. Since no significant perfomance loss was found due to recombining units, we would classify the *in vivo* linear prediction results into the category 3 in Sec. 4.3, of combining units with similar information. The comparisons with the selected units with highest variance resemble case C, in which combining very informative units with very non-informative units causes the most information loss due to pooling. Thus, it seems possible that similar tuning exists among neurons recorded on the same electrode, but *in vivo* results currently available are inconclusive.

Nonetheless, these results, together with the results from cases B and D, imply that 2- or 3-neuron multi-unit signals *may* provide all the precision needed for computational decoding. That led us to focus on the properties that would most influence the information loss due to pooling in the context of motor neurons. We asked what would be the precise nature of similarity in information for neural activity recorded in a BCI. Since cosine-tuning is commonly observed in motor neurons and exploited in BCI applications, we looked at differences in such tuning properties and their influence on MI loss in pooled versus labelled-line neurons. We extended

this to look at more than 1 electrode. We extended investigation into the problem by looking at sorted signals with sorting error and non-sorted neurons. We first examined the sensitivity of MI loss due to pooling and found that the loss decreased linearly with intra-electrode angular tuning differences if the electrodes carried the same information, and decreased slightly more slowly if the information between electrodes was orthogonal (Fig. 5.10). In the case in which the population information was maximized ($\phi_{inter} = 90^o$, Fig. 5.10f) and no spike error was present, the pooled population could tolerate less than 12^o of intra-electrode angular tuning difference before resulting in 5% MI loss (Fig. 5.12).

According to estimates made by Ben-Shaul et al. [73], differences in preferred directions would likely be much larger *in vivo*. Ben-Shaul et al. measured preferred directions from arrays of electrodes in the motor cortex. The preferred directions were measured for neurons on the same electrode and for neurons across regions of the motor cortex from multiple electrodes. They reported average differences in preferred directions during the preparation phase of center-out movements to be 76.0 ± 3.0^o for all pairs of neurons ($n = 270$), and 44.3 ± 9.5^o for pairs of neurons on the same electrode. At these preferred direction differences, eliminating spike sorting would result in 20-30% information loss for reasonable ranges of detection and sorting error (Fig. 5.32). Even if spike sorting error exceeded detection error by 20%, the non-sorted population would have 10% less information than the sorted counterpart.

Reich et al. [66] studied the information encoded by single-neuron responses in the primary visual cortex (VI) and investigated how much of this information is retained in the population response [66]. They show that the amount of information retained depends on how the population responses are considered. They showed that summing the activity of two neurons is not expected to cause significant information loss. However, for cluster sizes of three or more, they show roughly 20% information

is lost if the neurons' identities are lost. We considered cluster sizes of two and three only because the number of neurons discriminated on the same electrode is most often two or three [15, 66]. However, a caveat should be noted in attempting to apply their results to the questions we raise about the necessity of spike sorting for preserving sufficient information in a BCI. While Reich *et al.* showed that *independent* information cannot always be retained when neural activity is pooled, the population activity modelled in chapters 4 or 5 never reached as high a level of information as the sum of the individual rates as theirs. It is important to realize that the information in the neuron cluster would reach this maximum only if the conditional probability distributions were independent, and that the neuron cluster could encode more than any one of the neurons only if there was some non-mutual dependence between conditional probability distributions; i.e., the neurons were not exclusively mutually dependent [57]. In other words, the summed population activity only loses information that a discriminated population response would retain if the neurons are either encoding independent attributes of the same stimulus or are encoding independent stimuli. In the simulations of Chap. 4, we looked only at what the population response encodes about one stimulus attribute of interest (represented by the amplitude of $\boldsymbol{\theta}$). This would be a situation more relevant to a BCI, in which we are trying to decode one stimulus attribute, not multiple independent stimulus attributes of the same stimulus. On the other hand, we did not model in the cosine-tuned case the inclusion of neurons that do not encode movement direction at all. We were concerned only with the stimulus-modulated components, as our interest is in the information relevant to a computational decoder. Nevertheless, if unrelated neurons could not be avoided, using non-sorted signals in a BCI would not prove a viable option except in restricted cases as described in sec. 6.2.4.

Reich et al.'s results could indicate, in line with case A of our simulation study

of Chap. 4, that when the neurons are encoding independent information, some information is lost by summing the responses. Some studies indicate that the neurons detected on the same electrode respond to similar stimulus attributes [66, 73] and have correlated tuning curves [93]. However, [36] argues for the random distribution of preferred directions across regions of the motor cortex although their plots seem to show clusters of similar preferred directions and more scattered distributions on a larger scale. Furthermore, as described above, even though preferred directions are more similar for neighboring neurons than randomly-chosen pairs of neurons [73], the differences expected to be found within an electrode are large enough to cause unacceptable information losses in a BCI without spike sorting.

The main result of these analyses is to provide bounds on the conditions under which spike sorting could be excluded from BCIs. Spike sorting is less likely to be a crucial component of BCIs for somatotopic mapping codes and recording hardware that provides low rates of detection error. A somatotopic structure of the motor cortex would entail a representation of the body that is mapped onto the cortex, such that adjacent cortical neurons control adjacent muscles in the body. The contour maps generated for intra-electrode angular tuning differences and sorting error indicate that the changes in tuning across the cortex should not exceed 45^o per $100\mu m$. However, this kind of mapping would not be a sufficient criterion for determining the ability to pool neurons with tolerable information loss. The decoder would also need to extract the appropriate information. For example, if a neuron's firing rate contributed directly to the magnitude of a particular muscle activation, the BCI decoder would need to decode activation levels of corresponding muscles, and then apply the appropriate transformations to determine the desired target signal, such as limb position. Even with a somatotopic structure in the motor cortex, the preferred directions are not guaranteed to follow such a pattern unless preferred directions are

closely linked to muscle activiations. We discuss issues of encoding further in the following section.

6.2.3 Exception 3: Encoding based on magnitude of neural firing

We look here at an issue that plays a role in the argument that separating spikes is crucial to preserving information. We might expect great information loss due to pooling if the neurons encoded the target signal through precise temporal patterns that differ from each other, such that the uniqueness of these patterns is forfeited when combined. Combining even two or three spike trains might appear to be an unsatisfactory approach to analyzing multi-neuron data because the precise timing of each neuron's spikes would be lost. The results from Chaps. 4 and 5 did not indicate that combining neurons necessarily created large amounts of information loss. We suspect this was possible because the code was not based on precise temporal patterns.

If timing precision was important, that importance might be reflected in a direct relationship between the correlation of two spike trains and their information content. However, we did not observe such a relationship in Chap. 4. Instead, we observed that spike trains from the same stimulus and encoding mechanism were only approximately 20% correlated; yet, when combined as a multi-unit signal, the information rate was unchanged. In addition, $\dot{I}(s_1; \rho_{UB})$ was greater than $\dot{I}(s_1; \rho_{UA})$ although the average correlation between the multi-unit signal and ρ_1 in cases A and B (sec. 4.1) were both 0.70. Furthermore, we observed that at low correlation, mutual information between two spike trains can be low in some cases and high in others. Information can remain maximal at 20% correlation or drop to baseline levels at 70% correlation (Fig. 4.5). Hence, correlation to 1-ms resolution does not appear to encode most of the information in this model.

Instead, our simulations illustrate that neural coding could employ redundancy to make the population response more robust, as has been suggested in literature [94, 95]. Because of the stochasticity of spike trains, a given stimulus will not yield a unique sequence of spikes. Multiple responses in combination, however, may reduce the entropy in the system. The gain in information in case D of Chap. 4 supports redundancy as a mechanism to ameliorate information degradation due to random error as also seen in sections 5.3. This property encouraged us to explore multi-unit signals as a potential alternative to spike-sorted signals as the input into a decoding algorithm. In case B of the linearly-decoded information simulations, no information was lost from any of the individual spike trains although they were uncorrelated and unit 1's spikes could not be distinguished from unit 2's or unit 3's in the multi-unit signal. Thus, there are encoding scenarios in which knowing which neuron fired at precisely what time is not as important as knowing less precisely when clusters of neurons are firing. On the other hand, even with the same encoding mechanism, it is possible that non-redundant spike trains would be pooled and have detrimental effects on MI.

Information analysis on sorted and non-sorted signals in a different encoding system, such as a code of precise temporal patterns, would likely yield different results than seen here [96]. However, we do not know of evidence for such encoding in the motor cortex. We chose cosine-tuning as a property to model because of its relevance to a neuroprosthetic application. An appealing model of motor encoding entails a somatotopic mapping of motor cortical regions to muscles and direct control of muscle activation by the magnitude of firing of corresponding neurons in the motor cortex [97]. This model predicts directional cosine-tuning in neurons [97]. The results from preliminary analysis performed on borrowed *in vivo* data were consistent with this model [92]. These results showed that linear predictor performance was not signfi-

cantly worsened by undoing the spike sorting procedure for predictions of position and velocity and was even improved in the case of force predictions. Under such an encoding scheme, the results of this simulation study argue for the importance of improving spike sorting methods to reduce error, or implementing BCIs without any spike sorting and focusing on improving algorithms already being used to decode motor output from non-sorted multi-unit spike trains.

If a precise timing code was used, multi-unit signals would be expected to have less information than perfectly sorted signals unless the single-unit signals were completely redundant. Spiking patterns in some systems have been found to encode information at temporal precision on the order of ms (i.e., binary words on the order of 10s of ms, for which each bit represents 1-ms bins). However, this has not been shown in the motor cortex. Furthermore, binning to 100ms essentially reduces the encoding scheme to an average firing rate code. Even if the code is temporal, the only information currently captured in current BCI systems is in binned spike counts. Essentially, information in precise timing would be lost in the processing typically done on neural signals for neuroprosthetic applications.

Despite the efforts to obtain individual spike trains and the precise timing of the action potentials through spike sorting, it is uncertain whether information in individual spike times, even if it exists, is being exploited. Many of the algorithms that are currently used in BCI applications derive predictions of limb movement parameters (e.g., position, velocity, force) from neural spike activity that has been binned in time windows ranging from 20ms-100ms (most commonly 50 or 100ms). These binned counts are then employed in a linear decoding method [10, 71, 12, 70, 11]. A large proportion of the major work done in BCI development has used an optimal linear filter [10, 71, 12, 80].

It has been assumed without rigorous testing that the responses of individual

neurons must be kept separate. One potential weakness in this assumption is that no biological mechanism has been provided for a post-synaptic neuron's ability to distinguish between presynaptic inputs. Rather, neurons are considered to integrate postsynaptic potentials indiscriminately [98]. Meanwhile, biophysically complex, well-trusted models, such as the Hodgkin-Huxley neuron, do not distinguish between presynaptic currents [93]. While cortical neurons may summate the activities of presynaptic neurons, the contribution of individual cortical neurons to the overall population code is not known.

Secondly, current decoding algorithms that do not reproduce the individual responses precisely or with 100% accuracy have been successful. Timing information is lost because discriminated spike trains are binned, often to 100ms, as a pre-processing step. Accuracy is lost because spike sorting introduces spike identification error. In other words, single unit activity is characterized in current algorithms by somewhat inaccurate averages of single neuron spike counts over time. Higher accuracy and temporal precision in reproducing aggregate activity can be achieved by characterizing multi-neuron activity with an average of spike counts over the cluster. In fact, population activity is commonly characterized by instantaneous firing rate or peristimulus time histograms which sum activity across neurons [99, 100, 101].

Another approach used in neuroprosthetic control, aside from intracortical recordings, is to decode the motor output from the summed activity of general regions in the brain using EEG and ECoG recording techniques [85, 84, 8]. These studies have already indicated that useful motor information is available in the non-discriminated activity of many neurons. We focus on cortically-driven neuroprostheses because the precision of information obtained and the speed at which it can be decoded is more limited with EEG and ECoG recordings [102, 9]. Cortical BCIs are considered able to provide speed and movement precision to better mimic natural movements over

extracortical devices [103, 102]. However, these other types of BCIs lend evidence to the possibility of using aggregate neural activity for BCI control, perhaps only possible when encoding is not based on temporally precise patterns of spikes.

Eliminating spike sorting in a BCI would require reconsideration of the design of many components of the system, including decoding algorithm and perhaps adjusting electrode impedance to record from the desired neural signals, whether multiple single unit activity, multi-unit activity, or local field potentials [104]. However, low tolerance for the amount of detection error and preferred direction differences, as well as the unpredictability of the amounts of spike sorting error, make it seem unlikely that future versions of BCIs would do away with spike sorting.

6.2.4 Exception 4: Corrupting activity is masked

We have seen that redundancy in the information carried by spike trains can outweigh the corrupting effects of noise in spike trains. Outcomes 3 and 5 of Sec. 4.3 show that the information in each unit is not necessarily compromised in the multi-unit signal. In the case of non-redundant information, neurons which are not maximally encoding the stimulus corrupt the information whether the signal is single-unit or multi-unit. However, the degree to which it corrupts single-unit or multi-unit information depends on the processing algorithms used to process the signal.

We observed in Fig. 4.5 that adding a 3rd low-information unit did not corrupt I_U more, indicating that somehow the unit carrying high information and its (low-degree) redundancy managed to overcome the confounding effects. When all 3 units were corrupted similarly, adding each unit aided I_U through redundancy. This is similar to the case described in Sec. 5.4.5 and illustrated in Fig. 5.28, in which a unit with a high modulation depth was not as corrupted by a unit with much lower modulation depth even if the two had large differences in preferred direction and the

second unit carried potentially confounding information. The same effect was shown in Fig. 5.29. We found that redundancy was a desirable trait when neurons were pooled; for small differences in preferred directions, lower MI loss was incurred by pooling, and this same level of information loss was maintained for smaller differences in modulation depth, by which redundancy was magnified (for $\phi_{intra} < 90^o$, the contour lines spread out as δ decreased). When the angular differences increased, the MI loss increased, but it helped to have one neuron's activity mask the other, so that the same level of MI loss could be maintained when there were large differences in modulation depth (for $\phi_{intra} > 90^o$, the contour lines spread out as δ increased). Thus, it appears that the disparity in the magnitude of responses is another factor which may mitigate potential information loss caused by the dissimilarity of tuning between neurons that are pooled.

Information-theoretic analysis on *in vivo* data may help clarify what weight the single-unit response carries in the neural encoding process and whether computational decoding could perform better on accurate multi-unit data than on less accurate single-unit data. Relevant issues include whether single unit trains are as crucial to information processing as once believed and whether the population response as a whole masks the details of individual neuronal responses. Comparing the information rates in our model to the rates of *in vivo* multi-unit data might allow us to explain why it is or is not important to maintain the identity of the neurons detected on a channel. New avenues might be explored for reducing the demands on signal processing hardware while maintaining desired levels of information. For example, higher amplitude spikes may provide sufficient information and would reduce detection error. On the other hand, if the contribution each neuron made to the population information is not directly related to its spike amplitude, MI would inevitably confounded by using a simple threshold detector without spike sorting.

The effects the signal characteristics have on pooled and labelled-line MI were explained in section 5.4. We now have enough of an understanding of the effects of various parameters that, if the motor cortical population signals in BCI recordings could be characterized, we could estimate MI losses due to pooling. There can be a lot of variability in the mean rates and modulation depths present in any given set of recorded units. However, there are general characteristics that could be measured which would allow estimates of information loss for a given configuration of implanted electrodes. If there is somatotopic mapping in the motor cortex and cosine-tuning is the primary feature exploited for decoding, then the distances between electrodes will be a fundamental determining factor of the amount of motor movement information extracted.

6.3 Application of mutual information to gain insight on BCIs

Mutual information [37, 35] is a commonly accepted measure for quantifying neural encoding and the effects of signal processing on neural activity. Quantitative mutual information results provided a tool to give us confidence in the hypotheses we made based on intuition. We have noted the relevance and applicability of mutual information measures to the problem of assessing the benefits of spike sorting. We have applied these measures and demonstrated the these measures produce results that can also be intuitively explained.

We can see from the results in Sec. 5.4.4 that SNR was limited as a measure for our cosine-tuned model where MI was not. MI captures the asymptotic limit of a spike train's ability to convey the movement direction signal, while SNR does not. SNR increases with firing rate in a Poisson signal, while the ability to discriminate movement directions from responses can be worsened by increased firing rate. MI,

on the other hand, does capture this notion of discriminability. One problem is that SNR is not readily defined for an inhomogeneous signal. Nor is it readily defined for multi-dimensional signals. Even if all the information we desired to measure was linearly encoded and SNR was an appropriate measure of information, it is not conventionally defined for multi-dimensional signals. We are able to measure SNR for single-unit signals but not population signals. Meanwhile, we have been able to provide a logical explanation for the MI results found in these analyses.

The work in this dissertation builds upon previous work using Shannon information measures to analyze neural spike activity. Other groups have studied mutual information in single neurons [57, 51]; some have even estimated MI in populations of neurons [55, 91]. Some have compared the performance of decoders using spike trains [105, 106]. Their results in conjunction with the results presented in this dissertation provide a good, though far from comprehensive, understanding of the value of spike sorting. A few of those studies in particular are helpful to compare with the findings from our information-theoretic analysis.

Some BCI-related studies have reported measurements of mutual information in motor cortical populations for movement tasks, but the information was estimated from predicted motor output; i.e., population information was estimated by computing information between a target signal and a reconstruction of the target from a population of neurons [106, 105, 55]. This method quantifies the information extracted by the decoder but does not measure the actual information contained in the population activity. In one study on cosine-tuning and population vector modelling of motor control [105], MI was computed between the target signal and the decoded signal based on a population vector model. There is no way to verify in these *in vivo* experiments how well the decoders extract the population information.

The difficulty in increasing the population size is that the amount of data re-

quired to compute mutual information grows exponentially with the number of neurons. While four neurons certainly does not comprise a large population of signals, no study we have found shows information measurements directly computed on spike activity of 4 or more neurons without bias estimates. The limited amount of biological data available has been cited as the reason for estimating information and bias corrections. The simulations in Chap. 5 utilized the capability to simulate data of sufficient length to compute information directly on signals with realistic firing rates. Even with the ability to control input and output of the model neural system and to generate much more data under a controlled paradigm, the amount of data required for direct computation of information was estimated to be $1.6 \cdot 10^6$ samples of data per target. Generating such large quantities of data tested the limits of the computer used for these simulations and would be even more difficult to acquire *in vivo*. Our studies show the need for good information estimates for populations much larger than 4 neurons. The information loss in the 4-unit population in our simulations behaved similarly to 2-unit information loss; we suggest that information in 3 or more pairs of neurons would behave similarly and leave that analysis for future exploration. Information that is computed on *in vivo* signals will require bias estimates. However, the accuracy of these bias estimates cannot be verified because the true answers remain unknown [48]. Rigorous information-theoretic studies that have been applied to neurophysiologic data have focused on producing computationally efficient information estimates given small data sizes [48]. The estimates have been compared to information when neurons were treated independently and on the responses of 1 or 2 neurons [53]. None of these methods have been applied to cosine-tuned responses in the motor cortex or on true population spike activity.

6.4 Implications for optimal coding of biological neurons

While the main purpose of this work was to aid BCI design by providing an analytical understanding of spike sorting, we can make inferences about the behavior of coding by pooled neural activity from our results. First, we note that even with the noisiness of individual Poisson neurons, which have low information content, pooling activity can increase information through redundancy. The ability of a cluster of pooled units to carry information is furthered by adding more clusters. We estimated the number of pooled clusters of redundantly-tuned neurons needed to achieve 95% of the information capacity to be 25 (sec. 5.4.6). Multi-unit signals can, at best, maintain the information of the maximally-encoding single unit among the n pooled units. Using all n units could yield $\sqrt{n_{upc}}$ times more information. In this case, the number of non-sorted channels needed would be n_{upc} times the number of sorted channels needed to achieve the same information. It is still possible, however, for non-sorted multi-unit recordings to be more economical because n_{upc} is likely to be 2-3, while the sampling and processing rate needed for a channel with sorting is roughly 32 times that of a channel without sorting.

The sensitivity of mutual information in individual spike trains to spike sorting error could indicate that the error will reduce information content so much as to render the discriminated signals meaningless. It could also indicate that since the loss or addition of even a single spike could result in a significant loss of valuable information, individual spikes from every individual neuron is crucial to reconstructing the encoded information. The latter explanation would imply that each neuron's response must be precise and reliable, whereas the former explanation is in accord with the idea that the brain does not rely on a code that can be so easily corrupted but has large populations of neurons transmitting the same information in a more

robust code. This begs the question of whether single unit trains and preserving their identity are crucial to information processing and whether the population response as a whole masks the details of individual neuron responses. These are matters of neural encoding that scientists continue to investigate and debate.

In the cosine-tuning simulations of Chapter 5, we demonstrated the ability of multiple neurons to compensate for the noisness of single neurons, and the ability of redundancy to help reduce some of that information loss. The amount of information lost by pooling depended directly on the difference between the preferred directions of units being pooled. Pooling activity from pairs of neurons carrying motor information in cosine-tuned responses resulted in 56% information loss when the directional tuning of the neurons differed by 90^o, and 0.7% information loss when the tuning of the neurons was equal. When preferred directions were equal within electrodes, each electrode had 60% more information than the individual units alone for all angular differences, and the absolute preferred directions did not affect the information of a given pair (Fig. 5.8f).

Changes in MI due to pooling depended primarily on the differences in angular tuning. However, MI in individual units was shown to also depend on mean firing rate and modulation depth. Further investigation revealed the interplay between the Poisson nature and cosine-tuning properties of the spike activity in affecting these information differences in pooled and labelled-line activity. Information was maximized when modulation depth was maximized for a given mean rate and the mean rate was minimized for a given modulation depth. In other words, the range should be maximized for a given total amount of firing. Another way to say it is that if the range of response is restricted, then the least amount of firing should be used to achieve that range. Thus, if neural responses are truly Poisson, then the optimal information is obtained when the system is also in the lowest possible energy state.

If we allow ourselves to hypothesize a little bit, one feasible solution for efficient neural encoding would be for post-synamptic neurons to integrate currents from many presynaptic neurons. We are unsure how neurons would distinguish between incoming presynaptic inputs, but if extracellular currents are integrated by a neuron, then the best situation energy-wise for the neural network is also the one that provides maximum information. Redundancy of presynaptic currents pooled by a post-synaptic neuron allows reinforcement of noisy responses, while diversity in responses that are not pooled but are used jointly with the other pooled responses help create the full robust population response.

6.5 Concluding remarks

Groundbreaking work has been performed over many decades by many great scientists to bring about the possibility of multi-neuron recordings, neuromotor prostheses, and spike sorting. Multi-neuron recordings have enabled efforts to decipher the language by which neural ensembles process information and control behavior. Consequently, computational algorithms are being developed to interpret these population signals with the long-term goal of restoring motor control to movement-impaired patients. The goal of the first step in these computational decoders is to reproduce the activity of individual neurons. Yet, decoders using the aggregate activity of clusters of neurons obtained by multi-electrode array recordings without spike sorting have unexplored potential in BCIs. Little analytical work had been conducted to determine and understand the advantage spike sorting would provide even if it was done perfectly, with 100% accuracy and completely unsupervised. The possibility of maintaining the information content of each neuron, without the cost of sorting, provided motivation for us to analyze the information available in multi-unit signals and consider the possibility of using multi-unit signals in place of spike sorted signals. The feasibility

of using multi-unit recordings has not yet been directly demonstrated with *in vivo* data.

The analyses presented in this dissertation were aimed quantifying the effects of spike sorting on a BCI's ability to translate extracellular electrical activity into volitional movement. Discarding information that might be available in the identity of individual neurons appeared to be a detrimental source of information loss in a BCI system. However, two areas in BCI information processing were identified which might reduce the severity of MI loss in the absence of spike sorting. First, spike sorting may also present a source of significant information loss through classification error. Secondly, MI loss may not be severe when pooling neurons if those neurons have similar tuning. Information was found to be very sensitive to spike error, and could be significantly lost for typical amounts of spike sorting error. Thus, the amount of information in pooled multi-unit signals was investigated. MI in pooled spike activity was measured to develop a clearer understanding of the information loss that pooling incurs. The particular method employed allows us to view the measurements as the mutual information extracted by a linear decoder. While we do not know whether a linear decoder is the best approach in a BCI, these measurements are relevant to a BCI application in the absence of more knowledge about motor encoding. Linear decoders are the most common BCI decoders which have had success; thus, the results tell us what information pooled signals make available to current BCI decoders. Multi-unit signals were unable to preserve all the MI in the constituent single-units unless there were small differences between the single-unit MI. MI loss due to pooling was directly related to these differences.

We then examined these properties of information in pooled signals in the context of a cosine-tuned population. We simulated cosine-tuned activity and found the tolerance of preferred direction difference between pooled neurons under differ-

ent conditions of varying inter-electrode tuning differences, amounts of sorting error, mean levels, and modulation depths. We found that information was preserved in multi-unit signals in cases that we believe are possible based on the literature, namely that neurons in a local region are similarly tuned. We also found in sections 3.3 and 5.3 that information content has a low tolerance for spike sorting error. These findings motivate further examination of the encoding capacity of discriminated versus non-discriminated extracellular neural responses, so that the feasibility of an efficient multi-neuron recording and processing system utilizing multi-unit signals can be explored.

While the results in these studies might be easily explained intuitively, it was not obvious how to think about the effects of spike sorting prior to performing the analysis. Spike sorting continued to linger as a problem to be dealt with in the process of miniaturizing BCIs into an implantable system. These studies have brought the problem of spike sorting in a BCI to the foreground. We have begun the process of considering the many factors that play into spike sorting's role in a BCI and its benefit. The considerations that we have made begin to resolve the question of what role spike sorting should play in a BCI. Bypassing spike sorting in BCI applications would allow reductions in both size and complexity of high-demand signal conditioning, digitizing, and telemetry hardware. Instead of acquiring and transmitting high-resolution streams of data, a detector could be implemented through thresholding with less demanding hardware and transmission of a single bit when an action potential occurs. It is, therefore, important to consider the effects of spike processing in a BCI analytically to make the best design decisions for a BCI system, and potentially bring the realization of a fully implantable and usable BCI to patients more quickly.

Our analyses indicate that information loss due to pooling would be acceptable

only in particular cases. We have described those cases and found that they are not altogether unrealistic but restrictive enough that we could not confidently eliminate spike sorting from BCIs without expecting its performance to suffer to an unacceptable degree. Pooling could be tolerated if (1) classification error significantly exceeded detection error, (2) the motor cortex was somatotopically mapped with a gradient not exceeding 45^o per $200\mu m$, (3) the target motor variable to be decoded by the BCI was encoded in the magnitude of neural firing, and (4) the corrupting activity of non-redundant or independently-encoding neurons within a pool of neurons was masked by a neuron relevant to the task. We determined the tolerances for combinations of sorting error, detection error, and intra-electrode angular tuning differences for $< 5 - 10\%$ mutual information loss. Based on the knowledge we have from published *in vivo* results thus far, we do not advise leaving spike sorting out of a BCI, but instead recommend pursuing efforts to reduce error in spike sorting or develop decoding algorithms that are robust against such error for future BCI development.

Bibliography

[1] Wiest MC, Bentley N, and Nicolelis MAL, "Heterogeneous integration of bilateral whisker signals by neurons in primary somatosensory cortex of awake rats," *J. Neurophysiology*, vol. 93, pp. 2966–2973, 2005.

[2] CoGain, "Report on a market study and demographics of user population," *Information Society of Technologies: Network of Excellence*, vol. IST-2003-511598 (NoE), pp. 1–32, 2004.

[3] Kennedy PR and Adams KD, "A decision-tree for brain-computer interface devices," *IEEE Systems and Rehabilitation Engineering*, vol. 11, no. 2, pp. 148–150, 2003.

[4] "www.christopherreeve.org," Tech. Rep., Christopher Reeve Paralysis Foundation.

[5] "Spinal cord injury facts and figures at a glance," Tech. Rep., Spinal Cord Injury Information Network, May 2001.

[6] Nicolelis MAL, "Brain-machine interfaces to restore motor function and probe neural circuits," *Nature Reviews – Neuroscience*, vol. 4, pp. 417–422, May 2003.

[7] PBS, "I might walk ," Tech. Rep., PBS - Scientific American Frontiers, 2001.

[8] Birbaumer N, "Brain-computer-interface research: Coming of age," *Clinical Neurophysiology*, vol. 117, no. 3, pp. 479–483, 2006.

[9] Vaughan TM, Heetderks WJ, Trejo LJ, Rymer WZ, Weinrich M, Moore MM, Kübler, Dobkin BH, Birbaumer N, Donchin E, Wolpaw EW, and Wolpaw JR, "Brain-computer interface technology: review of the second international meeting," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 11, no. 2, pp. 94–109, 2003.

[10] Wessberg J, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature*, vol. 408, pp. 361–5, November 2000.

[11] Taylor DM, Helms Tillery SI, and Schwartz AB, "Direct cortical control of 3d neuroprosthetic devices," *Science*, vol. 296, pp. 1829–1832, June 2002.

[12] M Serruya, N Hatsopoulos, Paninski, M Fellows, and J Donoghue, "Instant neural control of a movement signal," *Nature*, vol. 416, pp. 141–142, March 2002.

[13] Serruya MD, Caplan AH, Saleh M, Morris DS, and Donoghue JP, "The braingate pilot trial: building and testing a novel direct neural output for patients with severe motor impairment," .

[14] "Cyberkinetics neurotechnology systems, inc," .

[15] Nicolelis MAL, Ghazanfar AA, Faggin BM, Votaw S, and Oliveira LMO, "Reconstructing the engram: simultaneous, multisite, many single neuron recordings," *Neuron*, vol. 18, pp. 529–537, 1997.

[16] Nicolelis MALN et al, "Grant application for darpa n66001-02-c-8022 to maln," Tech. Rep., Duke University, 2001.

[17] GL Gerstein and WA Clark, "Simultaneous studies of firing patterns in several neurons," *Science*, vol. 14, pp. 1325–1327, March 1964.

[18] BC Wheeler and WJ Heetderks, "A comparison of techniques for classification of multiple neural signals," *IEEE Transactions in Biomedical Engineering*, vol. BME-29, no. 12, pp. 752–759, 1982.

[19] MS Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials," *Network: Comput. Neural Syst.*, vol. 9, pp. 53–78, 1998.

[20] KD Harris, DA Henze, J Csicsvari, H Hirase, and G Buzsáki, "Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements," *J Neurophysiol*, vol. 84, pp. 401–414, 2000.

[21] Jung HK, Choi JH, and Kim TJ, "Solving alignment problems in spike sorting using frequency domain pca," *Neurocomputing*, vol. 69, pp. 975–978, 2006.

[22] Quiroga RQ, Nadasday Z, and Ben-Shaul Y, "Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering," *Neural Computation*, vol. 16, pp. 1661–1687, 2004.

[23] H. Wiggins, "Multichannel acquisition and processor system data sheet," Tech. Rep., Plexon, Inc., Dallas, Texas.

[24] Obeid I, Nicolelis MAL, and Wolf PD, "A multichannel telemetry sytstem for single unit neural recordings," *J Neurosci Meth*, vol. 133, no. 1-2, pp. 33–38, February 2004.

[25] Obeid I, Nicolelis MAL, and Wolf PD, "A low-power multi-channel analog front end for portable neural signal recordings," *J Neurosci Meth*, vol. 133, no. 1-2, pp. 27–32, February 2004.

[26] Jochum TA, Arora H, Callender SH, and Wolf PD, "A 16-channel custom integrated chip for processing neural signals in a brain-machine interface," *IEEE BMES Conference*, 2006.

[27] M Abeles and GL Gerstein, "Detecting spatiotemporal firing patterns among simultaneously recorded single neurons," *Journal of Neurophysiology*, vol. 60, no. 3, pp. 909–924, 1988.

[28] GL Gerstein, *The Neurosciences Second Study Program: Communciation and coding in the nervous system*, chapter Functional association of neurons: Detection and interpretation, pp. 648–661, Rockefeller University Press, New York, 1970.

[29] M. Abeles and Jr. M. Goldstein, "Multispike train analysis," *Proceedings of the IEEE*, vol. 65, pp. 762–773, 1977.

[30] G. L. Gerstein and D. H. Perkel, "Simultaneously recorded trains of action potentials: analysis and functional interpretation," *Science*, vol. 164, pp. 828–830, 1969.

[31] W. M. Roberts and D. K. Hartline, "Separation of multi-unit nerve impulse trains by a multi-channel linear filter algorithm," *Brain Research*, vol. 94, pp. 141–149, 1975.

[32] Nicolelis MAL and Chapin JK, "Controlling robots with the mind," *Scientific American*, September 16 2002.

[33] TM Cover and JA Thomas, *Elements of Information Theory*, John Wiley & Sons, Inc., New York, 1991.

[34] Strong SP, de Ruyter van Steveninck RR, Bialek W, and Koberle R, "On the application of information theory to neural spike trains," *Proceedings of the Pacific Symposium on Biocomputing*, vol. 3, pp. 619–630, 1998.

[35] F Rieke, D Warland, R de Ruyter van Steveninck, and W Bialek, *Spikes: Exploring the Neural Code*, The MIT Press, Cambridge, Massachusetts, 1997.

[36] Naselaris T, Merchant H, Amirikian B, and Georgopoulos AP, "Large-scale organization of preferred directions in the motor cortex. ii. analysis of local distributions," *J Neurophysiol*, vol. 96, pp. 3237–3247, 2006.

[37] CE Shannon, "A mathematical theory of communication," *Bell Sys. Tech. Journal*, vol. 27, pp. 379–423, 1948.

[38] DH Perkel, *The Neurosciences Second Study Program: Communciation and coding in the nervous system*, chapter Spike Trains as Carriers of Information, pp. 587–596, Rockefeller University Press, New York, 1970.

[39] Brunel N and Nadal J-P, "Mutual information, fisher information, and population coding," *Neural Computatino*, vol. 10, pp. 1731–1757, 1998.

[40] Abbott LF and Dayan P, "The effect of correlated variability on the accuracy of a population code," *Neural Computation*, vol. 11, pp. 91–101, 1999.

[41] Averbeck BB and Lee Daeyeol, "Coding and transmission of information by neural ensembles," *TRENDS in Neurosciences*, vol. 27, no. 4, pp. 225–230, 2004.

[42] Shadlen MN and Newsome WT, "The variable discharge of cortical neurons: implications for connectivity, computation, and information coding," *The Journal of Neuroscience*, vol. 18, no. 10, pp. 3870–3896, 1998.

[43] Salinas E and Sejnowski TJ, "Impact of correlated synaptic input on output firing rate and variability in simple neuronal models," *The Journal of Neuroscience*, vol. 20, no. 16, pp. 6193–6209, 2000.

[44] Zeitler M, Fries P, and Gielen S, "Assessing neuronal coherence with single-unit, multi-unit, and local field potentials," *Neural Computation*, vol. 18, pp. 2256–2281, 2006.

[45] Brown EN, Kaas RE, and Mitra P, "Multiple neural spike train data analysis: State-of-the-art and future challenges," *Nature Neuroscience*, vol. 7, pp. 456–461, 2004.

[46] MJ Chacron, "Nonlinear information processing in a model sensory system," *J Neurophysiol*, vol. 95, pp. 2933–2946, 2006.

[47] Strong SP, Koberle R, and de Ruyter van Steveninck RR and, "Entropy and information in neural spike trains," *Phys. Rev. Lett.*, vol. 80, pp. 197, 1998.

[48] L. Paninski, "Estimation of entropy and mutual information," *Neural Computation*, vol. 15, pp. 1191–1253, 2003.

[49] F Rieke, DA Bodnar, and W Bialek, "Naturalistic stimuli increase the rate and efficiency of information transmission by primary auditory affernts," *Proceedings of the Royal Society London B*, vol. 262, pp. 259–265, 1995.

[50] F. Gabbiani R. Wessel, C. Koch, "Coding of time-varying electric field amplitude modulations in a wave-type electric fish," *Journal of neurophysiology*, vol. 75, pp. 2280–2293, 1996.

[51] Y Tock and G Inbar, *Modern Techniques in Neuroscience Research*, chapter Information-theoretical analysis of sensory information, Springer-Verlag, Berlin, Germany, 1999.

[52] Schneidman Bialek and Berry, "Synergy, redundancy, and independence in population codes," *Journal of Neuroscience*, vol. 23, pp. 11539–11553, 2004.

[53] G Pola, RS Petersen, A Thiele, MP Young, and S Panzeri, "Data-robust tight lower bounds to the information carried by spike times of a neuronal population," *Neural Computation*, vol. 17, pp. 1962–2005, 2005.

[54] I Nemenman, W Bialek, and Rob de Ruyter van Steveninck, "Entropy and information in neural spike trains: Progress on the sampling problem," *Physical Review E*, vol. 69, 2004.

[55] D. Warland, P. Reinagel, and M. Meister, "Decoding visual information from a population of retinal ganglion cells.," *Journal of Neurophysiology*, vol. 78, pp. 2336–2350, 1997.

[56] Reinagel P and Reid RC, "Temporal coding of visual information in the thalamus," *Journal of neuroscience*, vol. 20, pp. 5392–5400, 2000.

[57] Reich DS, Mechler F, and Victor JD, "Formal and attribute-specific information in primary visual cortex," *J. Neurophys.*, vol. 85, pp. 305–318, January 2001.

[58] E Arabzadeh, S Panzeri, and ME Diamond, "Whisker vibration information carried by rat barrel cortex neurons," *The Journal of Neuroscience*, vol. 24, pp. 6011–6020, 2004.

[59] Rojiani, *Programming in C with Numerical Methods for Engineers*, Prentice Hall, New Jersey, 1996.

[60] Kim KA and Kim SJ, "Method for unsupervised classification," *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 4, pp. 421–431, 2003.

[61] Choi JH, Jung HK, and Kim TJ, "A new action potential detector using the mteo and its effects on spike sorting systems at low signal-to-noise ratios," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 4, pp. 738–746, 2006.

[62] AM Marmlouk et al, "Unsupervised spike sorting with ica and its evaluation using genesis simulations," *Neurocomputing*, vol. 65-66, pp. 275–282, 2005.

[63] F Wood, MJ Black, C Vargas-Irwin, M Fellows, and JP Donoghue, "On the variability of manual spike sorting," *IEEE Trans on BME*, vol. 51, pp. 912–918, 2004.

[64] A Luczak and NS Narayanan, "Spectral representation – analyzing single-unit activity in extracellulary recorded neuronal data without spike sorting," *Journal of Neuroscience Methods*, vol. 144, pp. 53–61, 2005.

[65] Poor HV, *An Introduction to Signal Detection and Estimation*, Dowden & Culver, Inc., New York, 1988.

[66] Reich DS, Mechler F, and Victor JD, "Independent and redundant infomration in nearby cortical neurons," *Science*, vol. 294, pp. 2566–2568, December 2001.

[67] Seung S, "Correlation, convolution, and the wiener-hopf equations," http://hebb.mit.edu/courses/9.29/lectures/lecture02.html.

[68] Butts DA, "How much information is associated with a particular stimulus?," *Network: Computation in Neural Systems*, vol. 14, pp. 35–60, 2003.

[69] G Pola, A Thiele, and K-P Hoffmann, "An exact method to quantify the information transmitted by different mechanisms of correlation coding," *Network: Computation in Neural Systems*, vol. 14, pp. 35–60, 2003.

[70] Georgopoulos AP, Kettner RE, and Schwartz AB, "Primate motor cortex and free arm movements to visual targets in three-dimensional space: Ii. coding of the direction of movement by a neuronal population," *Journal of Neuroscience*, vol. 8, no. 8, pp. 2928–2937, 1988.

[71] Carmena JM, Lebedev MA, Crist RE, O'Doherty JE, Santucci DM, Dimitrov DF, Patil PG, Henriquez CS, and Nicolelis MAL, "Learning to control a brain-machine interface for reaching and grasping by primates," *PLoS Biology*, vol. 1, pp. 1–16, 2003.

[72] N. Hatsopoulos, J. Joshi, and J.G. O'Leary, "Decoding continuous and discrete motor behaviors using motor and premotor cortical ensembles," *PLoS Biology*, vol. 1, pp. 1–16, 2003.

[73] Ben-Shaul Y, Stark E, Asher I, Drori R, Nadasdy Z, and Abeles M, "Dynamical organization of directional tuning in the primate premotor and primary motor cortex," *J Neurophysiol*, vol. 89, pp. 1136–1142, 2003.

[74] Georgopoulos AP, Kalaska JF, Caminiti R, and Massey JT, "On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex," *Journal of Neuroscience*, vol. 2, no. 11, pp. 1527–1537, 1982.

[75] Kemere C, Shenoy KV, and Meng TH, "Model-based neural decoding of reaching movements: a maximum likelihood approach," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 6, pp. 925–932, 2004.

[76] Reinkensmeyer D, Iobbi M, Kahn L, Kamper D, and Takahashi C, "Modeling reaching impairment after stroke," *Neural Computation*, vol. 15, pp. 2619–2642, 2003.

[77] Schwartz AB, "Direct cortical representation of drawing," *Science*, vol. 265, pp. 540–542, 1994.

[78] TM Cover and JA Thomas, *Elements of Information Theory*, chapter 2, pp. 12–21, John Wiley & Sons, Inc., New York, 1991.

[79] Georgopoulos AP, Schwartz AB, and Kettner RE, "Neuronal population coding of movement direction," *Science*, vol. 233, pp. 1416–1419, 1986.

[80] Schwartz AB, Taylor DM, and Tillery SIH, "Extraction algorithms for cortical control of arm prosthetics," *Current Opinion in Neurobiology*, vol. 11, pp. 701–707, 2001.

[81] Obeid I and Wolf PD, "Comparison of spike detection algorithms applied to *in vivo* data," *to be submitted*.

[82] Obeid I and Wolf PD, "Real-time spike detection for data reduction in wireless brain machine interface recordings," *to be submitted*.

[83] Shimazaki H and Shinomoto S, "A method for selecting the bin size of a time histogram," *Neural Computation*, in press (2006?).

[84] Wolpaw J and McFarland D, "Control of a two-dimensional movement signal by a non-invasive brain-computer interface in humans," *Proc Natl Acad Sci*, vol. 101, pp. 49–54, 2005.

[85] Kennedy PR, Bakay RA, Moore MM, Adams K, and Goldwaithe J, "Direct control of a computer from the human central nervous system," *IEEE Trans Neural Syst Rehabil Eng*, vol. 8, pp. 198–202, 2000.

[86] Kim KH, Kim SS, and Kim SJ, "Improvement of spike train decoder under spike detection and classification errors using support vector machine," *Med Biol Eng Comput*, vol. 44, pp. 124–130, 2006.

[87] DH Hubel and TN Wiesel, "Receptive fields, binocular interaction and functional architecture in cats visual cortex," *J. Physiol. Lond.*, vol. 160, pp. 106–154, 1962.

[88] VB Mountcastle, "Modality and toographic properties of single neurons of cat's somatic sensory cortex," *J Neurophsiol*, vol. 20, pp. 408–434, 1957.

[89] TA Woolsey and H Van der Loos, "The structural organization of layer iv in the somatosensory region (si) of mouse cerebral cortex. the description of a cortical field composed of discrete cytoarchitectonic units," *Brain Research*, vol. 17, pp. 205–242, 1970.

[90] Purves D et. al., *Neuroscience*, chapter 8 and 10 and 12, Sinauer Associates, Inc., Sunderland, MA, 1997.

[91] S Panzeri, F Petroni, RS Petersen, and ME Diamond, "Decoding neuronal population activity in rat somatosensory cortex: role of columnar organization," *Cerebral Cortex*, vol. 13, pp. 45–52, 2003.

[92] Won DS, O'Doherty JE, Carmena JM, Phelps EE, Nicolelis MAL, Henriquez CS, and Wolf PD, "A comparison of linear predictor performance using unsorted and sorted neural spike activity," presented at the Society for Neuroscience Annual Meeting, October 2004.

[93] Purves D et. al., *Neuroscience*, chapter 7, p. 129, Sinauer Associates, Inc., Sunderland, MA, 1997.

[94] Haft M, "Robust "topological" codes by keeping control of internal redundancy," *Phys. Rev. Letters*, vol. 81, pp. 4016–4019, 1998.

[95] Pinilla-Dutoit J and Woolley SI, "Robust perceptual coding of overcomplete frame expansions," *Proc. of SPIE*, vol. 4299, pp. 143–9, 2001.

[96] J-M Fellous, P.H.E. Tiesinga, P.J. Thomas, and T.J. Sejnowski, "Discovering spike patterns in neuronal responses," *The Journal of Neuroscience*, vol. 24, no. 12, pp. 2989–3001, March 2004.

[97] E. Todorov, "On the role of the primary motor cortex in arm movement control," *Progress in Motor Control III*, vol. 124, pp. 766–74, 2003.

[98] Mauk M, "The potential effectiveness of simulation versus phenomenological models," *Nature Neuroscience*, vol. 3, pp. 649–651, 2000.

[99] Gerstein GL and Kiang NYS, "An approach to the quantitative analysis of electrophysiological data from single neurons," *Biophysical Journal*, vol. 1, no. 1, pp. 15–28, 1960.

[100] Shimazaki H and Shinomoto S, "A recipe for optimizing a time-histogram," *Advances in Neural Information Processing Systems*, vol. 19, 2007.

[101] Dayan P and Abbott L, *Theoretical Neuroscience*, chapter 1, The MIT Press, 2001.

[102] Donoghue JP, "Connecting cortex to machines: recent advances in brain interfaces," *Nature*, vol. 5, pp. 1085–1088, 2002.

[103] Schwartz AB, "Cortical neural prosthetics," *Annu. Rev. Neurosci.*, vol. 27, pp. 487–507, 2004.

[104] Logothetis NK, "The neural basis of the bold fmri signal," *Phil. Trans. R. Soc. Lond. B*, vol. 357, pp. 1003, 2002.

[105] Georgopoulos A and Massey J, "Cognitive spatial-motor processes: 2. information transmitted by the direction of two-dimensional arm movements and by neuronal populations in primate motor cortex and area 5," *Experimental Brain Research*, vol. 69, no. 2, pp. 315–326, 1987.

[106] Taylor DM, Helms Tillery SI, and Schwartz AB, "Information conveyed through brain-control: cursor versus robot," *Journal of neuroscience*, vol. 8, no. 8, pp. 2913–2927, 1988.

Biography

If I had more time, I might have written about some of the memories I have of my parents providing a nurturing environment and instilling in me the desire to always try my best, and of my sisters acting as my teachers and playmates growing up. And I would need still more time and space to write about Alex and how everything I did became much more rich and meaningful once he began to join me (or should I say, once he finally agreed to let me drag him around with me?). There are so many others, too, who have just made life worth living – Joyce, Michael R., Scott, Liane, April, Shelia, Gordon, Mike G., Fionn, Stan, Helen, Dr. Hall, Chris Hooke and family, friends from ICF, CrossRoads, Alpha, UMD, SG .... All of these people and experiences have shaped who I am ... *much* more so than my birthrights as a Texan! But I think the apostle Paul captures best the sentiments I have about who I am in Galatians 2:19-20: *" For through the law I died to the law, so that I might live to God. I have been crucified with Christ. It is no longer I who live, but Christ who lives in me. And the life I now live in the flesh I live by faith in the Son of God, who loved me and gave himself for me."* Now, I eagerly anticipate what lies ahead. *"[N]ot that I have already obtained this or already been made perfect, but I press on to take hold of that for which Christ Jesus has laid hold of me.... I do not consider myself to have already obtained it, but one thing I do: forgetting what is behind and straining forward to what lies ahead, I press on toward the goal for the prize of the upward call of God in Christ Jesus."* (Phillipians 3:12-14).