BOSTON UNIVERSITY

GRADUATE SCHOOL OF ARTS AND SCIENCES

Dissertation

AN ELECTROPHYSIOLOGICAL INVESTIGATION OF HUMAN MOTOR CORTEX AND ITS APPLICATION TO SPEECH RESTORATION

by

JONATHAN S. BRUMBERG

B.S., University of Delaware, Newark, DE, 2002
B.A., University of Delaware, Newark, DE, 2002

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

2009

Approved by

First Reader

Frank H. Guenther, PhD
Associate Professor of Cognitive and Neural Systems
Boston University

Second Reader

Philip R. Kennedy, MD, PhD
Chief Scientist
Neural Signals, Inc.

Third Reader

Michael A. Cohen, PhD
Associate Professor of Cognitive and Neural Systems
and Computer Science
Boston University

ACKNOWLEDGMENTS

This research would not be possible if not for the tireless effort of Erik and Eddie Ramsay. Their dedication is truly inspiring and I thank them both very much for allowing me to participate in a project that was and continues to be a very personal affair.

The work described in the following dissertation was partially funded by grants from the National Institutes of Health R01 DC-007683 (F. Guenther, PI), the National Science Foundation SBE-0354378 (S. Grossberg, PI) and the National Institute on Deafness and other Communication Disorders R44 DC007050-02 (P. Kennedy, PI).

No project, dissertation or not, is conducted entirely by a single individual, and this dissertation research was no exception. I am grateful for the help and support from a vast network of researchers. First I would like to thank my advisor, Frank Guenther, and the director of Neural Signals, Inc., Philip Kennedy, for giving me the opportunity to work on such an exciting and promising area of research. I have received incredible support from my labmates (both current and former) with special thanks to Alfonso Nieto-Castanon and Jason Tourville. In addition, I am fortunate to be a part of the lab at Neural Signals and I especially thank them for their endless supply of answers to every question I could come up with. I would also like to recognize my dissertation committee for their insightful conversations that have helped to shape the directions taken in this dissertation.

Last I wish to thank my wife, Margaret. Five years ago, fresh out of college, we packed up and moved to Boston so I could begin my graduate study. My life would not be the same without her unconditional love and support and the confidence the instills in me every day.

AN ELECTROPHYSIOLOGICAL INVESTIGATION OF HUMAN MOTOR CORTEX AND ITS APPLICATION TO SPEECH RESTORATION

(Order No.)

JONATHAN S. BRUMBERG

Boston University Graduate School of Arts and Sciences, 2009

Major Professor: Frank H. Guenther, Associate Professor of Cognitive and Neural Systems

ABSTRACT

This dissertation provides a means for restoration of speech production for humans with profound paralysis through microelectrode recording and prediction of intended speech from correlated neural activity. The basis for this research comes from a human subject with locked-in syndrome implanted with a neurotrophic electrode (Kennedy and Bakay, 1998). Most locked-in patients have complete loss of voluntary movement, but have fully intact sensory and cognitive function. Therefore, it is possible, in this dissertation, to conduct scientific experimentation, as the subject is capable of perceiving and understanding instructions and provides debriefing information through limited eye movements.

The design and implementation of the neural prosthesis had three main parts. The first involved isolation of single units from an extracellular multiunit recording. A total of 56 neural units were identified from recordings of human speech motor cortex using the neurotrophic electrode during a range of neurological and cognitive states.

The second component of the neural prosthesis design involved the analysis of

task-specific neural units. The methodology employed relates speech production to previous studies of motor execution and utilizes recent algorithms for prediction of motor behavior from extracellularly recorded action potentials. The preferred states (e.g. preferred direction and position) of the neural units were found to have broad, though not uniform, coverage over the range of movements of the vocal tract necessary for speech production.

The final element in the implementation of the neural prosthesis was the application of results from the first two stages to the creation of a real-time device for decoding speech from neural activity and subsequent computer speech synthesis. Single unit activity was decoded into continuous-state acoustic measures of speech (i.e. formant frequencies) in a real-time environment and synthesized. The subject participated in an acoustic center-out task (c.f. Georgopoulos et al., 1986) consisting of three different vowel-vowel utterances. Decoding performance steadily improved over a two month trial period. The subject continues to participate in the study utilizing the neural prosthesis for speech production.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

2D	.....	Two dimensional
3D	.....	Three dimensional
A1	.....	First formant amplitude
A2	.....	Second formant amplitude
A3	.....	Third formant amplitude
ABS	.....	Absolute value
AIC	.....	Akaike information criterion
AF	.....	Frication amplitude
AV	.....	Voicing amplitude
ALS	.....	Amyotrophic lateral sclerosis
ANOVA	.....	Analysis of variance
BCI	.....	Brain computer interface
BIC	.....	Bayesian information criterion
CDF	.....	Cumulative distribution function
DIVA	.....	Directions into velocities of articulators
DTW	.....	Dynamic time warping
dPMC	.....	Dorsal premotor cortex
ECoG	.....	Electrocorticography/electrocorticogram
EEG	.....	Electroencephelograpy/electroencephelogram
EM	.....	Expectation-maximization

F1	.....	First formant frequency
F2	.....	Second formant frequency
F3	.....	Third formant frequency
fMRI	.....	Functional magnetic resonance imaging
G	.....	Gamma
GLM	.....	Generalized linear model
GOF	.....	Goodness-of-fit
HMM	.....	Hidden Markov model
ICI	.....	Intersection of confidence intervals
IG	.....	Inverse gaussian
IP	.....	Inhomogeneous Poisson
ISI	.....	Interspike interval
KDE	.....	Kernel density estimate
K-S	.....	Kolmogorov-Smirnov
LDA	.....	Linear discriminant analysis
LFP	.....	Local field potential
LG	.....	Linear gaussian
LGM	.....	Linear gaussian model
LTI	.....	Linear time invariant
MI	.....	Primary motor cortex
MISE	.....	Mean integrated squared error
ML	.....	Maximum likelihood
MLE	.....	Maximum likelihood estimation

NEO	.....	Nonlinear energy operator
PCA	.....	Principal components analysis
PMC	.....	Premotor cortex
PSTH	.....	Peristimulus time histogram
PV	.....	Population vector
PVA	.....	Population vector algorithm
Q-Q	.....	Quantile-quantile
RMS	.....	Root-mean-square
RMSE	.....	Root-mean-squared error
rRMS	.....	Running root-mean-square
SNEO	.....	Smoothed nonlinear energy operatory
SNR	.....	Signal-to-noise
SPC	.....	Superparamagnetic clustering
vPMC	.....	Ventral premotor cortex

CHAPTER 1

INTRODUCTION

The primary goal of this research is to develop a neural prosthesis for restoration of speech production in a human subject with profound paralysis. Every year, thousands of people in the United States and world-wide are diagnosed with neurologically debilitating diseases such as amyotrophic lateral sclerosis (ALS)[1] or suffer traumatic brain injuries (e.g. brain-stem stroke) rendering them "locked-in." Though locked-in patients are deprived of most, if not all, of their voluntary motor control, their cognitive faculties are spared. This condition effectively removes all forms of interpersonal communication between victims and the outside world. However, as their brain activity remains neurologically normal these patients make excellent candidates for attempts to restore some motor function aided by brain computer interfaces (BCI). Design of the neural prosthesis in this dissertation takes advantage of recent innovations in human microelectrode recordings and advances in neural decoding methodology. The result of this dissertation research was the implementation and production of the first ever neural prosthesis for continuous computer-aided speech synthesis.

For many years researchers in the fields of neurophysiology and brain computer interfacing have been trying to characterize the relationship between various portions of the central nervous system and behavior, such as effects of retinal stimulation on ganglion cell activation (Warland et al., 1997), spatial location and hippocampus

[1]There 20,000 active cases of ALS in the United States alone and 5,000 new diagnoses each year (National Institute of Neurological Disorders and Stroke, 2008).

activity (Brown et al., 1998), and activation of the motor cortices during complex arm and hand movements (Georgopoulos et al., 1982, 1986; Kennedy et al., 1992a, 2000; Moran and Schwartz, 1999a,b; Carmena et al., 2003; Brockwell et al., 2004; Paninski et al., 2004; Truccolo et al., 2005; Hochberg et al., 2006; Wu et al., 2006; Kim et al., 2007; Velliste et al., 2008). The results of these studies provide rough guidelines for development of brain computer interfaces for individuals with profound impairment. To that end, the direction of modern brain computer interface development can be divided into two main groups, those solving problems of impaired *inputs* and *outputs* of biological systems. In this terminology, the inputs to a biological system are its senses (i.e. sight, audition, tactile sensation, etc.) while the outputs are the end-effectors of the motor system.

A very common example of a BCI for solving an input problem is the cochlear implant. The device is designed for individuals who have a damaged cochlea, rendering them unable to sense sound, but have intact supporting neural architecture (i.e. auditory nerve) and higher brain function. For these patients, the device processes sound information and stimulates the appropriate cells in the cochlea imitating the response of a fully intact auditory system. Though no widespread publicly available device for solving the output problem is available at this time, the theory is the same. A motor output BCI will restore some aspect of voluntary motion (i.e. movement of a computerized mouse cursor; Kennedy et al., 2000; Wessberg et al., 2000; Serruya et al., 2002; Taylor et al., 2002; Paninski et al., 2004; Wu et al., 2003, 2006; Kim et al., 2007, virtual (Kennedy et al., 2004) or robotic hand (Carmena et al., 2003; Velliste et al., 2008), or even computer synthesized speech (this dissertation)) by intercepting motor commands generated by the subject and routing the information to artificial devices designed to restore the specific deficiency. A schematic of the neural prosthesis for speech production used in this dissertation research is shown in Figure 1·1.

Though specific for speech production, this BCI design more generally represents a system for motor output restoration.

1.1 Design hypotheses and assumptions

1.1.1 Speech is a motor task

The decoding methods used for this dissertation all exploit the same principle, namely that speech production, at the level of the motor cortex, is simply a task of continuous motor execution. A number of stimulation and functional imaging studies have reported correlated motor cortex activation during articulator movement and speech production. Most notably, Penfield and Roberts (1959) showed that stimulation of the face and tongue areas of motor cortex can activate or arrest speech articulators.

These findings have been studied extensively through numerical simulation of the Directions into Velocities of Articulators (DIVA; Guenther, 1994, 1995; Guenther et al., 1998, 2006) neurocomputational model for speech production. The DIVA model has incorporated many facial and vocal tract motor cortical regions as the basis for a comprehensive speech motor network (see Figure 1·2) used to simulate human speech production. The model has made theoretical predictions of fMRI BOLD response in the brain during speech production confirmed by subsequent functional neuroimaging studies (e.g. Guenther et al., 2006; Tourville et al., 2008; Ghosh et al., in press). This principle (as embodied by the DIVA model), that speech is a motor task, provides a theoretical framework for incorporating previous neural decoding research of arm and hand kinematics into the paradigm of speech production.

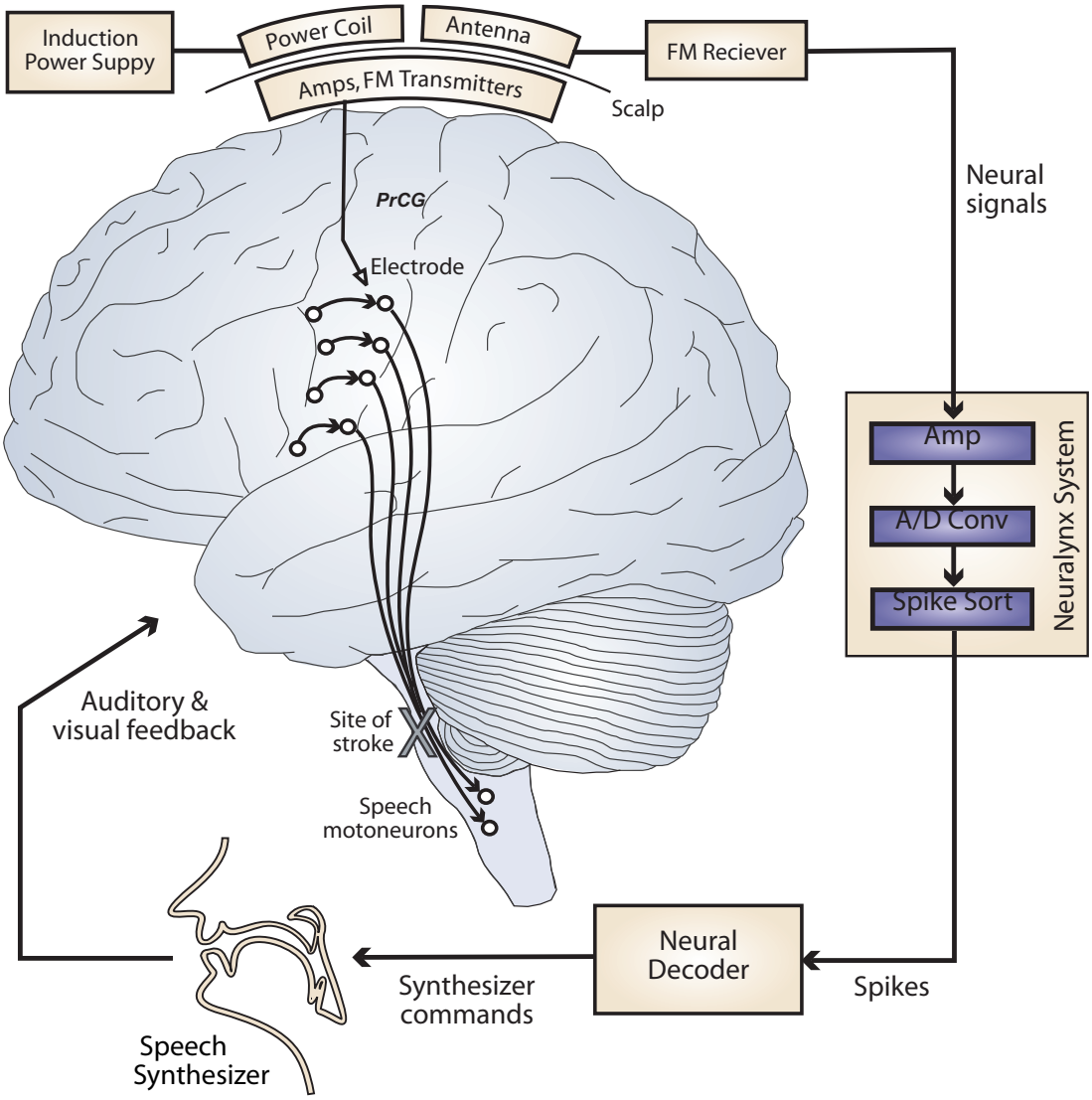


Figure 1·1: Schematic of speech production BCI used in this dissertation. The "X" indicates the location of the brain-stem stroke in subject S1, with the approximate electrode placement indicated on the precentral gyrus.

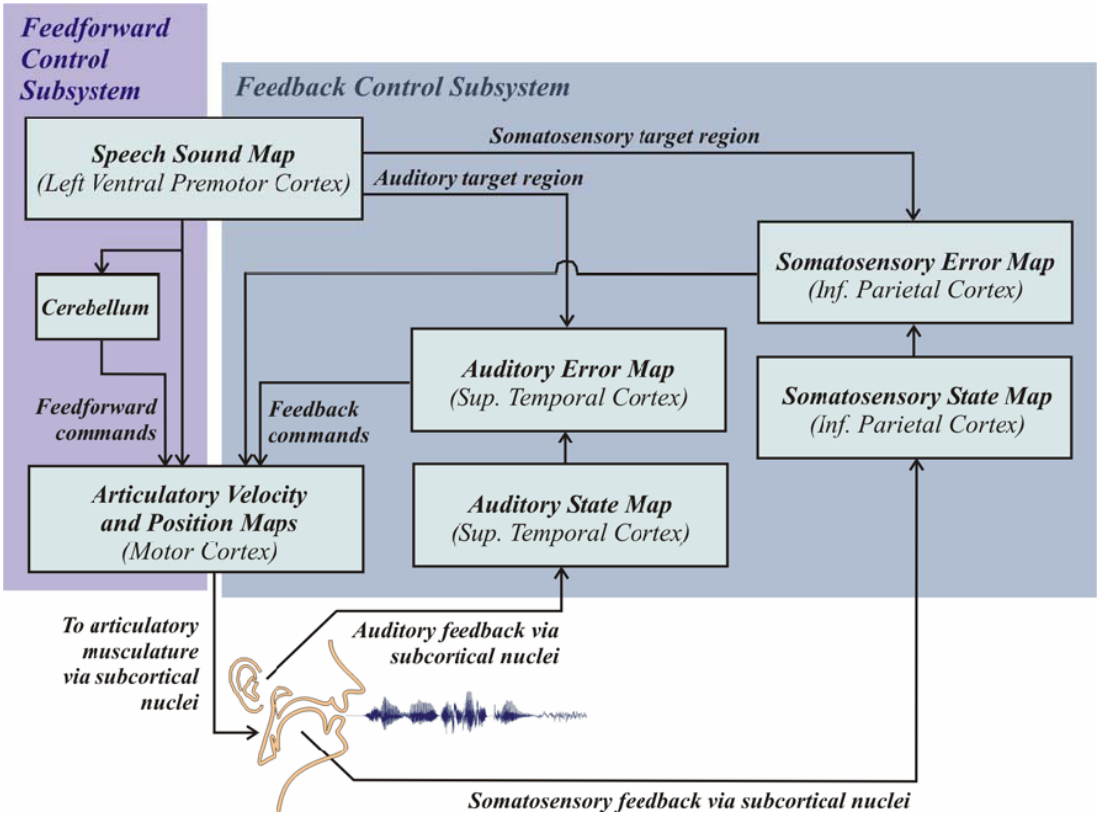


Figure 1·2: Schematic diagram of the DIVA model.

1.1.2 Speech motor cortex activity can represent auditory and articulatory information

The hypothesis that speech motor cortex is capable of equally representing acoustic and articulatory information is supported by two main factors. The first factor is based upon theoretical accounts of speech motor cortex representation spaces. The second exploits relationships between certain acoustic features and movements of the speech articulators.

Factor 1. The term "speech motor cortex" refers to the portions of primary motor and premotor cortex related to the activation of the speech articulators. According to the DIVA model, the form of the neural representation for speech varies depending on area of cortex. Specifically, the model explicitly represents the plan for an upcoming utterance in an acoustic space which informs an execution module responsible for activation of primary motor neurons innervating the vocal tract. Planning is thought to occur in the left ventral premotor cortex while execution in the primary motor cortex (Guenther, 1995; Guenther et al., 1998, 2006). The representations encoded by regions between the left vPMC and MI exist on a continuum ranging from acoustic planning spaces to executive motor command on an anterior-posterior axis. Therefore, depending on the implant location of a brain computer interface for speech production it may be more natural to develop a decoder in acoustic or articulatory domains, or both.

Factor 2. Speech sounds can be represented by resonant frequencies of the vocal tract, called formant frequencies. Put another way, formant frequencies are a low dimensional representation of the speech waveform directly related to vocal tract configuration. The first three formants have been shown to be the most important for accurate speech perception, though only the first two are needed for certain sounds (i.e. vowels). Additionally, the first two formants have the special characteristic of

being highly correlated to movements of the jaw and tongue (Deller et al., 1993; Stevens, 2000). Therefore, if speech utterances attempted with a speech production neural prosthesis are restricted to the vowel sounds, it will be possible to construct decoding models of articulatory movements in terms of acoustic variables without any loss of generalization.

1.2 Electrophysiological study of speech and motor activity

The history of invasive electrophysiology in human is very sparse with the most common opportunity for this type of study coming from pre-operative neurological evaluation (e.g. Horsley, 1886, 1890; Penfield and Roberts, 1959). The Horsley (1886, 1890), and later Penfield and Roberts (1959) studies are representative of typical invasive human neurological investigations primarily involving electrical *stimulation* (rather than passive recording) used to discover critical language and motor neural correlates in order to secure them during surgical procedures. In their seminal work, Penfield and Roberts (1959) reported on a comprehensive study of cortical networks involved in speech and motor tasks as assessed via combinations of electrocorticography (ECoG) and electrical stimulation in awake patients as well as analyses of speech and motor deficits as a result of cortical resections. Of particular interest are the behavioral observations of systematic electrical stimulation of the neocortex used to map the neural correlates of speech. The main results confirmed the observations of Broca and Wernicke, namely that electrical stimulation of the inferior frontal gyrus and posterior superior temporal gyrus, respectively, can interfere with normal speech. In addition, they show that stimulation of other regions in the motor and somatosensory cortices can cause speech arrest and distortion. These observations led to, and are congruent with, modern descriptions of the neural correlates of speech production.

Most passive electrophysiological studies involve non-human animal subjects; therefore, straightforward studies of speech production cannot be performed. An interesting exception involves examination of the similarities between song acquisition in birds and speech acquisition in humans (see Doupe and Kuhl (1999) for a review). Although there are some similarities, it is still unclear whether an appropriate analogy can be made between song and speech as well as between brain structures in songbirds and humans. However, many researchers *have* used microelectrode recording methods to investigate the role of the monkey motor, premotor and prefrontal cortex in standard arm and hand movements, such as reaching (Georgopoulos et al., 1982, 1986; Moran and Schwartz, 1999b; Truccolo et al., 2005; Hatsopoulos et al., 2007), tracing (Moran and Schwartz, 1999a) and action imitation (Kohler et al., 2002). While the preceding studies have all examined the neural activity associated with hand and arm movements, at least one other study examined facial movements. Ferrari et al. (2003) placed electrodes in the facial portion of monkey area F5 (ventral premotor cortex) and neural activity was recorded in response to observed and executed orofacial gestures, such as communicative and ingestive movements.

Much of human neuroscience operates under the assumption of transferability of conclusions based on non-human animal behavior and neural function (primate in particular) to humans. This assumption is questionable in many instances, but it is often the best possible alternative to direct human study. However, this model becomes insufficient when investigating the neural correlates of speech production, limiting speech researchers to indirect methods for assessing speech related neural activity in humans such as functional magnetic resonance imaging (fMRI) and positron emission tomography (PET). Though both methods are undeniably important to the study of speech production and perception in humans, they are wholly ineffective for use in a real-time BCI. On the other hand, microelectrode recordings are quite

reasonable as a basis for real-time decoding of neural signals, in particular for the primate motor system. The hypotheses detailed in Section 1.1.1 provide the conceptual bridge necessary to transfer primate motor electrophysiological methodology to the present case of speech production in humans.

There are many obstacles facing *in vivo* electrode recordings of human subjects. Typical human studies involve short duration microelectrode implantation, on the order of days or months, to avoid the effects of electrode impedance degradation shown to occur in long-duration studies. Further, standard electrodes are known to drift in spatial position over time, causing recorded action potentials to change shape and amplitude as the electrode moves closer to or farther from the neural source, confounding analysis of single unit activity. New electrode designs are available which provide stable recordings of neocortex with unprecedented longevity (Kennedy, 2006). Specifically, the Neurotrophic Electrode implant allows for wireless microelectrode recording (Kennedy, 1989; Kennedy et al., 2000) of up to 16 months in rat (Kennedy, 1989), 15 months in monkey (Kennedy et al., 1992a) and continued for over four years in human (Kennedy, 2006) until patient death. It is precisely this last property which makes the Neurotrophic Electrode uniquely suited for use in understanding the neural correlates of motor behavior and for application in a brain computer interface for speech production. Section 1.3 describes the construction and properties of the Neurotrophic Electrode in greater detail.

The Neurotrophic Electrode, or "cone electrode," has been successfully implanted in five patients all with varying degrees of paralysis and used to restore limited communication ranging from binary responses to mouse cursor movement. Kennedy et al. (2000) previously implanted a brain stem stroke patient, JR, with the Neurotrophic Electrode for over four years. Patient JR was devoid of all voluntary movement with residual facial and toe movement and required a tracheotomy to sustain respiration.

Despite his impairments and loss of speech, JR was alert and cognitively intact. Using the Neurotrophic Electrode implant, JR was able to learn to control a mouse pointer over a virtual keyboard on which he could type (Kennedy et al., 2000) his thoughts as well as learn to flex the digits of a cyberhand (Kennedy et al., 2004). Most recently, a "locked-in" brain-stem stroke patient ER, who will be referred to as S1, was implanted with the Neurotrophic Electrode (Kennedy et al., 2006) with the primary goal of decoding the neural activity related to speech production and providing an alternative means for S1 to communicate.

1.3 Subject and protocol

The same single subject, data acquisition and experimental protocol were used for all portions of this dissertation research.

1.3.1 Subject

A single 24 year old locked-in subject, S1, volunteered for implantation of the Neurotrophic Electrode device and participation in a study of human speech production with the ultimate goal of restoration of his own vocal communication. S1 is nearly completely paralyzed as a result of a brain stem stroke suffered during an automobile accident, though he maintains neurologically normal cognitive and sensory faculties. Previous Neurotrophic Electrode implant subjects ranged in their degree and source[2] of paralysis, though all were considered locked-in. Like many locked-in patients S1 is unable to make most voluntary movements, but he retains rudimentary communication in the form of slow eye movements used to indicate an affirmative or negative response. Since his implantation, S1 has participated in a number of experiments to characterize cortical activity in response to various motor tasks such as controlled

[2]two previous subjects were late stage ALS patients while two others of brain stem stroke

eye movements and imagined face, jaw and tongue movements (Bartels et al., 2007; Kennedy et al., submitted) as well as imagined speech production (Brumberg et al., 2007; Miller et al., 2007; Wright et al., 2007).

Preoperative functional imaging (fMRI) investigations of common speech production tasks (i.e. picture naming, word repetition, etc.) were used to activate and study the speech production network in S1 and to identify a putative set of suitable electrode implantation sites. According to recent fMRI studies, a standard, overt speech production network has been found in humans including the primary motor, premotor and supplementary motor cortex, among other cortical and sub-cortical areas (Guenther et al., 2006; Bohland and Guenther, 2006). Figure 1·3 indicates the regions activated during the preoperative fMRI investigation with the activated areas intersecting the overt speech production network. Specifically, the active area on the precentral gyrus yielded the strongest response to the speech production tasks and was subsequently chosen as the implantation site. S1 was implanted with the Neurotrophic Electrode device in December of 2004 in this region on the crown of the left precentral gyrus near the premotor-primary motor cortex boundary, and the electrode is still viable three years post-implantation (Kennedy et al., 2006). Postoperative X-ray and CT scan (Figure 1·3) confirm the implant site on the precentral gyrus of the left hemisphere.

1.3.2 Microelectrode implant

The implant, shown in Figure 1·4, consists of a single Neurotrophic Electrode with two or three recording wires, a reference wire and supporting subcutaneous hardware for transcutaneous wireless transmission of implant device power and recorded voltage potentials (Kennedy et al., in press). The Neurotrophic Electrode was designed specifically for long-term recordings in mammalian cortex. The recording wires are

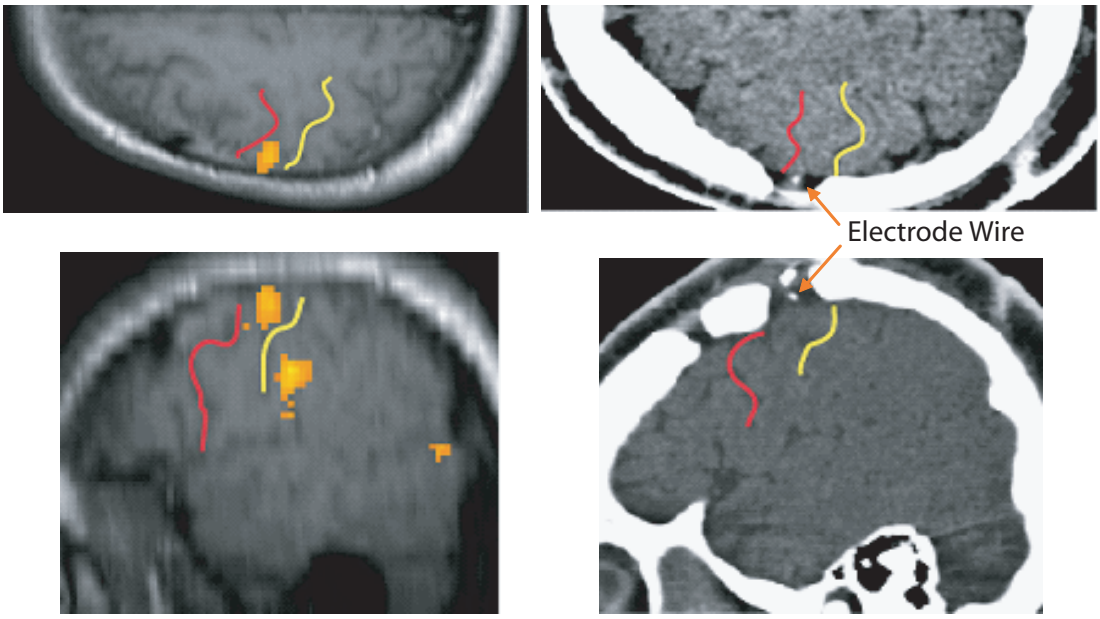


Figure 1·3: fMRI and CT images of subject S1. Left panels: Coronal (top) and sagittal (bottom) slices showing brain activity during a word generation fMRI task by the volunteer prior to implantation. Red lines (left) denote pre-central sulcus; yellow (right) denotes post-central sulcus. Right panels: corresponding images from a post-implant CT scan showing location of electrode.

attached to the inside of a 1-2mm long glass cone (Kennedy, 2006; Kennedy et al., in press) and is inserted obliquely into the cortex settling the cone tip near the pyramidal motor neuron layer. The cone is filled with a neurotrophic factor (Kennedy et al., 1992b) promoting neurite growth from nearby neurons (as far as $600\mu m$ or more) through the length of the cone. Histological analysis of rats and monkeys have indicated the presence of mostly myelinated axons rather than unmyelinated axons and dendrites (Kennedy et al., 1992b) yielding electrical recordings of action potentials. Other supporting neural tissue has been observed within the glass cone, suggesting the neurotrophic factor promoted new growth into the cone and there is evidence of neurite growth from upper cortical layers (i.e. layer 2) through the wide end of the cone. In addition, the analysis indicated axons that enter the cone tip tend to stream out the other end, effectively anchoring the electrode in place within the cortex and eliminating electrode drift.

1.3.3 Data Acquisition

Extracellular voltage potentials of axonal origin are recorded by a two-channel Neurotrophic Electrode. These potentials are very small, on the order of ten to fifty microvolts, and require amplification. Each recording channel is 100x gain amplified by subcutaneous hardware before being transcutaneously transmitted by wireless radio. Once received, the potentials are again 100x gain amplified. The potentials on each channel are then sampled by a Neuralynx, Inc. Cheetah data acquisition system at 30303 Hz where the signals are duplicated with one set of signals (RAW) lowpass filtered at 9000 Hz and the other (FILT) filtered between 300 Hz and 6000 Hz. Both data sets are then recorded onto digital tape and sent to the Cheetah acquisition software where they are committed to file.

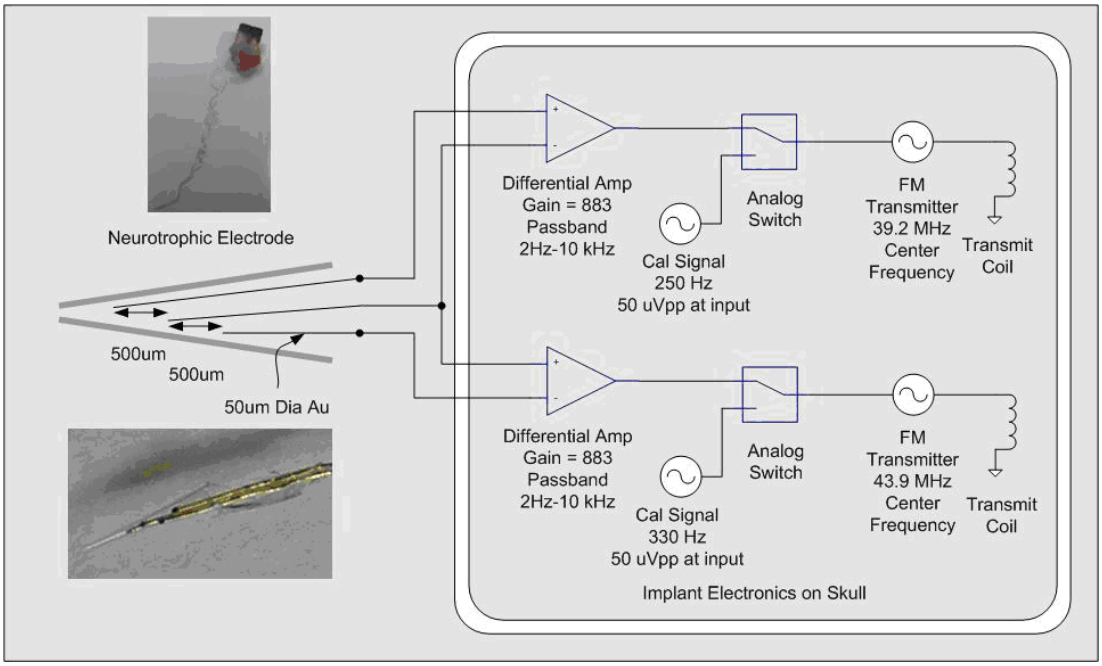


Figure 1·4: Photos of electrode tip and coils, with functional diagram of telemetry system on right. The wire endings appear as black dots inside the glass cone at the end of the gold wires (photo bottom left). The pointed tip is about 50 microns in diameter and is 500 microns from the end of the first wire that appears as a black dot. The photo on the left upper part of the figure is a magnified view of the electrode showing the coiling of the wire and connector.

1.3.4 Experimental paradigm

The ideal neural decoding paradigm involves recording both neural signals and overt external behavior by humans and non-human animals, then prediction of new behavior from neural activity and comparison against the observations. For example, Brown et al. (1998) recorded both firing rates of single units from rat hippocampus *and* spatial location of a freely foraging rat and Truccolo et al. (2005) obtained neural recordings of monkey motor cortex and wrist, hand and arm positions and velocities during grasping tasks. In both these cases it is straightforward to construct a decoder for predicting new behavior given observed neural activity and overt behavior. However, for this dissertation research, analysis of neural activity with respect to observed behavior was impossible due to subject S1's locked-in condition. In this way S1's natural state can be described as open-loop. That is, though he was able to understand instructions and internally (i.e. mentally) perform tasks, there was no way to measure his intended response. In these terms, the goal of this dissertation is to develop a brain computer interface that can "close the loop." Unfortunately, this requires training the neural decoder using unknown intended behavior. A natural assumption, chosen for analysis in this research, was to use the training stimulus as the idealized, unknown state.

Two main experimental paradigms were employed for the research study in this dissertation. The first was an open-loop speech production task used for initial analysis of the neural signals in an offline manner and to train a neural decoder for speech. The second was a closed-loop task aimed to evaluate the trained neural decoder in a real-time production environment. The former is briefly described below and discussed in more detail in Chapter 5 while the latter is detailed in Chapter 6. S1 was administered a 200 mg dose of Provigil before every recording session. The desired

effect was to ensure he was completely alert during the session and to establish an experimentally controlled baseline neural activity across recording sessions. All procedures were approved by the US Food and Drug Administration, Neural Signals, Inc., Gwinnett Medical Center and Boston University Internal Review Boards.

Two offline experimental tasks were employed in this research. The first (REPETITION task) was designed for preliminary offline analysis of neural activity in response to intended speech production. In this task S1 was required to listen to single vowel sounds in isolation, then to repeat the sound just as he heard it. The second task was designed specifically to train the neural prosthesis for real-time control of a speech synthesizer. The open-loop training task required S1 to speak in head continuously along with an auditory stimulus (SPEAK-ALONG). The neural signals and auditory stimulus were used to train a neural decoder (Chapter 5) with the stimulus taken as S1's unknown attempted production. The trained decoder was then applied in real-time to the closed-loop scenario described in Chapter 6.

1.4 Organization

The remainder of this dissertation will detail the steps taken to transform the raw voltage waveform recordings into speech with particular emphasis on application in a real-time environment. First, a review of speech synthesis technology is given in Chapter 2 with performance guidelines for use in a neural speech prosthesis. In Chapter 3, techniques for isolation of single units from the multiunit signal will be discussed. Estimation of neural activity from single unit spike trains is studied in Chapter 4. Chapter 5 presents a review of popular methods for neural decoding of motor cortex with results from the open-loop tasks. Chapter 6 describes the implementation of the neural prosthesis as the application of the research outlined in previous chapters.

Decoder and behavioral results from the closed-loop task are presented and used to evaluate decoder and subject performance. The dissertation concludes in Chapter 7 with a summary of results and discusses possible future directions for both the near and long term.

CHAPTER 2

METHODS OF SPEECH SYNTHESIS

2.1 Introduction

This thesis requires the use of a viable speech synthesizer that is readily available and computationally suited for real-time feedback of decoded speech from neural activity. Design of a prosthesis without instantaneous auditory feedback is unacceptable as continuous monitoring of speech output is crucial for learning to produce intelligible speech. To that end, this chapter consists of a review of methods for artificial speech synthesis and the design requirements needed for implementation in a real-time neural prosthesis for speech. The method of choice must be intelligible and natural sounding and must operate with low computational overhead. All three conditions must be met in order to "convince" the subject that synthesized speech from neural signals are truly his own. Significant delays in acoustic feedback have been shown to cause difficulty in speech production in normal subjects (i.e. not locked-in), a problem that would also emerge in the current scenario. Though it is possible to visually display both auditory and articulatory information for decoder feedback, the natural (and most informative) choice of feedback modality for speech is audio.

Scientists have been attempting to reproduce natural sounding human speech through artificial means for well over two centuries. Initial attempts of simulated speech synthesis by von Kempelen and Kratzenstein in the late 18th century were mechanical models utilizing bellows as artificial lungs, reeds as vocal folds and reso-

nance tubes as upper vocal tracts (von Kempelen, 1791; Dudley and Tarnoczy, 1950; Flanagan, 1972). Kratzenstein used a limited set of resonators, one for each sound, for vowel production (Flanagan, 1972) while von Kempelen utilized a single resonator, manipulated by hand, to produce both vowels and consonants (von Kempelen, 1791; Dudley and Tarnoczy, 1950; Flanagan, 1972). The twentieth century saw speech synthesis systems utilizing electrical circuits in place of bellows, reeds and resonators of earlier approaches. In particular, researchers at Bell Labs developed the electrical speech synthesis system, Voder, which was exhibited at the 1939 and 1940 World's Fair and was capable of producing all phoneme types (Dudley et al., 1939; Flanagan, 1972).

Eventually the electrical circuits of the Voder were replaced by digital software with the growing accessibility of modern digital computers. These new systems were all based on a simplification of the biological speech system under the framework of the acoustic theory of speech production (Fant, 1960). Briefly, this theory makes it possible to view speech as "the outcome of the excitation of a linear filter by one or more sound sources" (Klatt, 1987). In this case, the source is broadband, periodic, turbulent airflow generated by the passage of air through the oscillating vocal folds in the larynx. The linear filter is determined by the configuration of the upper vocal tract (i.e. pharyngeal, oral and nasal cavities). The zeros and poles of the linear filter reflect the resonant frequencies of the vocal tract and are known as *formant frequencies* in the acoustic phonetics literature. Changes in vocal tract configuration by movement of the articulators cause formant frequencies to change making the linear filter time-varying. Using the source-filter framework, Fant designed and produced a digital formant speech synthesizer in which intelligible phrases could be produced by manipulation of the formant frequencies and their bandwidths (Fant, 1960).

Modern speech synthesis methodologies continue this work and fall into two main

classes, formant and articulatory synthesizers. The former technique shapes the source impulse by a digital filter defined by formant frequencies and bandwidths. The latter uses explicit models of the (sub)glottal airflow and upper vocal tract to simulate source excitation of the vocal tract transfer function. Two available methods include the Klatt (1980) formant synthesizer and Maeda (1990) articulatory synthesizer. Both types of synthesizers are comparable for vowel production but articulatory synthesizers have the advantage of explicitly accounting for closure during consonant productions that must be approximated using formant synthesizers.

The mechanical and electrical methods of artificial speech replication described above are all examples of continuous speech synthesis. Another main class of synthesis models are concatenative, or discrete, models, including modern text-to-speech systems. This type of synthesis transforms symbolic representations of speech sounds into acoustic patterns by concatenation of saved waveform segments. These methods, though natural sounding, suffer from disruptive glitches at the concatenation point and also require excessive memory to store the necessary information for quality synthesis. The electrical continuous methods described above suffer in terms of naturalness compared to the concatenative models, but excel in terms of intelligibility. In addition, simple modulations to the source and filter characteristics (i.e. realistic pitch profiles) improve on the naturalness of continuous methods while retaining their intelligibility. Most importantly, there is no evidence for discrete representations of speech sounds in the ventral motor cortex where the Neurotrophic Electrode is implanted. Concatenative speech synthesis makes more sense within a discrete lexical (i.e. phoneme) decoding paradigm. As such, discrete models are not considered in detail in this dissertation research. The two examples of continuous synthesis, formant (e.g. Klatt) and articulatory (e.g. Maeda), are described below.

2.2 Contemporary synthesis methods

2.2.1 Formants

The Klatt (1980) synthesizer is a well known formant synthesizer capable of producing natural sounding, intelligible speech. It allows for complete specification of speech sound characteristics through definition of 63 parameters; however, it is simply not possible, nor it is necessary, for S1 to control every parameter using the neural activity on a single Neurotrophic Electrode. This would require precise control of all 63 parameters using only 56 identified single units. It is much more reasonable to control a two to four degree of freedom device as demonstrated by recent monkey and human motor prosthesis studies (Kennedy et al., 2000; Wessberg et al., 2000; Serruya et al., 2002; Taylor et al., 2002; Carmena et al., 2003; Paninski et al., 2004; Hochberg et al., 2006; Wu et al., 2006; Kim et al., 2007; Velliste et al., 2008). Therefore, speech sounds can be synthesized according to the decoded neural signals in a simplified two-dimensional formant frequency space; that is, the user need only control the first (F1) and second (F2) formant frequencies. This system should allow for reasonable production of all vowel sounds while keeping the control problem relatively simple including creation of natural-sounding vowel sounds, along with a few other phoneme types (glides, liquids, and diphthongs). All of these sounds are produced with an open vocal tract, in contrast with many consonants. In order to produce consonants, the vocal tract must be completely or nearly completely obstructed by the tongue or lips; though Klatt (1980) details the procedure required to produce all consonant types using a formant synthesizer with changes to parameters in addition to F1 and F2. Stop consonants, for example, involve complete elimination of sound energy from the vocal tract at the time of closure, a condition that is not accounted for in a speech synthesis framework in which only the first two formant frequencies are specified while

the sound output level is left constant.

Interestingly, simulations of the DIVA neurocomputational model of speech production (e.g., Guenther et al., 1998), showed that the model can learn to produce intelligible speech involving both consonants and vowels using a Klatt synthesizer while varying only eight parameters (three for formants F1, F2 and F3, one for voicing AV, three for formant amplitude A1, A2 and A3 and one for frication amplitude AF). Though in this case eight parameters are specified, the approach requires the user to directly control only *three* continuous dimensions (F1, F2, and F3) and one discrete dimension (AV)[1] in order to produce a wide range of speech sounds, while the amplitude parameters can be automatically controlled by the synthesizer software. Evidence of such simplified control comes from simulations of early versions of the DIVA model utilizing a realistic 2D vocal tract model to estimate values for A1, A2, A3, and AF near closure.

Acoustic analyses of speech signals (e.g., Stevens, 2000) indicate that F1 varies systematically with the degree of closure of the vocal tract at the tightest constriction above the larynx (typically a tongue or lip constriction), with very low F1 values (below 200 Hz) occurring at the point of closure for stop consonants, and relatively high F1 values occurring for vowels involving a wide open vocal tract. Therefore, it is possible to make use of this relationship for creation of a natural decrease in sound amplitude when going into a vocal tract closure, allowing the user to produce a stop closure simply by decreasing the value of F1. This mimics the sound amplitude decrease that occurs when F1 decreases during closure in a real vocal tract without requiring the user to explicitly control the amplitude decrease. Furthermore, the amplitude of frication can be related to F1 as well permitting user control of fricative consonants. However, the current system, described in Section 2.3, utilizes

[1] A total of only four user controllable parameters.

constant amplitude restricting production to vowel sounds only, though manipulation of amplitude based on formant frequencies is being considered for future study.

2.2.2 Articulators

The use of an articulatory synthesizer, for instance the Maeda (1990) synthesizer, is particularly logical given the location of the electrodes in regions of motor cortex coding speech articulation. The synthesizer takes as input voicing parameters and seven articulatory parameters: jaw height, larynx height, lip aperture, lip protrusion, tongue body height, tongue body position, and tongue tip position. The output of such a synthesizer is the acoustic signal resulting from the articulator configuration and voicing source. Unlike formant synthesizers, articulatory synthesizers require no special relationships for production of all types of consonants (i.e. stops and fricatives). Therefore, control of an articulatory synthesizer is desired for the implementation of a BCI for speech restoration capable of producing all sounds necessary for fluent speech.

The Maeda synthesizer has been shown in computer simulations to be capable of intelligible, reasonably natural-sounding synthetic speech. Furthermore, simulations of the DIVA model have verified that intelligible speech can be maintained despite a reduction from seven articulatory parameters to as few as three. Such a reduction in user-controlled parameters is necessary for S1 given the presence of only one Neurotrophic Electrode. These parameters can be derived by first transforming acoustic speech signals from an existing database, such as the Peterson and Barney (1952) database, into articulator positions of the Maeda synthesizer via simulations using the DIVA model. Afterward a principal components analysis (PCA) on the resulting articulations should reveal the three most useful articulatory movement dimensions (likely to consist of synergies involving more than one articulator) for reproducing

speech signals. Control of an articulatory synthesizer by the method described above is not yet available for the neural prosthesis but will be pursued by future study.

2.3 Choice of synthesis for speech prosthesis

Both the Klatt and Maeda synthesizers are applicable for usage in the speech prosthesis design. As discussed above, both methodologies are capable of production of natural, intelligible speech using only 2-3 parameters making the task of controlling speech a two or three dimensional problem, which already has shown possible in other human and nonhuman brain computer interface studies (Kennedy et al., 2000; Wessberg et al., 2000; Taylor et al., 2002; Serruya et al., 2002; Paninski et al., 2004; Hochberg et al., 2006; Wu et al., 2006; Kim et al., 2007; Velliste et al., 2008). In addition, computer programs for both algorithms are readily available, in fact both have been utilized as speech synthesis modules during previous simulation studies involving the DIVA model by Guenther and colleagues.

The last requirement, that the methods be computationally efficient is where these two methods diverge. Benchmark tests showed that synthesis of a 10 ms speech waveform using the Klatt synthesizer required less than one millisecond of computational time. However, the Maeda synthesizer requires 100 ms to process the same speech waveform. Though not an extremely large computational overhead, the processing time of the Maeda method would introduce delays in acoustic feedback likely to disrupt speech production attempts. Therefore, the Klatt method for formant synthesis was chosen for the current speech prosthesis design. Tables 2.1 and 2.2 provided a detailed report of all parameters and their values defined for the Klatt synthesizer differentiating between static and user controlled parameters. Efforts are still ongoing to reduce computational time needed by the Maeda method as articulatory synthe-

sis is likely to outperform formant synthesis in production of the full range of fluent speech.

N	Symbol	Description	Value
1	DU	Duration of the utterance (ms)	10
2	UI	Update interval for parameter reset (ms)	1
3	SR	Output sampling rate (Hz)	22050
4	NF	Number of formants in cascade branch	2
5	SS	Source switch (1=impulse, 2=natural, 3=LF model)	1
6	RS	Random seed (initial value of random # generator)	8
7	SB	Same noise burst, reset RS if AF=AH=0, 0=no,1=yes	1
8	CP	0=Cascade, 1=Parallel tract excitation by AV	0
9	OS	Output selector (0=normal,1=voicing source,...)	0
10	GV	Overall gain scale factor for AV (dB)	60
11	GH	Overall gain scale factor for AH (dB)	60
12	GF	Overall gain scale factor for AF (dB)	60
13	GI	Overall gain scale factor for AI (dB)	60

Table 2.1: Summary of Klatt synthesizer speaker definition parameters used in the neural speech prosthesis.

N	Symbol	Description	Value
14	F0	Fundamental frequency (dHz)	1000
15	AV	Amplitude of voicing (dB)	60
16	OQ	Open quotient (voicing open-time/period), in %	65
17	SQ	Speed quotient (rise/fall time, LF model), in %	200
18	TL	Extra tilt of voicing spectrum, dB down @ 3 kHz	0
19	FL	Flutter (random fluct in f0), in % of maximum	0
20	DI	Diplophonia (alt periods closer), in % of max	0
21	AH	Amplitude of aspiration (dB)	0
22	AF	Amplitude of frication (dB)	0
*23	F1	Frequency of 1st formant (Hz)	400 (default)
24	B1	Bandwidth of 1st formant (Hz)	60
25	DF1	Change in F1 during open portion of period (Hz)	0
26	DB1	Change in B1 during open portion of period (Hz)	0
*27	F2	Frequency of 2nd formant (Hz)	1800 (default)
28	B2	Bandwidth of 2nd formant (Hz)	90
29	F3	Frequency of 3rd formant (Hz)	2500
30	B3	Bandwidth of 3rd formant (Hz)	150
31	F4	Frequency of 4th formant (Hz)	3250
32	B4	Bandwidth of 4th formant (Hz)	200
33	F5	Frequency of 5th formant (Hz)	3700
34	B5	Bandwidth of 5th formant (Hz)	200
35	F6	Frequency of 6th formant (Hz) (applies if NF=6)	4990
36	B6	Bandwidth of 6th formant (Hz) (applies if NF=6)	500
37	FNP	Frequency of nasal pole (Hz)	280
38	BNP	Bandwidth of nasal pole (Hz)	90
39	FNZ	Frequency of nasal zero (Hz)	280
40	BNZ	Bandwidth of nasal zero (Hz)	90
41	FTP	Frequency of tracheal pole (Hz)	2150
42	BTP	Bandwidth of tracheal pole (Hz)	180
43	FTZ	Frequency of tracheal zero (Hz)	2150
44	BTZ	Bandwidth of tracheal zero (Hz)	180
45	A2F	Amp of fric-excited parallel 2nd formant (dB)	0
46	A3F	Amp of fric-excited parallel 3rd formant (dB)	0
47	A4F	Amp of fric-excited parallel 4th formant (dB)	0
48	A5F	Amp of fric-excited parallel 5th formant (dB)	0
49	A6F	Amp of fric-excited parallel 6th formant (dB)	0
50	AB	Amp of fric-excited parallel bypass path (dB)	0
51	B2F	Bw of fric-excited parallel 2nd formant (Hz)	250
52	B3F	Bw of fric-excited parallel 3rd formant (Hz)	300
53	B4F	Bw of fric-excited parallel 4th formant (Hz)	320
54	B5F	Bw of fric-excited parallel 5th formant (Hz)	360
55	B6F	Bw of fric-excited parallel 6th formant (Hz)	1500
56	ANV	Amp of voice-excited parallel nasal form. (dB)	0
57	A1V	Amp of voice-excited parallel 1st formant (dB)	60
58	A2V	Amp of voice-excited parallel 2nd formant (dB)	60
59	A3V	Amp of voice-excited parallel 3rd formant (dB)	60
60	A4V	Amp of voice-excited parallel 4th formant (dB)	60
61	ATV	Amp of voice-excited par tracheal formant (dB)	0
62	AI	Amp of impulse (dB)	0
63	FSF	Formant Spacing Filter (1=on, 0=off)	0

Table 2.2: Summary of Klatt synthesizer parameters used in the neural speech prosthesis. * parameters dynamically updated by neural decoder.

CHAPTER 3

IDENTIFICATION OF NEURAL UNITS

3.1 Introduction

The major component of this dissertation is the statistical analysis and prediction of intended behavior from extracellular microelectrode recordings. The raw signal obtained from the electrode is a continuous waveform representing the extracellular voltage potential resulting from current flows down the length of the many axons "captured" by the Neurotrophic Electrode. In their raw form, these potentials carry information about the continuously sampled electrical state of the recorded axons over time; however, subsequent analysis is tailored to individual units, or ensembles of individual units. Therefore, the raw signal must be broken down and processed to identify putative action potentials, isolate real from false spikes, classify similar real spikes into clusters and record their arrival times. This process is termed *spike sorting*.

Utilizing the experimental setup described in Section 1.3, action potential voltage waveforms are recorded using the two electrode wires and sampled to isolate single unit activity (SUA), multiunit activity (MUA) and the local field potential (LFP). The SUA and MUA appear within the 300 Hz to 6000 Hz frequency range and the SUA is clustered using standard spike sorting practices (described below) while the MUA is taken as an estimate of the spiking activity in a small region around the electrode tip (Stark and Abeles, 2007). The LFP contains slower oscillations appearing below

300 Hz and represents the global activity from a somewhat larger area around the electrode tip (Mitzdorf, 1985; Stark and Abeles, 2007). Both the LFP and MUA require simple discrete-time signal processing to be isolated and have been shown to provide information about cognitive state. The information conveyed by LFP, MUA and SUA is redundant, but there exist specialized techniques for extracting key features from each signal type not available in the others (i.e. spectral analysis of LFP and rate coding/decoding of SUA). However, the LFP and MUA are beyond the scope of this chapter and will not be discussed further in this dissertation. To assess SUA obtained from an extracellular microelectrode one must correctly identify occurrences of action potentials (detection) and accurately assign them to specific classes (classification).

3.2 Detection

Detection of single unit action potentials begins with identification of a putative spike by bandpass filtering raw voltage potentials. The passband frequency range is crucial for isolation of single units, and is defined by a 300 Hz passband, to remove the local field potential (LFP) and a 6000 Hz stop-band, to eliminate high-frequency noise. Typically, the experimenter manually obtains a voltage threshold designed to discriminate between real action potential events and background noise. The detection itself depends on a few design considerations such as signal transformations and type of thresholds. Many signal transformations exist and are listed in Table 3.1. By far the most common approach is using the positive amplitude voltage crossing (Lewicki, 1998; Obeid and Wolf, 2004). The simple threshold detectors (e.g. positive, negative and absolute value) have the advantage that they are extremely easy to implement, but they fail in low signal-to-noise ratio (SNR) scenarios. Conversely, more com-

plex energy-based detectors (e.g. NEO and running RMS) are more computationally intensive but actively emphasize spikiness and attenuate noise (Mukhopadhyay and Ray, 1998; Kim and Kim, 2000; Csicsvari et al., 1998; Rutishauser et al., 2006).

The manner in which the threshold value is calculated can be manipulated as well. The threshold described in the previous example is manually fixed at a value determined by the experimenter based on expert knowledge. A class of alternative methods are data-driven (i.e. standard deviations, Quiroga et al., 2004) where the optimal threshold is found from the data but is held fixed. Similarly, adaptive thresholds are data-driven but are allowed to change over time to eliminate effects due to local changes in background noise and to extend processing capabilities to non-stationary data (Obeid and Wolf, 2004; Rutishauser et al., 2006; Tan et al., in press). The clear advantage of the non-adaptive methods is their ease of computation (i.e. thresholds are calculated only once) while the adaptive methods must be continuously updated. The benefits of adaptive thresholds are equally clear and must be weighed against their potential costs.

Method	Description
The positive part of the signal, x	$[x]^+$
The negative part of x	$[x]^-$
The absolute value of x	$\texttt{abs}[x], \lvert x \rvert$
The squared signal	x^2
Nonlinear energy operator	$\texttt{NEO}_\delta[x] = \psi\{x[n]\} = x^2[n] - x[n+\delta]x[n-\delta]$
Running RMS	$\texttt{rRMS}[n] = \left[\frac{1}{M}\sum_{m=1}^{M}\left(x[n-m] - \bar{x}[n]\right)^2\right]^{\frac{1}{2}}$ where $\bar{x}$ is a running average

Table 3.1: Signal transformations used for spike detection via threshold crossings (Obeid and Wolf, 2004; Mukhopadhyay and Ray, 1998; Kim and Kim, 2000; Bankman et al., 1993; Rutishauser et al., 2006).

Once a spike has been located, some time window of samples (usually 1-3ms) around the spike threshold crossing are stored and the waveforms are systematically aligned at a specific point. For example, the waveforms can be aligned to the threshold detection point, positive peak, negative peak or some combination (i.e. alignment to the peak or valley with the maximum absolute amplitude; Rutishauser et al., 2006). Fee et al. (1996) noted while spikes may be properly aligned to their maximum peak, the true maximum is often hidden between two successive samples due to insufficient sampling rate. Therefore, they propose to upsample the raw spike waveform, realign to the true peak (or valley), then resample. Another source of error was identified by Rutishauser et al. (2006), namely that alignment to the "incorrect" peak artificially creates additional erroneous spike clusters during classification. Their solution is also to upsample the spike waveforms, as in Fee et al. (1996), then determine the appropriate alignment location based on peak amplitude 'significance' and temporal order (Rutishauser et al., 2006, Algorithm 3). This section detailed just one method for detection of putative spikes from a multiunit signal, namely threshold-crossing detection. Other methodologies exist, such as those requiring no prior sampling (i.e. no bandpass filtering) and data-driven filters, and will be investigated by future work.

3.3 Classification

Classification refers to the grouping of spike waveforms based on some measure of similarity. This process is more widely discussed than spike detection and is summarized by two excellent reviews (Schmidt, 1984; Lewicki, 1998). Unfortunately, the dimensionality of the spike classification problem can be very large (i.e., dimension = 1-3ms $\times$ sampling rate $\times$ # spikes) and has previously been computationally intractable. For example, in this research each spike waveform was represented by 32

samples (1.05 ms) and over 100,000 putative spikes were evaluated in just a 500 ms interval. In an effort to make the problem more tractable, it is possible to define the spike clusters on the most salient characteristics of the spike waveforms (Schmidt, 1984; Lewicki, 1998). These characteristics or features include the peak amplitude, valley amplitude, location of peak and valley, ratio of peak to valley, waveform energy and principal component scores (Abeles and Goldstein, 1977).[1]

The simplest and most common method for spike waveform classification is manual cluster cutting. This process refers to a supervised method for determining boundaries between discontinuous spike clusters. Recently a number of semi-automated clustering methods for spike class assignment have been developed (Harris et al., 2000; Quiroga et al., 2004; Rutishauser et al., 2006). Automated methods cluster spike waveforms according to their statistical properties, rather than by largely intuitive visual grouping as used by manual expert sorters. Three spike clustering environments are examined in this dissertation: SpikeSort3D (Neuralynx, Inc.), KlustaKwik (Harris et al., 2000) and WaveClus (Quiroga et al., 2004). The first is a manual convex hull cluster cutting technique while both KlustaKwik and WaveClus are partially automated.

A common theme in spike sorting is the determination of "how many clusters." An obvious choice for a cluster quantity constraint is the cluster likelihood.[2] However the Gap statistic (Tibshirani et al., 2001) has been used in non-neurophysiological environments and has achieved good results in noisy situations with overlapping cluster boundaries. An analysis of these constraints is beyond the scope of this research but are of great importance for accurate spike sorting and will be addressed by future study.

[1]Features can also be defined as the entire spike waveform as this simply increases the dimensionality of the feature space.

[2]Employed by the KlustaKwik method.

3.3.1 Manual Cluster-Cutting: SpikeSort3D

Manual cluster-cutting entails direct user interaction for the creation of spike waveform clusters. For example, Lewicki (1998, Figure 6b) shows a plot of spike height (spike maximum - spike minimum) vs. spike width (time between spike maximum and minimum) complete with potential cluster boundaries. Cluster boundaries are computed through the iterative definition of distinct convex hulls.[3] In the context of cluster-cutting, a user defines a convex polygon around regions of putative clusters. The procedure is repeated for as many pairs of features as required to define a set of stable clusters. The cluster-cutting package SpikeSort3D (Neuralynx) was used to manually sort potential spikes into discernible clusters using the convex-hull technique.

The main advantage of manual cluster-cutting comes from the easy and intuitive nature of defining cluster boundaries. A user simply draws a bounding polygon on a computer screen around a visually dense portion of the feature space and records what spikes lay within the region. In addition, these methods can be extended to include all waveform data points in place of, or in addition to, the subset of features already discussed.

However, there seem to be at least two major disadvantages with manual clustering, both of which involve the human element. First, as the number of features increases, so does the number of pairwise inspections of the feature space.[4] Increasing the dimensionality of the feature space can make the time needed for cutting increase dramatically and can introduce additional human error, discussed next. The second disadvantage arises due to large variability inherent in manual spike sorting (Harris

[3]In two dimensional space, the convex hull is the minimal bounding polygon with all internal angles less than 180° and all line segments between polygon vertices remain within the polygon.

[4]The number of pairwise feature space comparisons is equal to $\binom{N}{2}$ where N is the number of features.

et al., 2000; Wood et al., 2004). Harris et al. (2000) compared manual cluster definitions of extracellular recordings to those of a 'master' intracellular recording. They found significant Type I and Type II errors of up to 30%, which can lead to degraded significance in the evaluation of coincident or synchronous firing activity (Pazienti and Grün, 2006). Wood et al. (2004) investigated this phenomenon by determining human spike sorter cluster definition variability for spikes recorded extracellularly for use in a neural prosthesis application. They reported similar results of 23% Type I errors and 30% Type II errors in cluster assignment of synthetic data. They also showed significant variability between sorters for real data; only 25% of identified single units were agreed upon by all sorters (N=5). The results of both studies conclude the need for accurate, stable automatic algorithms for spike classification.

3.3.2 Bayesian Clustering

Harris et al. (2000) developed a semi-automatic spike classification algorithm in response to their conclusion that manual spike sorting was susceptible to human error. Therefore, they used the AutoClass (Cheeseman and Stutz, 1996) algorithm to find the optimal ellipsoidal boundaries between feature-space clusters. However, they found that AutoClass had a tendency to overcluster the data and required user input to join clusters with identical sources. These results led to the development of the KlustaKwik automatic classification system which, again, uses optimal ellipsoidal boundaries, determined by the Expectation-Maximization (EM) algorithm. The EM algorithm is a standard technique for clustering applications in general (e.g. Fraley and Raftery, 2006). In the case of KlustaKwik, the optimal number of clusters is determined by maximizing the Akaike Information Criterion (AIC; Akaike, 1974), or the Bayesian Information Criterion (BIC; Schwartz, 1978). The BIC measure is known to penalize model selection more than AIC as the number of model parameters (i.e.

spike clusters) increases. Application of the BIC penalty to the clustering algorithm tends to yield fewer spike clusters than with the AIC penalty. The main difference between KlustaKwik and AutoClass lies in the ability of KlustaKwik to periodically check for decreases in model fit penalty due to the merging of existing clusters. The original usage of AutoClass (and KlustaKwik) for spike classification (Harris et al., 2000) was based on a feature decomposition of the raw waveforms (i.e. spike height, principal components analysis, etc). However, the method can be easily extended to clusters based on the full spike waveform.

3.3.3 Superparamagnetic Clustering

Quiroga et al. (2004) described a semi-supervised clustering method for spike waveform sorting. The method includes automatic spike detection using a simple data-driven amplitude threshold, wavelet decomposition of the spike waveforms (for dimension reduction) and automatic clustering of the wavelet coefficients using the superparamagnetic clustering technique (SPC; Blatt et al., 1996). The threshold is obtained with respect to the background noise standard deviation defined by $N\sigma$ where σ is a robust estimator of the standard deviation[5] and N is chosen by the experimenter. The wavelet coefficients are analogous to user-defined features and principal component scores mentioned above.

The details of SPC are beyond the scope of this dissertation. Briefly, SPC is an iterative algorithm, like k-means, which continuously updates the likelihood of cluster membership for all data points. A single parameter, temperature, is used to determine the number of clusters and is defined by the user. Low temperature values yield few clusters while high values allow many clusters. The final set of cluster definitions is taken automatically from the superparamagnetic, or middle, temperature range.

[5]The WaveClus data-driven threshold was defined as: $\sigma = \text{median}\left\{\frac{|x|}{0.6745}\right\}$.

The SPC method eliminates many human factors in the spike detection and sorting process, thereby increasing the likelihood of stable cluster definitions over repeated sortings. However, a human user is still necessary to make final judgments of cluster quality. While the method requires minimal human intervention, the manipulation of the temperature parameter is less intuitive than formation of convex hull boundaries described used in manual cluster cutting.

3.4 Results

Spike detection methods were examined through implementation of four signal transformations (absolute value, NEO, smoothed NEO (SNEO) and running RMS) using a data-driven threshold for an objective comparison. The methods were qualitatively evaluated and number of total putative spikes were compared. Quantitative analysis of spike detection performance was not attempted but future research plans will be designed to rigorously test each method for usefulness in the speech BCI.

Spike classification methods were compared against each other to determine the best method for clustering the spike data obtained from the Neurotrophic Electrode implant. A benchmark test was used to qualitatively evaluate each method. It entailed the identification of two main groups of spike waveform shapes: initial (1) positive and (2) negative depolarizations were expected depending on recorded axon geometry relative to the recording and reference wires. Statistical tests of cluster uniqueness were performed for quantitative comparison of each method.

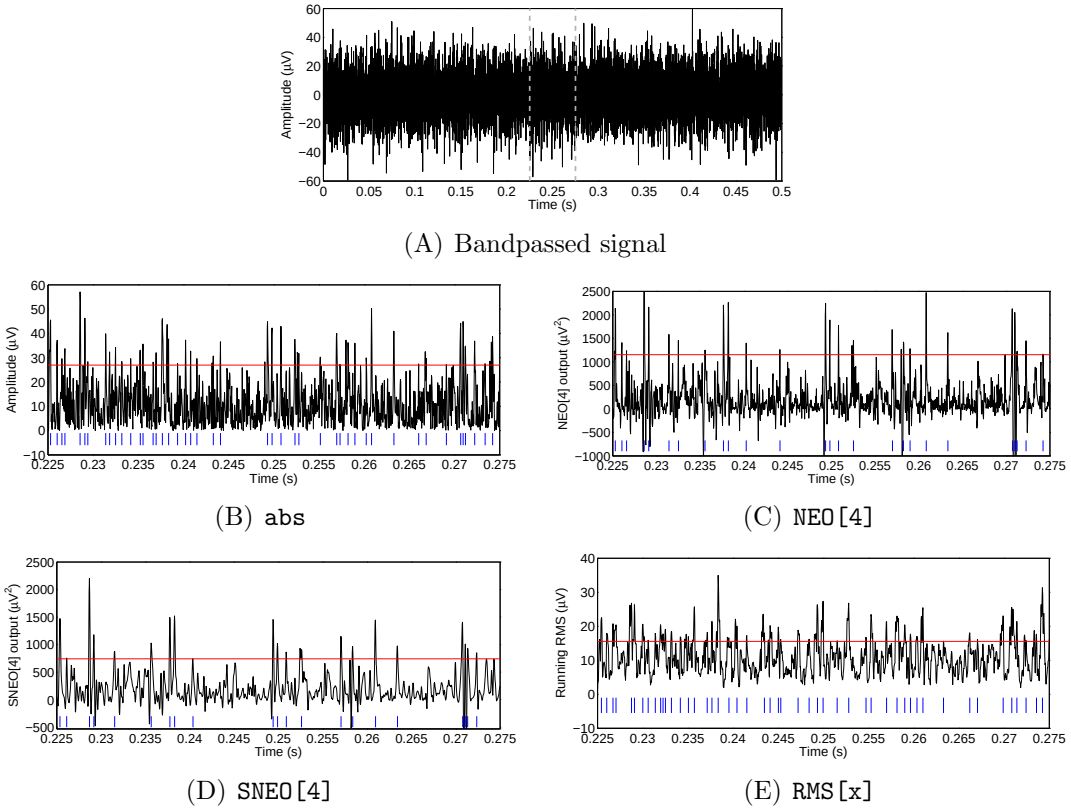
3.4.1 Detection

A 32 second recording was taken using the Neurotrophic Electrode implant, amplified and bandpass filtered according to methods described in Section 1.3.3. The signal was spike preemphasized according to the absolute value, NEO and running RMS

(0.25 ms window) methods for subsequent detection analysis. A smoothed (8-point Bartlett window, 0.264 ms) version of the NEO was also used as previous studies have shown it has desirable properties above the standard NEO (Mukhopadhyay and Ray, 1998; Kim and Kim, 2000). Putative spikes were detected utilizing a data-driven threshold for objective comparison between each method. Both the NEO and SNEO were computed with $\delta = 4$ which, when centered on a spike peak, yields the maximum value as spike widths are typically 0.25 ms in duration, exactly the size of the $\delta = 4$ NEO window. The threshold, T, was chosen as 3σ greater than the mean preemphasized signal ($T = (27.0\mu V, 1152.3\mu V^2, 743.0\mu V^2, 15.6\mu V)$ for ABS, NEO, SNEO and rRMS, respectively). Putative spike waveforms were centered around the 8^{th} sample and assigned to the maximum point of transformed signal within 8 points of the threshold crossing.

Examples of each preemphasis transformation are shown in Figure 3·1. The raw voltage waveform was recorded using the Neurotrophic Electrode, amplified and filtered (a 500 ms sample is shown in Figure 3·1(A)) and the four transformations were applied (Figures 3·1(B)-(E)). Threshold crossings were identified and 32-sample spike waveforms extracted from the original bandpassed recordings. Spike occurrences are shown as rasters in Figures 3·1(B)-(E).

The example results in Figure 3·1 graphically indicate the spike-shape amplification (and non-spike attenuation) feature of the NEO and SNEO preemphasis techniques. Putative spikes found using the NEO and SNEO transformation agreed with those found through simple absolute-value transformation but were more selective. That is, spikes found with NEO and SNEO were also found with ABS but ambiguous spikes found with ABS were rejected by NEO and SNEO. The running RMS method was not as conservative as the NEO methods and appears more similar to ABS though it too achieves some non-spike attenuation not captured by ABS. The



(A) Bandpassed signal

(B) `abs`

(C) `NEO[4]`

(D) `SNEO[4]`

(E) `RMS[x]`

Figure 3·1: Examples of signal transformations from Table 3.1. (A) A 500 ms bandpassed sample recorded from subject S1. (B)–(E) Portion of signal between dashed gray lines in (A) preemphasized according to the absolute value, NEO[4], SNEO[4] and running RMS transformations. Red lines indicate the 3σ threshold level and putative spikes are shown as blue rasters on bottom of each figure.

number of putative spikes determined by threshold crossing of the four preemphasis methods is summarized in Table 3.2. These results confirm the observation that the NEO and SNEO methods are more conservative in their estimate of putative spikes.

Method	Threshold (3σ)	# spikes
ABS	$27.0\mu V$	31118
NEO[4]	$1152.3\mu V^2$	13997
SNEO[4]	$743.0\mu V^2$	10081
rRMS	$15.6\mu V$	35110

Table 3.2: Putative spike summary using four preemphasis transformations: ABS, NEO[4], SNEO[4] and rRMS.

3.4.2 Supervised cluster-cutting

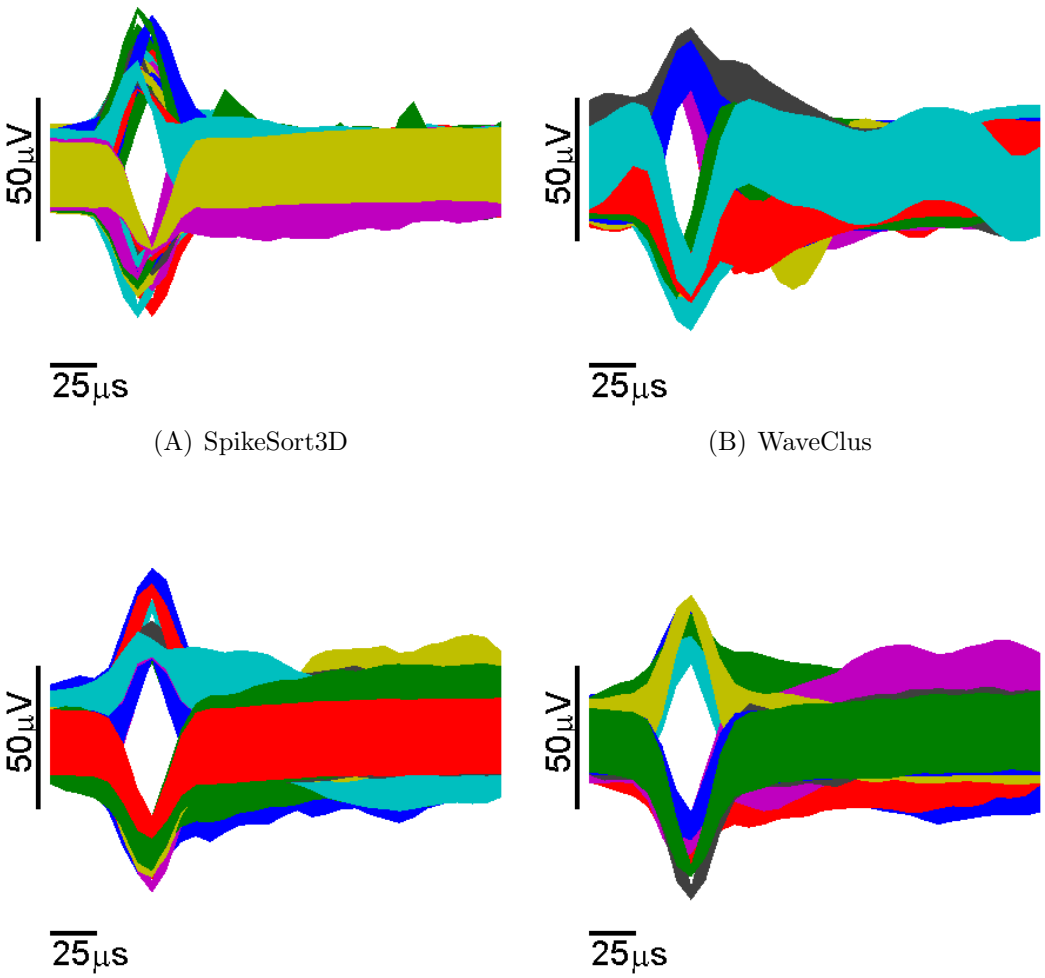
The Neuralynx data acquisition and spike sorting software has built-in functionality for automatic spike detection via voltage threshold crossings and spike sorting through manual definition of spike cluster boundaries. A threshold was manually chosen based on expert analysis and set at 10 μV for detection of putative spikes for the data obtained in this dissertation research. The spike waveforms were clustered by an expert along the following dimensions: peak amplitude, valley amplitude, spike height and 8^{th} sample amplitude. The 8^{th} sample amplitude is important as it was used as the alignment point for spike detection. Interspike interval histograms and autocorrelograms were used to describe the firing characteristics of each spike cluster. Cluster boundaries were iteratively refined over a long interval (i.e. months) to obtain the minimal boundary such that the units within remained uncorrelated as determined by a cross-correlation analysis. Classification according to this convex hull technique yielded 29 spike clusters from the first recording wire of the Neurotrophic Electrode and 27 spike clusters from the second (Kennedy et al., in submission).

Mean waveforms from each cluster region on the first recording wire are shown

in Figure 3·2(A). Each cluster mean waveform is assigned a different color with the width of each waveform equal to one standard deviation around the mean. The 8th sample amplitude - spike waveform energy 2D feature plane is illustrated in Figure 3·3(A). Classified spike waveforms are represented by the points plotted in the figure and spike clusters are differentiated by color. Immediately observable is the separation between two broad clusters of spikes indicated by the left and right clusters of points. As the voltage amplitude of the 8th sample is plotted on the abscissa, these two groups of spikes represent those with negative (left) and positive (right) depolarizations. Further cluster boundary definitions within both of these groups describe subtle differences between spikes of the same depolarization. The inter-spike interval distributions and auto-correlations for the clusters found by the manual cluster-cutting method have remained stable for over three years in subject S1, a feat not matched by any other *in vivo* microelectrode (Kennedy, 2006). Such signal stability and longevity is a requirement for any long-term human neural prosthesis design.

3.4.3 Unsupervised cluster-cutting

For the remaining two methods, KlustaKwik and WaveClus, the initial spike detection from the Neuralynx system was used to maintain consistency between all three techniques, and sorting was restricted to putative spikes found on the first Neurotrophic Electrode recording wire for simplicity of analysis. Spikes sorted using the KlustaKwik system used the spike energy, height and 8th sample amplitude as discrimination features, while wavelet decomposition coefficients were used for spikes sorted by WaveClus. In addition, KlustaKwik was applied twice, once with the AIC penalty and once with the BIC penalty. The mean spike waveforms for each cluster are illustrated in Figure 3·2 for all methods, and examples of the feature spaces used



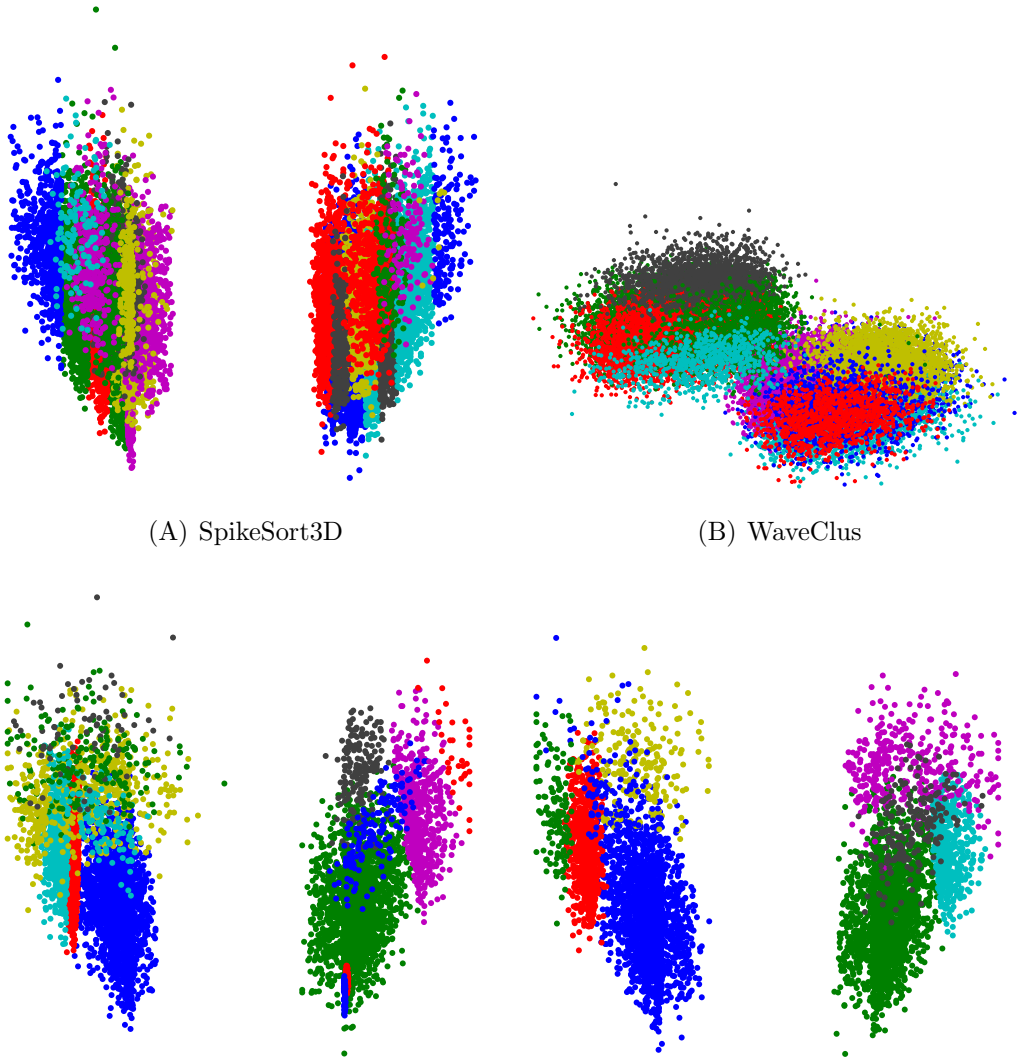
(A) SpikeSort3D

(B) WaveClus

(C) KlustaKwik (AIC)

(D) KlustaKwik (BIC)

Figure 3·2: Mean ($\pm\sigma$) spike waveforms for each spike sorting method. (A) SpikeSort3D, N=29 clusters, (B) WaveClus, N=11 clusters, (C-D) KlustaKwik with (C) AIC penalty, N=17 clusters and (D) KlustaKwik with BIC penalty, N=9 clusters. Mean waveforms from each cluster are represented by different colors and the width of each waveform equals one standard deviation from the mean.



(A) SpikeSort3D

(B) WaveClus

(C) KlustaKwik (AIC)

(D) KlustaKwik (BIC)

Figure 3·3: Example spike feature spaces used for sorting. The feature space shown for manual sorting (via SpikeSort3D) and both AIC and BIC penalized KlustaKwik are the 8^{th} sample amplitude - spike waveform energy plane (A,C and D) while the space defined by the second and third wavelet coefficients is shown for WaveClus (B). Colors indicate spike cluster.

for each method are shown in Figure 3·3. Both unsupervised methods found the two broad categories of positive and negative depolarized action potentials which is consistent with the results found by manual sorting.

Upon visual inspection it appeared the KlustaKwik method often formed spike clusters around seemingly "noise" spikes. That is, under the statistical optimality criteria used to create, split and merge clusters some putative spikes that were most likely not biological in origin were grouped together; therefore, it was necessary to manually prune those clusters found by KlustaKwik. The KlustaKwik method found 17 spike clusters under the AIC penalty and 9 clusters under the BIC penalty after manual pruning of noise clusters. As expected, the AIC penalty yielded many more spike clusters than its BIC penalized counterpart. The clusters found using WaveClus were chosen according to minimum cluster size and temperature. In an iterative fashion, clusters were first identified from the paramagnetic temperature region and fixed. Then new clusters were obtained from the set of spikes exclusive from the previously fixed clusters after the temperature was reduced into the superparamagnetic range. This selection procedure resulted in selection of 11 spike clusters.

Both unsupervised techniques obtained far fewer spike clusters than manual cluster cutting. There is not enough data at this time to properly discuss the stability and longevity of the clusters found by the two automatic sorting systems as he Neuralynx spike sorting system was consistently used in all recording sessions. However, one can hypothesize their stability would be comparable to the convex-hull technique since the main feature of the Neurotrophic Electrode is signal stability. Planned future analysis will confirm or reject this hypothesis.

3.4.4 Objective cluster quality assessment

Schmitzer-Torbert et al. (2005) introduced a metric, L_{ratio}, for objective quality as-

sessment of sorted spike clusters based on a set of standard spike waveform features. The measure is defined by,

$$L(C) = \sum_{i \notin C} 1 - CDF_{\chi^2_{df}}(D^2_{i,C}) \tag{3.1}$$

$$L_{\text{ratio}}(C) = \frac{L(C)}{n_C} \tag{3.2}$$

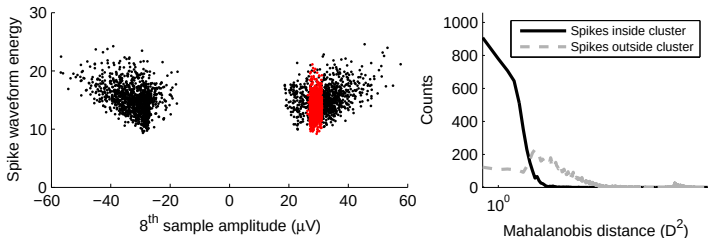
where i are the spikes, C are the spike clusters, $D^2_{i,C}$ is the Mahalanobis distance from spikes i to the center of cluster C taken in *df* dimensions (i.e. the dimension of the feature vectors used to describe each spike waveform) and $L(C)$ is the L-value for cluster C. $L_{\text{ratio}}(C)$ is taken with respect to $L(C)$ and is simply the L-value normalized by the number of spikes in the cluster, n_C. Briefly, the L_{ratio} describes how well a particular cluster is separated from all other points not in the cluster with low values indicating well separated clusters. This measure for spike cluster quality is based on the assumption that clusters are Gaussian distributed. If so, then the Mahalanobis distances will be χ^2 distributed permitting more rigorous statistical evaluation. The value of L_{ratio} is highly dependent on the choice of features and has been shown to monotonically decrease as the number, but not necessarily quality, of included features increases. However, it can be an informative measure of cluster separation when the evaluation feature space is held constant across comparisons. This measure was used to quantitatively examine the quality of cluster separation among the four clustering techniques evaluated on spike energy and 8^{th} sample feature dimensions. While L_{ratio} was designed specifically for spike waveform clustering applications, a more general approach could consist of a stepwise forward and backward methodology utilizing Neyman-Pearson criteria (Neyman and Pearson, 1933) for cluster selection. Additional cluster selection and quality measurement techniques exist (e.g. Fisher's linear discriminant; Fisher, 1936) but were not considered in this dissertation. Comparisons

of alternative measures against the L_{ratio} will be investigated by future work.

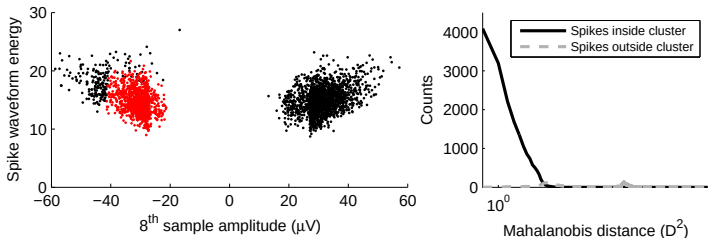
Figure 3·4 graphically illustrates the spike cluster separability for the best separable spike cluster of each sorting technique.[6] On the left, clustered spike waveforms are plotted on the 8^{th} sample amplitude - spike waveform energy plane. The best separated cluster is shown in red while all others in black. The Mahalanobis distance was computed for each spike with respect to the best separated cluster in all feature dimensions. The distribution of squared Mahalanobis distance for both sets of points is illustrated in the right column of Figure 3·4 with the within cluster distribution in solid black and outside distribution in dashed gray. Mahalanobis distance measures the dissimilarity of data samples and cluster centroids, therefore, the optimally separated cluster will have zero overlap of spikes with Mahalanobis distance of spikes within and outside the selected cluster. Comparison among best separated clusters for each technique indicated that the BIC penalized KlustaKwik method yielded the best separation ($L_{\text{ratio}} = 0.001$) followed by the AIC penalized KlustaKwik, manual sorting via SpikeSort3D and WaveClus ($L_{\text{ratio}} = 0.002,\ 0.088$ and 0.17, respectively).

A one-way ANOVA of L_{ratio} was performed to assess global separability among all four clustering algorithms. The results were inconclusive, as there was no significant effect of sorting method ($F = 0.06, p = 0.98$). Subsequent *post-hoc* Tukey-Kramer tests of group means also yielded an inconclusive result. No group means were significantly different, although analysis of the general trend indicated that BIC penalized KlustaKwik had the lowest mean L_{ratio} (but the highest variance) followed by AIC penalized KlustaKwik, manual sorting and WaveClus ($\bar{L}_{\text{ratio}} = 3.8,\ 4.2,\ 4.3$ and 4.8, respectively).

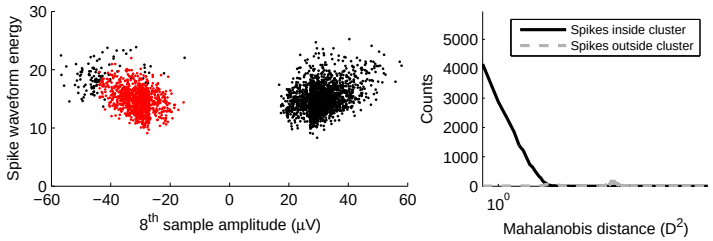
[6] It is not practical to display all combinations of clusters, therefore only the best separated clusters were chosen for Figure 3·4.



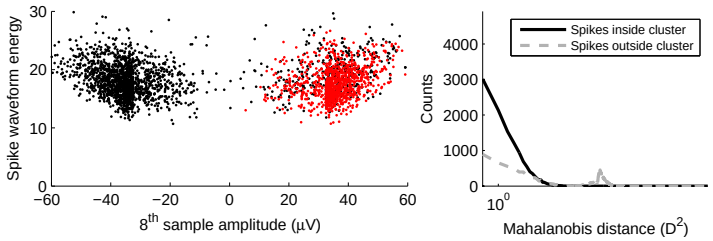
(A) SpikeSort3D



(B) KlustaKwik (AIC)



(C) KlustaKwik (BIC)



(D) WaveClus

Figure 3·4: Graphical representation of cluster separation for a single cluster of each sorting technique. Left: spike cluster distribution in the 8^{th} sample amplitude - spike waveform energy plane. Red points indicate selected cluster and black points are the remaining spikes. Right: distributions of squared Mahalanobis distance of within-cluster (solid black) and outside (dashed gray) spikes.

Source	Sum Sq.	d.f.	Mean Sq.	F	Prob>F
Clust. Alg.	6.2246	3	2.0749	0.0627	0.9793
Error	$1.9857e^3$	60	33.0947		
Total	$1.9919e^3$	63			

Table 3.3: ANOVA summary for tests of clustering method separability factor, L_{ratio}. There is no significant main effect.

3.5 Discussion

This chapter brings to light the vast number of parameters and variables that must be accounted for during each analysis session. Choosing the optimal filter settings, threshold type and value as well as ensuring proper spike waveform alignment while minimizing methodological and human error is not a trivial task.

Four different techniques were used to separate putative action potentials from background noise obtained from raw waveform recordings of the Neurotrophic Electrode implanted in human speech motor cortex. The different methods varied in exclusiveness of putative spike occurrences. In particular, the NEO methods (NEO and SNEO) showed they were very conservative in their estimate while the ABS and rRMS maintained more relaxed criteria. Though a detailed quantitative analysis was not performed, previous studies have shown gains in signal detection utilizing the NEO and SNEO preemphasis (Mukhopadhyay and Ray, 1998; Kim and Kim, 2000). Data-driven thresholds are desirable as they are an objective separator of signal from noise allowing unbiased comparison of preemphasis techniques. However, the manually set positive and negative (i.e. absolute value) voltage threshold performed reliably and appeared to find a superset of putative spikes obtained via the NEO and SNEO techniques.

The comparison of four different methods for isolation of individual units found on a multiunit extracellular electrode showed a relative insensitivity of sorting technique

to cluster quality. Individual cluster quality analysis showed that while BIC penalized Bayesian clustering achieved the single best spike cluster separation, no method had statistically significantly better global separation than any other. Qualitatively, manual cluster-cutting and Bayesian clustering were of similar cluster separation quality while clustering by SPC yielded suboptimal results. Further, though Bayesian clustering performed better in general, it showed excessive variance of cluster discrimination.

3.6 Conclusion

Despite the effort and possible error involved in manual selection of spike clusters, it remains the method of choice for most experimental neuroscientists. It is possible for an expert sorter to define regions based on experience and intuition (as well as visual grouping) leading to a finely tuned spike classification. In addition, the Neuralynx, Inc. data acquisition system has built-in support for spike sorting using SpikeSort3D in real-time, which is a necessary requirement for the implementation of a neural prosthesis, let alone one for production of speech. Both semi-automated methods evaluated in this dissertation are grounded in rigorous mathematics and find statistically optimal spike clusters. However, the spike acquisition and analysis environment currently in use precluded real-time utilization of either method.

The combined evidence of spike detection performance and spike classification cluster quality led to selection of simple, expert user determined, voltage thresholds and manually defined spike cluster convex-hull regions for use in the real-time BCI developed in this dissertation. Though all three methods are likely to provide reasonable isolation of individual units recorded on the Neurotrophic Electrode, at this time only SpikeSort3D has the required real-time capability. Therefore, despite the statistically optimal clustering utilized in the semi-supervised techniques, the SpikeSort3D

manual cluster cutting method was selected for use in the BCI implementation. Ongoing research is aimed at investigating the application of additional techniques to the real-time system and proper estimation of the optimal number of useful spike clusters.

CHAPTER 4

ESTIMATION OF NEURAL FIRING RATES

4.1 Introduction

Spike trains obtained through spike sorting are the basis for a large majority of invasive neural decoding techniques. Some methods operate directly on the neural spike trains, explicitly deriving correlations between synchronous firing of groups of single units within an information-theoretic framework (Grün et al., 2001a,b; Thorpe et al., 2001; Pipa and Grün, 2003). However, the vast majority of neural decoding methods, at some level, require estimation of the firing rate of each individual unit.

Perkel et al. (1967) established the neural spike train as a stochastic point process. This realization has enabled rigorous statistical analysis of the properties and characteristics of recorded neurons. For instance, the Poisson process is a simple class of stochastic point process that can characterize neural spike trains (Perkel et al., 1967). Under this definition, the basic measure of neural firing activity is the Poisson rate parameter (i.e. the average firing rate). Such a measure has been successfully used to describe the firing characteristics of monkey motor cortex (Georgopoulos et al., 1982), but makes the incorrect assumption that firing rate is constant. Today, a number of techniques exist for capturing instantaneous firing activity including temporal effects, biological constraints and more.

The conditional intensity function (Daley and Vere-Jones, 2003)[1] is a more sophisticated measure of neural activity and is defined for any point in time, t, in the

[1] Also known as the hazard function in survival analysis

sequence of spike times $(0, T]$,

$$\lambda(t|H_t) = \lim_{\Delta \to 0} \frac{Pr(N(t+\Delta) - N(t) = 1|H_t)}{\Delta}, \tag{4.1}$$

where H_t is the spike history, $N(t)$ is the count of spikes in $(0, t]$ and $Pr(\cdot|\cdot)$ is the conditional probability (Brown et al., 2004). The stochastic process defined by the full conditional intensity with time and history effects is known as a Cox process. Conceptually, the conditional intensity can be thought of as the neural firing rate. The Poisson rate parameter, discussed above, is a special case of the conditional intensity that is independent of time and history effects,[2] specifically the conditional intensity is taken as a constant value. This rate estimator assumes that the stochastic point process is stationary, or homogeneous, across all time. Another special case of the Cox process includes time but not prior spiking history. Under this definition, the spike train is a non-stationary, or inhomogeneous, Poisson process described by $\lambda(t|H_t) = \lambda(t)$. Rate is represented as a parameter of the model used to characterize the firing behavior. The methods discussed below all are estimators of the inhomogeneous Poisson (IP) or Cox processes.

In the following sections, a number of models will be used to estimate the conditional intensity (i.e. firing rate) of example fast and slow firing neurons both simulated and recorded from subject S1. Statistical evaluation will be used to determine the effectiveness of each model for its application in a neural prosthesis for speech production.

[2]The conditional intensity, $\lambda(t|H_t)$, is taken as a constant value λ.

4.2 Firing rate models

This dissertation considers two main classes of firing rate estimators: 1) direct rate estimators or smoothers and 2) Interspike interval (ISI)-based convolution filter methods. The former attempts to estimate the instantaneous firing rate at any point in time by first computing the rate in consecutive adjacent local regions (e.g. the histogram method). The latter estimates rate based on convolution of an appropriate filter with the spike train:

$$\hat{y}(t) = \int_{-\infty}^{\infty} \delta(t)h(t-\tau)d\tau \tag{4.2}$$

where $\delta(t)$ is the spike train represented as a sum of delayed impulses and $h(t)$ is an appropriate filter.

4.2.1 Direct rate estimators

Binned Histograms

Constructing histograms of spike frequency is a very common, and very simple, non-parametric measurement of instantaneous firing rate. In this case, firing rate is defined as an estimate of the IP rate parameter, $\lambda(t)$, approximated by the spike frequency histogram. Simply put, this estimator counts the number of spikes occurring in a particular time bin, with bin size Δt. Firing *rate* is obtained by division of the bin count, $N(t+\Delta t) - N(t)$ by the time interval Δt,[3]

$$\hat{y}(t) = \frac{N(t+\Delta t) - N(t)}{\Delta t}. \tag{4.3}$$

While extremely fast and easy to compute, they assume constancy of rate over bins. In addition, spike rate histograms tend to have high variability requiring smoothing,

[3]This is equivalent to convolution of the spike train with a rectangular window of size Δt.

the simplest of which is just convolution of the spike rate histogram with a Gaussian window.

Kernel density estimates

Kernel density estimates (KDE) compute a smoothed version of the binned histogram by convolution of the spike train with a suitable (e.g. Gaussian) kernel. Suitable kernels, K, are positive symmetric and integrate to one:

(i) $K \geq 0$

(ii) $K(t) = K(-t)$

(iii) $\int_{-\infty}^{\infty} K(t)dt = 1$

Parzen's (1962) original kernel density estimator is given by,

$$\hat{f}(y) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x_i - y}{h}\right), \tag{4.4}$$

where x_i is an independent and identically distributed random variable of size N, K is a suitable kernel, and h is the kernel bandwidth, or smoothing parameter. Like the histogram approach, the Parzen KDE assumes constant bandwidth. Modern versions of the KDE seek to estimate the optimal bandwidth, h^*, as a fixed quantity (Park and Marron, 1990; Sheather and Jones, 1991) or adaptively (Loftsgaarden and Quesenberry, 1965; Breiman et al., 1977; Katkovnik and Shmulevich, 2002; Comaniciu, 2003).

Early modifications to the fixed KDE attempted to account for local density in the data by allowing the smoothing parameter, h, to vary. One such method determines the bandwidth as the Euclidean distance from the estimation point, y, to the k^{th}

nearest sample point, x_i, given by

$$\hat{f}(y) = \frac{1}{nh(y)} \sum_{i=1}^{n} K\left(\frac{x_i - y}{h(y)}\right) \tag{4.5}$$

where $h(y)$ is a nearest-neighbor smoothing parameter (Loftsgaarden and Quesenberry, 1965). Another formulation calculates the smoothing parameter by the Euclidean distance from the sample point, x_i to the k^{th} nearest other sample point (Breiman et al., 1977), given by

$$\hat{f}(y) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{h(x_i)} K\left(\frac{x_i - y}{h(x_i)}\right) \tag{4.6}$$

where $h(x_i)$ is the nearest-neighbor smoothing parameter. Terrell and Scott (1992) termed the former method a *balloon* estimator and the latter a *sample smoothing* estimator and concluded that both estimators do not improve performance against a Parzen (1962) estimator with optimally selected bandwidth.

Modern versions of both the fixed and adaptive bandwidth KDE have been formulated to obtain an optimal kernel bandwidth. A "plug-in" method has been developed to find the optimal fixed bandwidth (Park and Marron, 1990; Sheather and Jones, 1991). Briefly, it finds the bandwidth that minimizes an asymptotic approximation of the mean integrated squared error (MISE) between the estimated density and the unknown, true density. The plug-in estimator is a numerical method that iteratively searches for the optimal bandwidth given a pilot distribution. The optimal bandwidth is estimated by solution of:

$$\hat{h}^* = \left(\frac{R(K)}{\sigma_K^4 R(f'')N}\right)^{1/5} \tag{4.7}$$

where $R(g) = \int_0^\infty g(x)dx$ and $\sigma_K^2 = \int_0^\infty x^2 g(x)dx$. Notice, this method requires knowledge of the derivatives of the unknown function $f(x)$, which is iteratively estimated

using kernel density approximations to $f(x)$.

Two methods developed to compute the optimal adaptive bandwidth use the intersection of confidence intervals (ICI) rule (Katkovnik and Shmulevich, 2002) and mean shift-variable bandwidth (Comaniciu and Meer, 2002; Comaniciu, 2003). Both methods allow the kernel bandwidth to change as a function of the local data statistics and are direct descendants of nearest-neighbor adaptive bandwidth kernel density estimators (e.g. Loftsgaarden and Quesenberry, 1965). The adaptive bandwidth KDE method selected for comparison (i.e. the variable bandwidth mean-shift method) iteratively estimates the mode(s) of the underlying distribution utilizing an adaptive gradient ascent (i.e. the mean shift procedure), then computes the most stable local bandwidth using the mean shift trajectory across multiple analysis scales (see Comaniciu, 2003 for a detailed description).

Kernel density estimates, by their definition, are used to estimate the probability density of a data set of which there is no *a priori* knowledge. The most common application is in image processing for segmentation and parcellation (Comaniciu and Meer, 2002; Comaniciu, 2003; Raykar and Duraiswami, 2006), and these methods have not traditionally been used for neural firing rate. However, Olson et al. (2000) evaluated the Sheather and Jones plug-in bandwidth KDE as an estimate of the neural intensity function, $\hat{\lambda}(t)$. Both the plug-in methods of Sheather and Jones (1991) and adaptive bandwidth methods of Katkovnik and Shmulevich (2002) and Comaniciu (2003), when used to estimate the local density of neural spike trains, can be interpreted as non-parametric, data driven estimates of the unknown IP rate.

Regression splines

Regression splines offer another method for smoothing the spike rate histogram. Olson et al. (2000) explores the usage of cubic spline regression of the log neural firing rate

initially estimated using the histogram approach. The log-linear regression spline follows the form:

$$\log \hat{\lambda}(t) = \beta_0 + \sum_{p=1}^{P} \sum_{k=1}^{K} \beta_j (t - \xi_k)_+^p \tag{4.8}$$

with appropriately chosen knots ξ_k, spline order P (i.e. cubic spline, $P = 3$) and coefficients β_j ($j = (p-1)K + k$) solved via generalized linear model (GLM). Additional regression spline techniques have been used by other researchers (e.g. Kass and Ventura, 2001), but were not considered in this research. They will likely provide insight into the underlying spike train process and will be investigated in future research.

4.2.2 ISI-based convolution filter estimators

The firing rate models from the previous section all attempt to directly estimate the spiking activity without *a priori* information about the neuron's firing characteristics. As noted in Section 4.2.1, the histogram and KDE firing rate estimation techniques can be thought of as convolution of the neural spike train with rectangular and Gaussian windows, respectively. The methods below can be considered variants of the Gaussian KDE utilizing causal convolution filters as opposed to acausal (e.g. Gaussian). These filters are derived through modeling of the ISI distribution, then the IP rate parameter is estimated via convolution of the ISI-defined filter with the neural spike train.

Parametric models

The ISI distribution describes the probability with which a neuron fires at some time after the last occurred spike. Often, these distributions are characterized by a simple Poisson process, defined by the exponential probability distribution:

$$p_{\text{Exp}}(w_n|\alpha) = \alpha e^{-\alpha w_n}, \tag{4.9}$$

where w_n are the ISIs and α is the estimated rate parameter representing the average ISI (also reciprocal of average firing rate). The exponential family of distributions are, in general, appropriate estimates of the empirical ISI distribution; furthermore the gamma (G) and inverse Gaussian (IG) densities[4] have been suggested as more accurate than the simple exponential (Bishop et al., 1964; Tuckwell, 1988; Iyengar and Liao, 1997; Barbieri et al., 2001; Brown et al., 2004) with both the G and IG densities having additional parameters to better model the integrate-and-fire properties of neurons (Barbieri et al., 2001; Brown et al., 2004). In particular the gamma distribution,

$$p_{\mathrm{G}}(w_j|\alpha,\beta) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} w_j^{\alpha-1} e^{-\frac{w_j}{\beta}} \tag{4.10}$$

with location parameter α and scale (or rate) parameter β, is used to simulate the behavior of an integrate-and-fire model with constant Poisson inputs. The inverse Gaussian distribution,

$$p_{\mathrm{IG}}(w_j|\mu,\lambda) = \left(\frac{\lambda}{2\pi w_j^3}\right)^{1/2} e^{-\frac{1}{2}\frac{\lambda(w_j-\mu)^2}{\mu^2 w_j}} \tag{4.11}$$

with mean μ and shape parameter λ, simulates the same integrate-and-fire model with Brownian changes in membrane voltage potential (Brown et al., 2004).

Once the model is chosen and fit to the data, an appropriate filter can be constructed (e.g. $h(z) = p_{\mathrm{Exp}}$ for the exponential distribution). Then an estimate of the nonstationary firing rate can be obtained according to Equation 4.2.

Non-parametric ISI density estimates

It is possible that the natural ISI distribution does not adequately follow one of the three parametric models proposed in Section 4.2.2. Therefore, following the approach in Section 4.2.1, it is possible to approximate the ISI distribution via KDE with

[4]The Exponential probability density is a special case of both the G and IG densities.

optimal static bandwidth selection. Similar to the process outlined in Section 4.2.2, this *non-parametric* estimate of the ISI distribution is taken as the convolution filter, $h(z)$ in Equation 4.2, and the nonstationary firing rate is obtained.

Non-stationary parametric models

A novel, alternative parametric model was developed to obtain both an instantaneous firing rate estimate utilizing knowledge of the ISI distribution *and* a more accurate model of the ISI distribution. The model is based on the standard exponential distribution described in Section 4.2.2 but differs by allowing the rate parameter to slowly change according to baseline spiking rate, implicitly incorporating all previous spiking history. This model is defined by the following equations:

$$y_k = \log\left(1 + e^{y_{k-1} - \gamma\delta_k/\tau_{k-1}}\right) \tag{4.12}$$

$$\tau_k = \tau_{k-1} + \left(1 - e^{-\varepsilon\delta_k}\right) \cdot (\delta_k - \tau_{k-1}) \tag{4.13}$$

where γ is a scaling factor, τ_k is the adaptive rate parameter, ε is the learning rate parameter for τ_k and δ_k is the time since the last spike, $t - t_k$. The normalized firing rate, or neural activity, is estimated by y_k and depends on the slowly-varying decay parameter, τ_k. A suitable value of 0.5 was chosen for ε which implies the model will ideally produce rates normally distributed with mean $\frac{1}{\varepsilon}$. The models described in Section 4.2.2 assume that the ISI distribution does not change over time. This assumption is not always valid. In contrast, the model described in Equation 4.12 is variable over time, adapting to local changes in the spike train. This variability is characterized by longer filters for slow spike rates and narrow filters for fast spiking activity, having the effect of normalizing the estimated firing rate despite wide changes in the baseline rate. This formulation of an adaptive exponential filter was developed due to observations of slow changes, and wide variability in global neural activity on

the order of hours to days. By adapting the time constant, τ_j, of the exponential filter as a function of the baseline rate an instantaneous estimate, y_k, is obtained that is *normalized* across and within recording sessions. Interestingly, this method, though kernel-based, is related to the class of penalized linear splines described in detail by Cai et al. (2002). This relationship can provide a theoretical validation of the adaptive exponential rate estimator.

4.2.3 Goodness-of-fit

The most straight forward metric used for GOF tests is the Akaike information criterion (Akaike, 1974), and is calculated via the negative log-likelihood of the fitted density:

$$\text{AIC} = -2 \log L(\hat{\theta}|N_{0:T}) + 2q \tag{4.14}$$

where $\log L(\hat{\theta}|N_{0:T})$ is the log-likelihood, $N_{0:T}$ are the spike counts from $0 : T$ and q is the dimension of parameter θ. Brown et al. (2004) have noted that while AIC is a good measure for model selection, it is only a single value and cannot describe *how* a particular model fails a GOF test. Therefore a novel application of the time-rescaling theorem has been applied by Brown et al. (2001) to develop more descriptive GOF tests.

The time-rescaling theorem states that any renewal process can be transformed into a homogeneous Poisson process with unit rate (see Brown et al. (2001) for details). The transformation is as follows,

$$\tau_j = \int_{u_{j-1}}^{u_j} \lambda(u|H_u)du, \tag{4.15}$$

where $\lambda(u|H_u)$ is the conditional intensity function for a spike train with spike history H_u and spike times u_j for $j = 1, \cdots, J$. Typically, the rescaled time points, τ_j, if

correct should be Poisson distributed with unit rate and can be transformed into independent uniform random variables according to,

$$z_j = 1 - e^{-\tau_j}. \tag{4.16}$$

The transformed uniform random variables, z_j can be used to estimate the model quantiles needed to perform statistical tests of agreement between the unknown unknown random variable (i.e. z_js) and the known uniform distribution. The z_js are ordered and plotted against the cumulative distribution function (CDF) of the uniform distribution (i.e. quantile-quantile, or Q-Q, plot). The model GOF is established by evaluating the degree to which the transformed points lie on the 45° line (Brown et al., 2001). Unfortunately, the assumption of unit Poisson rate rescaled time points is not valid for the adaptive exponential method and yields misleading GOF results. It is possible to relax the unity rate constraint by testing that the rescaled time points are simply homogeneous Poisson distributed. The GOF test remains largely the same, except the rescaled points, τ_j, are plotted against the exponential distribution. For a truly exponential function, these points should lie on a straight line with slope equal to the unknown rescaled exponential rate parameter. Confidence bounds for the model fit are determined using the distribution of the Kolmogorov-Smirnov (K-S) statistic, specifically the 95% confidence interval is approximated by $b_k \pm 1.36/n^{1/2}$. The K-S confidence bounds plotted with the empirical (uniform CDF, or unknown rescaled exponential) and model (rescaled times) quantiles yield the K-S plot for graphical evaluation of model fit (Brown et al., 2001).

4.3 Materials and methods

Multiunit recordings were taken from the Neurotrophic Electrode implanted in subject S1 according to the methods described in Section 1.3 and spike trains were computed by manually defined convex hull classification of putative spikes detected though hand-tuned positive and negative voltage threshold crossings (see Chapter 3). A 50 s sample for example slow and fast spiking units taken from the continuous SPEAK-ALONG task[5] was extracted for evaluation of the firing rate estimators described within this chapter. In addition, simulated slow (1 Hz) and fast (10 Hz) firing homogeneous Poisson spike trains were constructed for objective (i.e. non-task specific, nor constrained to the present experimental setup) evaluation.

Estimated instantaneous firing rates for both real and simulated data were taken at 15 ms intervals (67 Hz) using all methods detailed in Section 4.2. The KDE and adaptive exponential methods require a "learning" period, a time for which the estimate is underspecified and the KDE methods require an additional "cool down" period;[6] therefore, all statistical analyses of the rate estimators will consider only the 30 s time region [10 s,40 s] for simulated data. An additional constraint on analysis window was employed for real data. The constraint limits analysis to periods of time in which the ISIs are relatively small (i.e. $<$10 s). The addition of the ISI constraint limited analysis of the real data to the time region [10 s,30 s] as the example slow spike train had a large ISI between the spikes at $t_0 = 30$ s and $t_1 = 40$ s. For the spline based method, knots are assigned at midpoints of steady portions of the acoustic stimulus (i.e. every 1.3 s starting at 0.5 s after the onset of the stimulus).

For all methods, quantitative and graphical analysis of goodness-of-fit were com-

[5]See Chapter 5 for a description of the task. The time range of neural activity taken included 10 s prior to the onset of the speak-along stimulus to 40 s after.

[6]The KDE methods require data points on both sides of the sample point and the adaptive exponential method needs to acquire a stable decay factor, τ_k.

puted using the modified rate-rescaling theorem method (Brown et al., 2001) described in Section 4.2.3. In addition, statistical tests for normality were applied to the distributions of estimated firing rates to test rejection of the alternative hypothesis that the firing rate distributions come from a non-Gaussian distribution. This test is important as subsequent analyses (see Chapter 5) rely on assumptions of log-normal distributions of firing rates. To that end, all firing rates are log-transformed[7] as this transformation has the effect of making the data more normal.

4.4 Results

4.4.1 Simulated data

The direct estimates of firing rate for the low and high frequency simulated spike trains are shown along with spike rasters in Figure 4·1. The rate histogram with 150ms bin width is shown in gray. The remaining three direct rate estimators, static KDE, variable KDE and regression spline are shown in blue, red and green, respectively. Parametric (exponential, gamma and inverse Gaussian) and non-parametric (KDE) model fits of the ISI distribution are illustrated in Figure 4·2 with corresponding ISI-convolution rate estimates in Figure 4·3. In both figures the ISI and rate histogram is shown in gray and the ISI-convolution rate estimates in blue, red, green and magenta for the exponential, gamma, inverse Gaussian and KDE methods, respectively. It is not possible to show an ISI distribution fit using the adaptive exponential estimator given its non-stationary nature though the estimated rate is displayed in cyan in Figure 4·3. Both simulated spike trains are stationary and Poisson, therefore estimated firing rate should be approximately flat. In addition, since they are stationary, they can be adequately described by the estimated mean firing rate.

[7] $\log(x+1)$, alternatives include the square-root transform.

The rate histogram for both slow and fast firing spike trains[8] are highly variable and, as is clear from the slow rate histogram estimate, sparsely populated yielding excessive zero firing rates, reducing the amount of information conveyed by the rate model. On the other hand, the mean firing rate for the static KDE was 0.79 Hz and 10.36 Hz (see Table 4.1 for a statistical summary of each method) for the slow and fast spike trains while mean rate was estimated 0.79 Hz and 10.59 Hz for the slow and fast spike trains according to the regression spline model. Both the static KDE with plug-in bandwidth and regression splines appear to well approximate the firing rate of the underlying simulated homogeneous Poisson processes and remain flat, or constant for the entire spike train duration. Interestingly, the mean shift adaptive bandwidth KDE underestimates the firing rate in both simulated spike trains. The method certainly captures changes in local spiking activity, though the ideal instantaneous firing rate estimate ought to be flat for a stationary spike train.

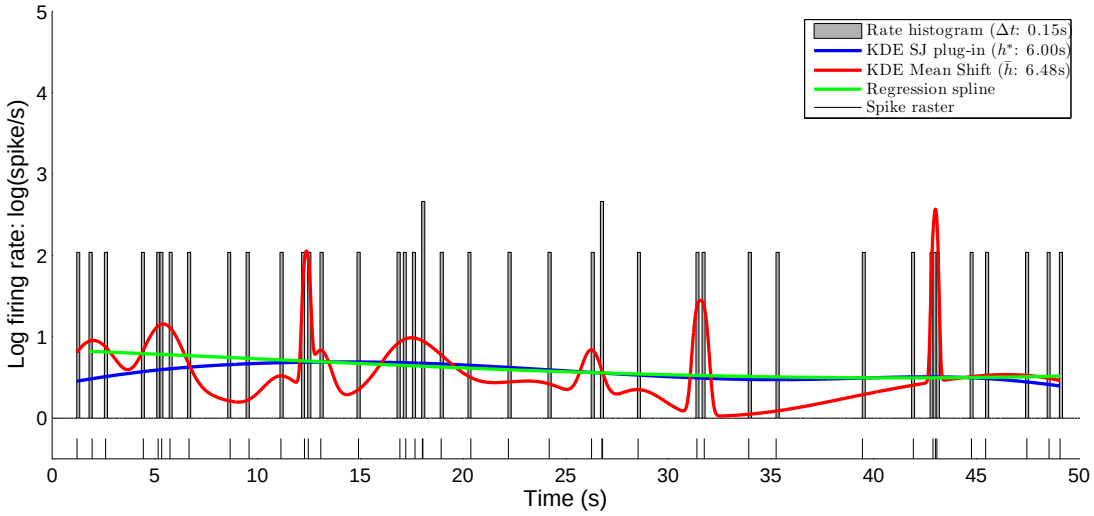
All of the ISI-convolution methods for rate estimation slightly underestimate the true mean firing rate of each spike train type, though all (except the adaptive exponential due to its normalization property) approach the true rate. These methods capture the same local dynamics found by the adaptive KDE technique while maintaining stable estimates of the true firing rate. Notice, in Figure 4·2, that all four ISI distribution fits follow the same approximate shape for both slow and fast simulated spike trains. The location parameters shift the peak of the distribution away from $t = 0$ s for the gamma and inverse Gaussian models as did the non-parametric KDE in the case of the slow spike train. The KDE fit shows additional distribution complexity (i.e. multiple modes) at later ISIs. The same features are captured by inverse Gaussian and KDE model fits only for the simulated fast spike train.

[8]Recall that slow spike trains are homogeneous Poisson processes with 1 Hz (0.69 log Hz) rate and fast with 10 Hz (2.40 log Hz) rate.

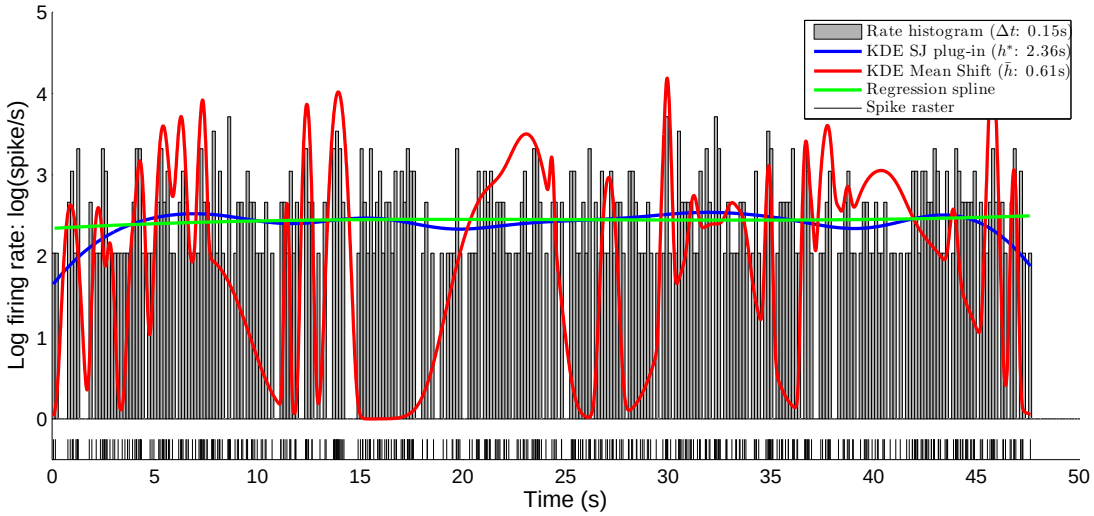
The modified K-S model quality test (see Figure 4·4) was applied via time-rescaling theorem (Brown et al., 2001) to examine the goodness-of-fit for each rate estimation technique. The rescaled time points were plotted against the exponential distribution and the slope of the straight line fit was taken as an estimate of the unknown homogeneous Poisson rate parameter of the rescaled firing rates recorded in Table 4.1. The points were then scaled according to the estimated homogeneous Poisson rate and normalized to occupy the unit range (0,1). Most estimators, when time-rescaled, were exponentially distributed with approximately unity rate. However, exceptions did occur in the slow spike train as the histogram, inverse Gaussian and adaptive exponential deviated from unit rate as did the variable KDE and all ISI-convolution estimators for the fast spike train. All direct rate estimates (except for the rate histogram) fell completely within the 95% K-S boundary for the slow spike train though they tended to overpredict the correct rate in middle quantiles (approximately 0.25-0.75). Only the variable KDE did not entirely fall between the 95% K-S interval for the fast spike train though all methods showed tendencies away from the diagonal at upper quantiles. All ISI convolution filter methods were good fits to the slow spike train with slightly lower overall deviation from the ideal as compared to the direct rate estimators. Only the adaptive exponential remained totally within the K-S bounds for the fast spike train, though the inverse Gaussian makes only slight deviations roughly between the 0.5 and 0.75 quantile range.

According to subsequent analysis constraints, the ideal estimator will generate instantaneous firing rate values that are normally distributed. The χ^2 and one sample Kolmogorov-Smirnov tests for normality along with estimates of kurtosis and skewness were used to determine the degree of normality achieved by each technique. These measures, shown in Table 4.1, indicated that only the adaptive exponential method obtained normally distributed firing rates. Graphically, this result is verified by the

approximately Gaussian shape of firing rate counts shown in Figure 4·5. Rather than capturing the true firing rate of the simulated spike trains, the adaptive exponential rate estimator obtained firing rates that were distributed according to the normal distributions, $\mathcal{N}(1.35, 0.18)$ and $\mathcal{N}(1.92, 0.28)$ for slow and fast spike trains. Under certain conditions, the exponential, gamma, inverse Gaussian and KDE methods for ISI convolution, appear to be Gaussian distributed, but under both tests for normality, all can reject the null hypothesis of normality with high significance. Interestingly, though the direct rate methods tended to have more stable firing rate estimates, none come close to achieving a normal distribution of values.

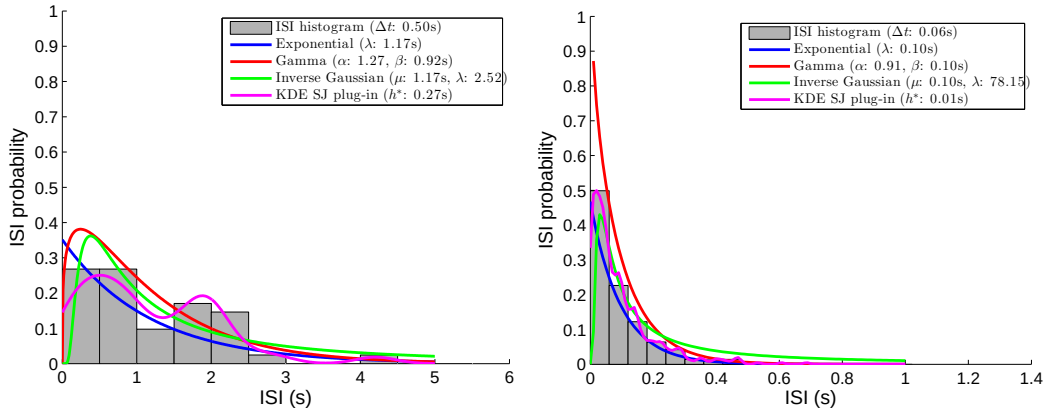


(A)



(B)

Figure 4·1: Direct rate estimates for a 50 second sample from example slow (top) and fast (bottom) firing simulated spike trains. Rate histogram is shown in gray, static KDE in blue, variable KDE in red and regression spline in green. Legend contains fitted model parameters.



(A) (B)

Figure 4·2: Model fits of the simulated ISI distribution. The ISI histogram is shown in gray, exponential in blue, gamma in red, inverse Gaussian in green and ISI KDE in magenta. Estimates were taken from slow (left) and fast (right) simulated spike trains. The adaptive exponential is not explicitly shown but is characterized by the static rate exponential fit. Legend contains fitted model parameters.

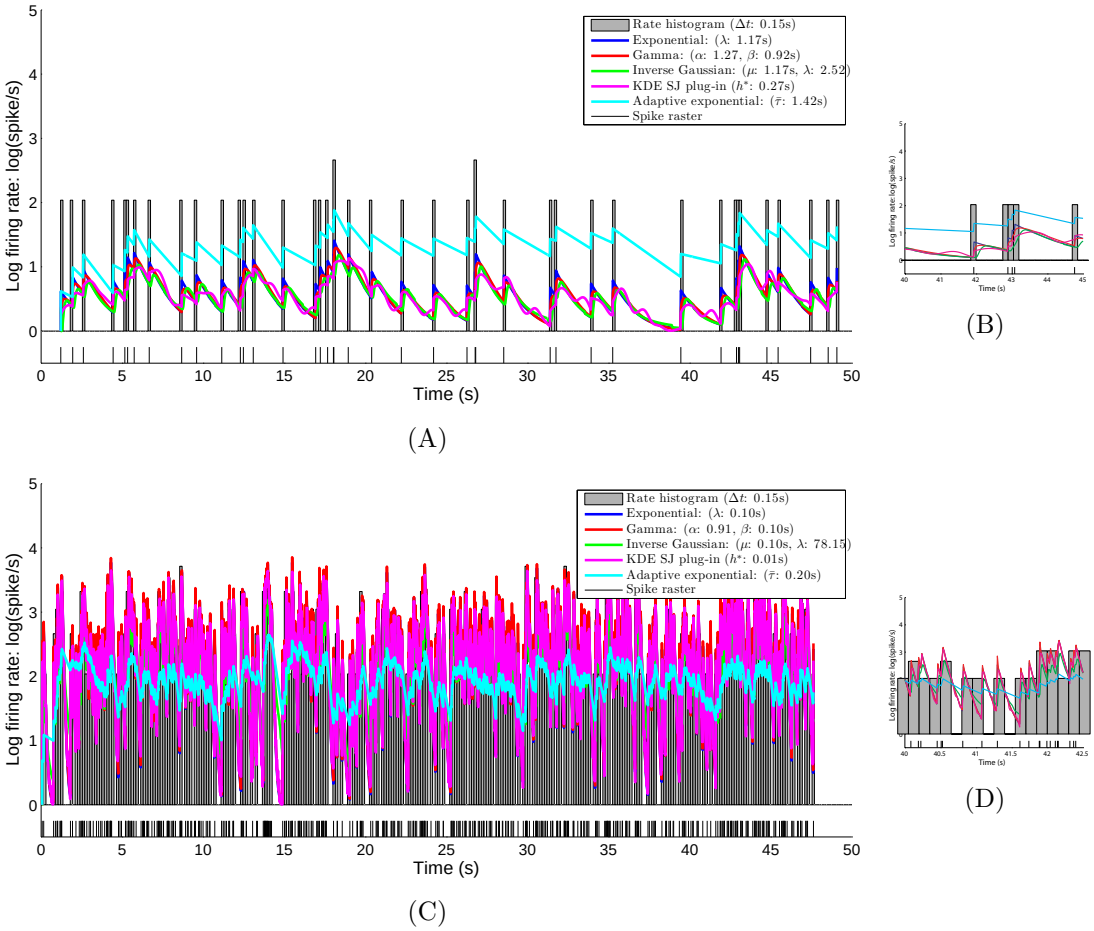


Figure 4·3: ISI-convolution rate estimates for a 50 second sample from example slow (top) and fast (bottom) firing simulated spike trains. (Left) Rate histogram in gray, static exponential in blue, gamma in red, inverse Gaussian in green, ISI KDE in magenta and adaptive exponential in cyan. Legend contains fitted model parameters. (Right) Zoomed in portions of the estimated rate illustrating the dynamics of each estimator.

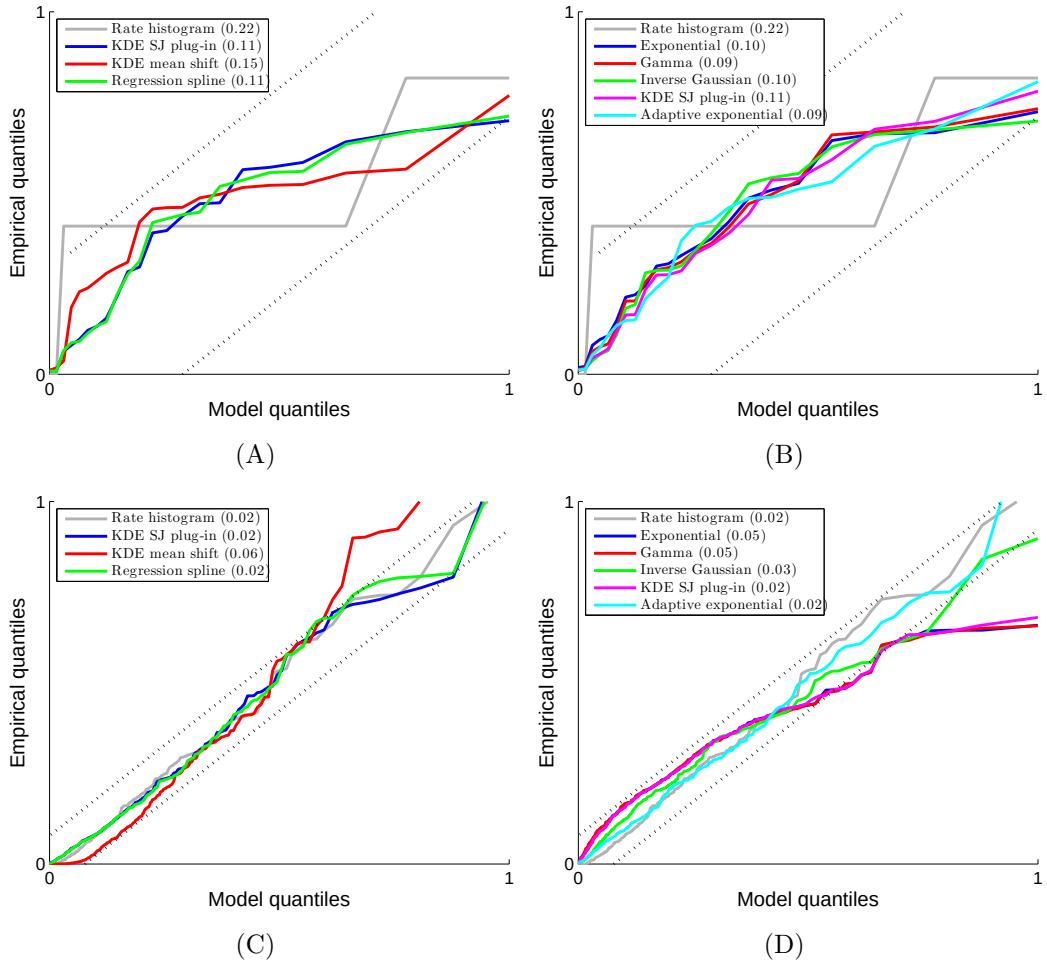


Figure 4·4: K-S plot of modified time-rescaled conditional intensities vs. the exponential. Valid models correspond to points on the 45° line. Dotted lines are 95% confidence intervals based on the K-S statistic. (Left) Direct rate models. (Right) ISI convolution methods. (Top) Simulated example slow firing units. (Bottom) Simulated example fast firing units. Legend contains RMSE of plotted points from the diagonal.

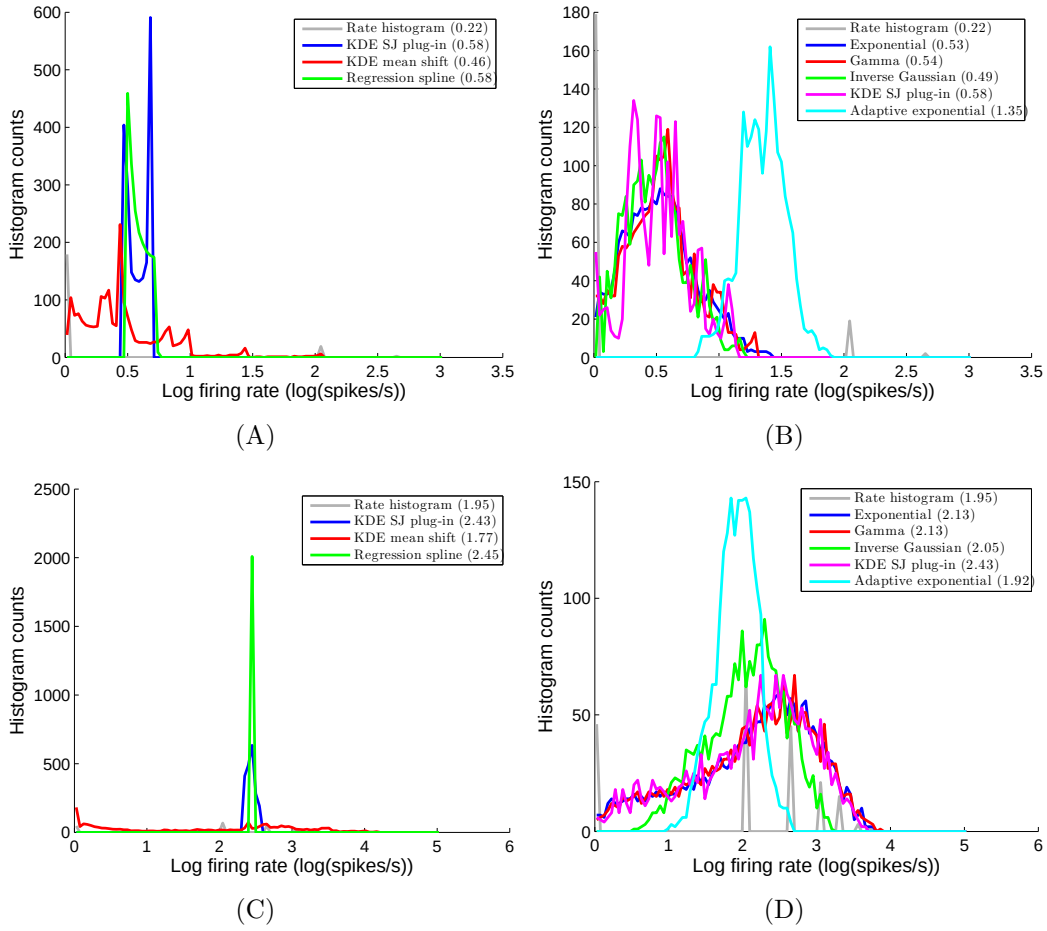


Figure 4·5: Histograms of simulated spike train firing rate distributions for each model. (Left) direct rate models. (Right) ISI convolution methods. (Top) slow firing units. (Bottom) fast firing units. Legend contains mean firing rate for each estimator.

Method		Quality metrics: Mean rate log Hz (st. dev.)	Quality metrics: Est. rescale rate (R^2)	Quality metrics: K-S distance	Normality measures: Kurtosis	Normality measures: Skewness	Normality measures: χ^2 test p-value	Normality measures: K-S test p-value
Histogram	slow	0.22 (0.65)	0.79 (0.52)	0.21	8.03	2.63	NA	$p \ll 0.01$ ($KS = 0.528$)
	fast	**1.95 (1.15)**	**1.08 (0.99)**	**0.02**	2.27	-0.77	NA	$p \ll 0.01$ ($KS = 0.300$)
KDE plug-in	slow	**0.58 (0.08)**	**0.92 (0.86)**	**0.11**	1.34	0.01	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.140$)
	fast	**2.43 (0.06)**	**1.08 (0.99)**	**0.02**	2.07	0.06	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.055$)
KDE mean shift	slow	**0.46 (0.34)**	**0.80 (0.78)**	**0.14**	6.56	1.52	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.144$)
	fast	1.77 (1.25)	1.26 (0.96)	0.06	1.57	-0.10	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.135$)
Regression spline	slow	**0.58 (0.07)**	**0.92 (0.85)**	**0.11**	1.90	0.45	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.101$)
	fast	**2.45 (0.00)**	**1.10 (0.99)**	**0.02**	3.25	-0.65	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.092$)
Exponential	slow	**0.53 (0.28)**	**0.93 (0.90)**	**0.10**	2.74	0.40	$p \ll 0.01$	$p < 0.05$ ($KS = 0.034$)
	fast	**2.13 (0.84)**	**0.86 (0.96)**	**0.05**	2.61	-0.61	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.083$)
Gamma	slow	**0.54 (0.27)**	**0.96 (0.91)**	**0.09**	2.81	0.35	$p \ll 0.01$	$p < 0.01$ ($KS = 0.046$)
	fast	2.13 (0.84)	0.86 (0.96)	0.05	2.55	-0.54	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.072$)
Inverse Gaussian	slow	**0.49 (0.24)**	**0.85 (0.87)**	**0.10**	2.74	0.40	$p \ll 0.01$	$p < 0.05$ ($KS = 0.034$)
	fast	2.05 (0.53)	0.72 (0.98)	0.03	2.54	-0.33	$p \ll 0.01$	$p < 0.01$ ($KS = 0.044$)
ISI-KDE	slow	**0.52 (0.25)**	**0.94 (0.94)**	**0.07**	2.78	0.26	$p \ll 0.01$	$p < 0.01$ ($KS = 0.048$)
	fast	2.10 (0.82)	0.83 (0.96)	0.05	2.56	-0.58	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.085$)
Adaptive exponential	slow	**1.35 (0.18)**	**3.41 (0.92)**	**0.08**	**2.95**	**-0.09**	$\mathbf{p > 0.05}$	$\mathbf{p > 0.06}$ ($KS = 0.034$)
	fast	**1.92 (0.28)**	**0.60 (0.99)**	**0.02**	**2.93**	**-0.24**	$\mathbf{p > 0.26}$	$\mathbf{p > 0.07}$ ($KS = 0.029$)

Table 4.1: Statistical summary of firing rate estimators for simulated data. Slow spike train mean rate is 0.69 log Hz and fast spike train mean rate is 2.40 log Hz. Kurtosis and skewness for a normal distribution are 3.0 and 0.00 respectively. Quality metrics in bold for models which are statistically valid at all model quantiles. Normality measures in bold for models that reject the alternative hypothesis of normally distributed firing rates.

4.4.2 Real data

The direct estimates of firing rate for the low and high frequency real spike trains with spike rasters are shown in Figure 4·6. The rate histogram with 150ms bin width is shown in gray with the remaining three direct rate estimators in blue, red and green (static KDE, variable KDE and regression splines). Model fits of the ISI distribution can be found in Figure 4·7 with corresponding ISI-convolution rate estimates in Figure 4·8. As opposed to the simulated spike trains discussed in the previous section, the real spike trains cannot be assumed to be homogeneous and Poisson; at the very least they are non-stationary.

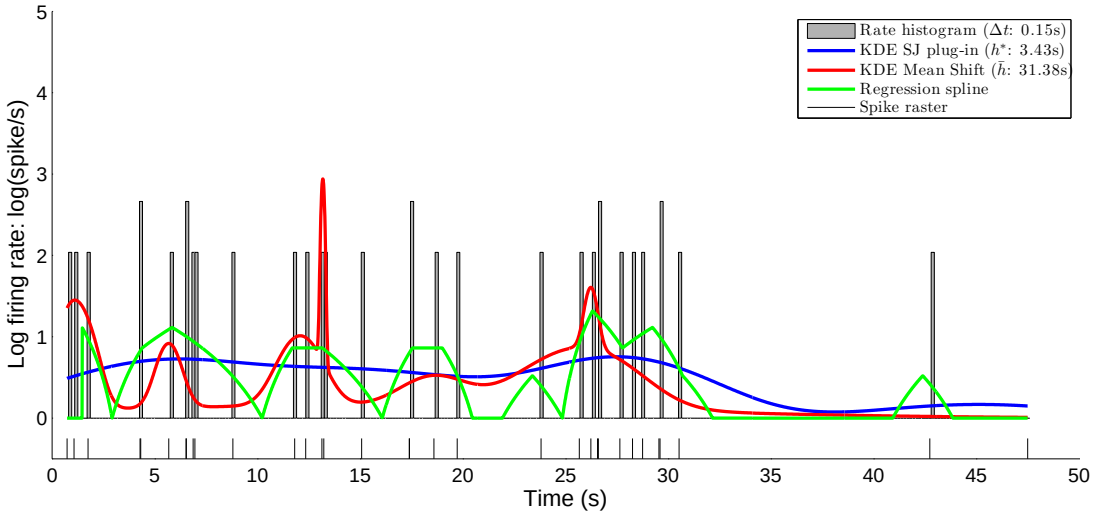
As in the case of the simulated data, the rate histogram for both slow and fast firing spike trains were highly variable and sparsely populated in the case of the slow spike train. The static bandwidth KDE stayed relatively constant, illustrating only slow changes in the baseline rate of the spike train, while both the adaptive bandwidth KDE and regression splines appear to capture local dynamics of the spike train. Similarly, the ISI-convolution filter methods all captured changes in the spike train local dynamics and all but the adaptive exponential method obtained similar mean firing rates. Interestingly, despite the differences in behavior between the four direct rate and five ISI-convolution filter methods, nearly all achieved similar mean rates (except for the adaptive exponential and rate histogram). As expected given results of the previous simulation study, the inverse Gaussian and KDE fits to the ISI distribution showed time shifts in the peak distribution estimate for the slow spike train and for the gamma as well for the fast spike train. The degree of shift in the slow spike train was much greater than in the fast spike train as in the latter, the estimated distribution peak was very close to $t = 0$ s. In addition, the KDE fit illustrated local increases in ISI probability as bumps, or multiple modes, at longer

ISIs. The primary peak shift is likely due to biological effects, such as refractory periods, while the modes found via the KDE distribution estimate may reflect other effects such as interactions between units or simply system noise.

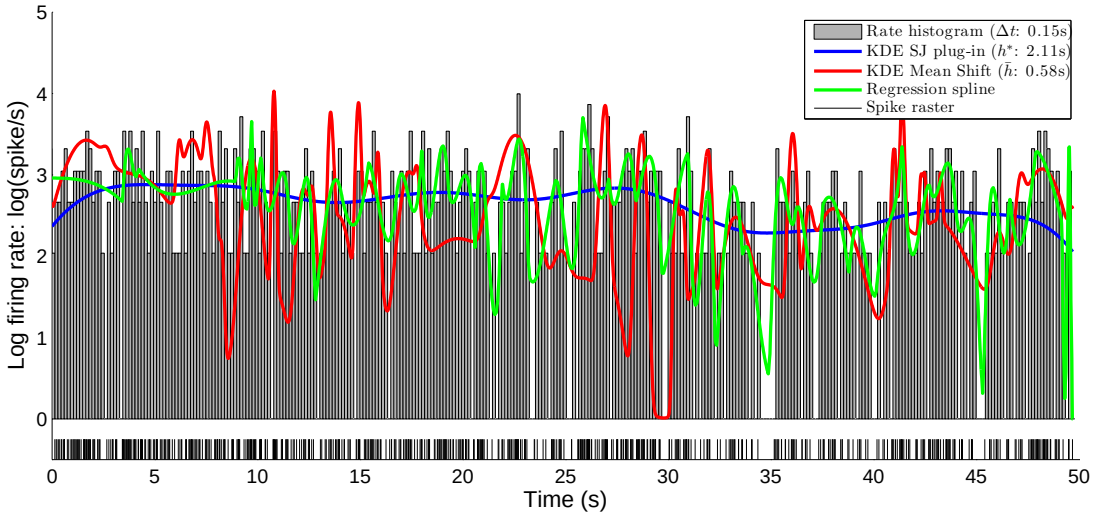
The modified K-S model quality test (see Figure 4·9) was applied to the real data to examine the goodness-of-fit for each rate estimation technique. All direct rate estimates fell within the 95% K-S boundary for the slow spike train though the rate histogram and regression splines tended to overpredict the correct rate at low quantiles (0-0.25) and slightly underpredict upper quantiles (approximately 0.75-1). The static and adaptive kernel density estimators were valid through all model quantiles (though the adaptive method briefly overpredicted the rescaled rate at very high quantiles) while the histogram and regression splines fit well for the first half quantile but underpredicted rate in the latter half. All ISI convolution filter methods were good fits to the slow spike train, and only the adaptive exponential was completely within the confidence bounds for the fast spike train. The exponential, gamma and KDE filter models for ISI-convolution all deviated significantly at upper quantiles (>0.67).

As in the case of the simulated spike trains, the χ^2 and one sample Kolmogorov-Smirnov tests for normality along with estimates of kurtosis and skewness were used to determine the degree of normality achieved by each rate estimation technique. These measures, shown in Table 4.2, indicated that only the adaptive exponential method, applied to the fast spike train, obtained normally distributed firing rates. Graphically, this result is verified by the approximately Gaussian shape of firing rate counts shown in Figure 4·10(D) while the counts of firing rates in Figure 4·10(B) exhibit a bimodal distribution. The adaptive exponential model outperforms the other ISI convolution-filter models in terms of normality for the fast spike train, but that result is not as convincing for the slow spike train. Though not immediately

obvious from the remaining plots in Figure 4·10, Table 4.2 more thoroughly indicates that static exponential, inverse Gaussian and KDE ISI models approach normality but all still significantly reject the null hypothesis of a normal distribution of firing rates.

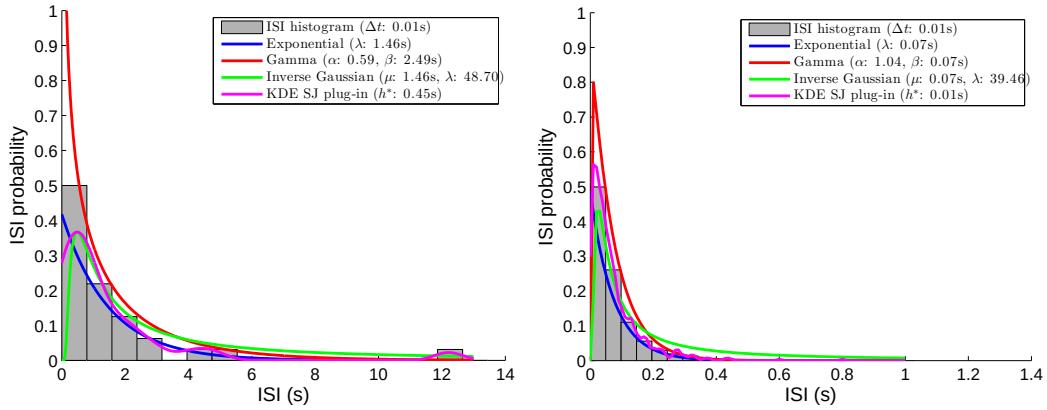


(A)



(B)

Figure 4·6: Direct rate estimates for a 50 second sample from example slow (top) and fast (bottom) real spike trains. Rate histogram is shown in gray, static KDE in blue, variable KDE in red and regression spline in green. Legend contains fitted model parameters.



(A) (B)

Figure 4·7: Model fits of the real data ISI distribution. The ISI histogram is shown in gray, exponential in blue, gamma in red, inverse Gaussian in green and ISI KDE in magenta. Estimates were taken from slow (left) and fast (right) real spike trains. The adaptive exponential is not explicitly shown but is characterized by the static rate exponential fit. Legend contains fitted model parameters.

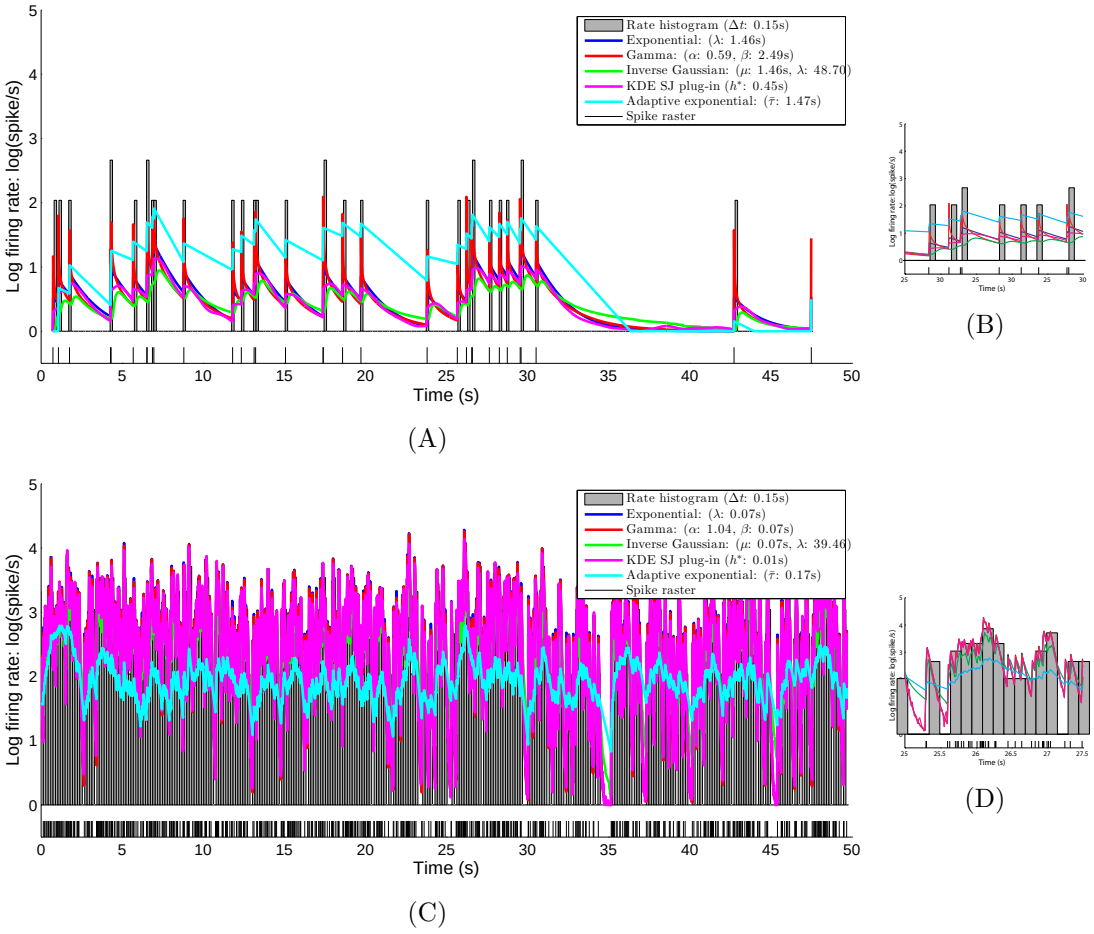


Figure 4·8: ISI-convolution rate estimates for a 50 second sample from example slow (top) and fast (bottom) real spike trains. (Left) Rate histogram in gray, static exponential in blue, gamma in red, inverse Gaussian in green ISI KDE in magenta and adaptive exponential in cyan. Legend contains fitted model parameters. (Right) Zoomed in portions of the estimated rate illustrating the dynamics of each estimator.

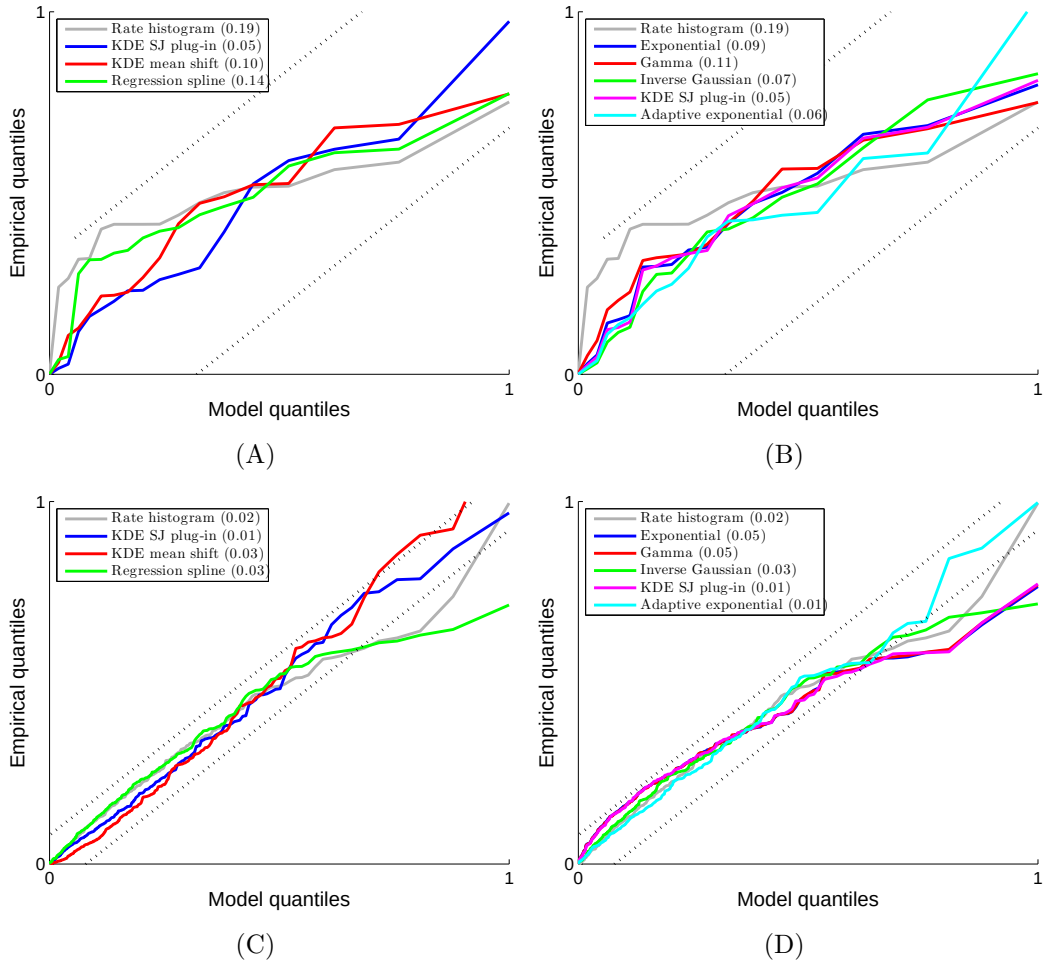


Figure 4·9: K-S plot of modified time-rescaled conditional intensities vs. the exponential. Valid models correspond to points on the 45° line. Dotted lines are 95% confidence intervals based on the K-S statistic. (Left) Direct rate models. (Right) ISI convolution methods. (Top) Recorded example slow firing units. (Bottom) Recorded example fast firing units. Legend contains RMSE of plotted points from the diagonal.

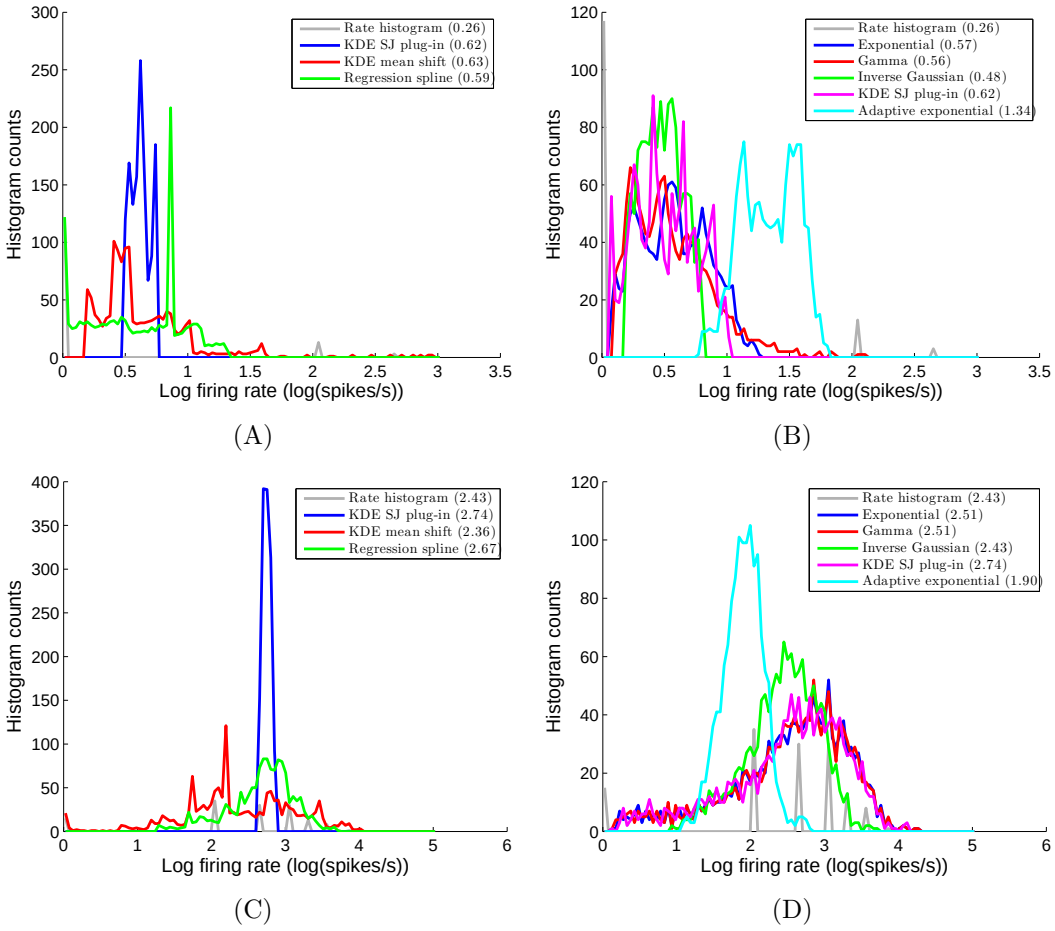


Figure 4·10: Histograms of real spike train firing rate distributions for each model. (Left) direct rate models. (Right) ISI convolution methods. (Top) slow firing units. (Bottom) fast firing units. Legend contains mean firing rate for each estimator.

Method		Quality metrics: Mean rate log Hz (st. dev.)	Quality metrics: Est. rescale rate (R^2)	Quality metrics: K-S distance	Normality measures: Kurtosis	Normality measures: Skewness	Normality measures: χ^2 test p-value	Normality measures: K-S test p-value
Histogram	slow	**0.26 (0.71)**	**0.84 (0.77)**	**0.19**	8.03	2.63	NA	$p \ll 0.01$ ($KS = 0.522$)
	fast	2.43 (1.00)	0.96 (0.98)	0.03	2.27	-0.77	NA	$p \ll 0.01$ ($KS = 0.236$)
KDE plug-in	slow	**0.62 (0.08)**	**1.03 (0.97)**	**0.05**	1.34	0.01	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.068$)
	fast	**2.74 (0.05)**	**1.00 (0.99)**	**0.01**	2.07	0.06	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.064$)
KDE mean shift	slow	**0.63 (0.38)**	**1.21 (0.92)**	**0.10**	6.56	1.52	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.150$)
	fast	2.36 (0.77)	0.96 (0.99)	0.03	1.57	-0.10	$p \ll 0.01$	$p < 0.01$ ($KS = 0.062$)
Regression spline	slow	**0.58 (0.37)**	**0.92 (0.83)**	**0.14**	1.90	0.45	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.126$)
	fast	2.67 (0.44)	0.93 (0.97)	0.03	3.25	-0.65	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.094$)
Exponential	slow	**0.57 (0.27)**	**0.92 (0.93)**	**0.09**	2.12	0.05	$p \ll 0.01$	$p < 0.01$ ($KS = 0.054$)
	fast	2.51 (0.80)	0.85 (0.96)	0.06	3.34	-0.84	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.081$)
Gamma	slow	**0.54 (0.27)**	**0.88 (0.90)**	**0.11**	4.65	1.07	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.075$)
	fast	2.13 (0.84)	0.86 (0.97)	0.05	3.39	-0.86	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.085$)
Inverse Gaussian	slow	**0.51 (0.25)**	**0.73 (0.95)**	**0.07**	2.07	0.10	$p \ll 0.01$	$p < 0.01$ ($KS = 0.045$)
	fast	2.48 (0.79)	0.72 (0.98)	0.03	2.99	-0.48	$p \ll 0.01$	$p < 0.01$ ($KS = 0.050$)
ISI-KDE	slow	**0.48 (0.16)**	**0.79 (0.95)**	**0.08**	2.01	0.10	$p \ll 0.01$	$p < 0.01$ ($KS = 0.059$)
	fast	2.48 (0.49)	0.83 (0.96)	0.05	3.18	-0.80	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.092$)
Adaptive exponential	slow	**1.34 (0.23)**	**3.43 (0.96)**	**0.06**	2.07	-0.17	$p \ll 0.01$	$p \ll 0.01$ ($KS = 0.084$)
	fast	**1.90 (0.27)**	**0.39 (0.99)**	**0.01**	**3.30**	**-0.03**	$\mathbf{p > 0.90}$	$\mathbf{p > 0.60}$ ($KS = 0.020$)

Table 4.2: Statistical summary of firing rate estimators for real data. Kurtosis and skewness for a normal distribution are 3.0 and 0.00 respectively. Quality metrics in bold for models which are statistically valid at all model quantiles. Normality measures in bold for models that reject the alternative hypothesis of normally distributed firing rates.

4.4.3 Comparison of neural activity to other studies

General analysis of the neural activity from the ventral precentral gyrus in this study revealed two broad classes of single units, the first (N∼35) had mean firing rates (i.e. constant rate) approximately between 1 and 10 spikes per second (Hz) and the second (N∼21) with mean rates between 10 and 80 Hz. The lower firing units had local maxima ranging 10-30 Hz while faster (>10 Hz mean rate) units exhibited local maxima between 70-80 Hz. These firing rates seem to be typical of those recorded elsewhere in human motor cortex electrophysiological study (i.e. Hochberg et al., 2006), though not of early primate studies (c.f. Georgopoulos et al., 1982) which were characterized by average firing rates of hundreds of neurons near 40 Hz and local maxima (i.e. task dependent activation) of more than 150 Hz. In these studies, cells were carefully identified using microelectrodes capable of being moved farther into the cortex, or removed entirely and replaced elsewhere in order to obtain the strongest, most salient action potentials. Further investigation of recent human and primate motor cortical studies revealed that most studies have been focused on neural prosthesis development; therefore, electrodes were implanted without precise localization of electrode tips near actively firing neurons, as is the case in human applications. In these studies, firing rates were, on average, similar to what was observed in the current study of human motor cortex utilizing the Neurotrophic Electrode. Specifically, primate and human firing rates were found to range between 0-60 Hz with mean rates laying in the 5-20 Hz range. Primate firing rates appear to be generally higher than their human counterparts. Table 4.3 provides a summary of reported firing rates for a number of motor cortical studies.

Study	Species	Electrode	Units per subject	Mean activity (Hz)	Activity range (Hz)
Georgopoulos et al. (1982)	primate	Single	151	30	0–190
Schwartz et al. (1988)	primate	Single	284	40	0–130
Kalaska et al. (1989)	primate	Single	52	20	0–140
Moran and Schwartz (1999a,b)	primate	Single	241	NA	0–150
Shenoy et al. (2003)	primate	Microwire	29	NA	0–80
Paninski et al. (2004)	primate	Array	40	5–10	0–40
Suner et al. (2005)	primate	Array	43	1–10	0–60
Hochberg et al. (2006)	human	Array	53	1–10	0–30
Kennedy et al., (submitted) *this dissertation*	human	Neurotrophic Electrode	56	1–10	0–80

Table 4.3: Summary of approximate firing rate means and ranges from extracellular recordings in the motor cortex using both single electrodes and microelectrode arrays in humans and primates. In this context, single electrodes (and microwires) are those capable of precisely recording from single neurons. Electrode array and the Neurotrophic Electrode were placed with less specificity as is the case in real BCI applications. All data is approximate and taken from stated or plotted values in the cited works.

4.5 Discussion

The conditional intensity function of stochastic point processes was used to estimate the instantaneous firing rate of simulated and real neural spike trains. Comparison of modified standard goodness-of-fit measures of nine methods for estimation of the conditional intensity of a simulated homogeneous Poisson spike train illustrated that only the static KDE, regression spline and adaptive exponential were statistically valid for all distribution quantiles. The static bandwidth KDE and regression spline methods for direct rate estimation were shown to be excellent models of homogeneous Poisson behavior as the predicted firing rates were nearly constant for the entire analysis window, as is expected from a stationary signal. The adaptive bandwidth KDE and ISI-convolution filter methods found captured local dynamics of the spike train, but only the ISI-convolution methods maintained stable firing rate estimates. The adaptive exponential and inverse Gaussian ISI-convolution methods were the best estimators of their class despite the inverse Gaussian partially failing the K-S GOF test. The same general result was obtained after analysis of real spike train data. Only the static KDE and adaptive exponential estimators proved to be valid for all rescaled distribution quantiles though the variable KDE and inverse Gaussian only partially failed. The adaptive bandwidth KDE and regression spline methods both found deviations from local constancy in the slow and fast spike trains while the static KDE maintained fairly constant estimates of firing rate with only slight firing rate modulation.

It is interesting to note that not all models yielded unit rate time-rescaled conditional intensities as expected from the time-rescaling theorem. Most obtained rescaled homogeneous rates *close* to one (i.e. ± 0.1) but many deviated even further from unity. In particular the variable KDE, inverse Gaussian, ISI KDE and adaptive exponential

all deviated greater that ± 0.1 from 1.0. Of these, the adaptive exponential was furthest from unity, with estimated rescaled rates of 3.43 and 0.39 for real data. These estimated rates reflect the effect of normalization built in to the adaptive exponential rate estimator. In particular, under low rate scenarios, the adaptive exponential "boosts" estimated firing rates toward a mean rate of $\frac{1}{\varepsilon}$, or in this case 2 log Hz. Similarly, under high rate conditions, the estimated rate is reduced in the opposite direction. The effect is characterized by the relatively high rescaled slow and low rescaled fast exponential rate parameters.

Model fit, however, is not the only qualification of a good spike rate estimator. Normality of the distribution of estimated firing rates is extremely important since many methods for neural decoding (Chapter 5) assume that observations are normally distributed. As the analyses in this chapter make clear, only the adaptive exponential ISI-convolution filter method is capable, even in the best scenario, of estimation of normally distributed firing rates. According to the results of χ^2 and K-S tests for normality and general inspection of normality measures, kurtosis and skewness, the adaptive exponential method when applied to both simulated and real spike trains yielded the desired result while none of the other methods were able to confirm a normal distribution of estimated firing rates. Analysis of the real spike trains was somewhat more ambiguous as the firing rates obtained by adaptive exponential ISI convolution for the fast spike train exhibited a normal distribution, though not for the slow spike train. Firing rate estimation of the slow firing rate condition was bimodally distributed for the adaptive exponential, though modulated by a Gaussian "envelope." The two modes represent the two major firing rate states of the slow spike train, namely single spike response and grouped spike response. Since the average interspike interval was quite long, all methods tended toward zero after every spike event. However, when there were multiple spikes their contributions were com-

bined yielding a higher expected firing frequency. Such "bursting" events were rare, but frequent enough to be captured by the adaptive exponential as it allows for slight changes in expected ISI. The adaptive exponential model can be tuned for this behavior by adjusting the learning rate parameter, ε. Currently, the same parameter value is used for all units though not all units fire with the same baseline rate. It is clear from these results that the learning parameters are near optimal for fast firing units but not for those with low firing frequency. Additional investigations are needed to estimate the optimal value of ε for slow firing units.

For practical use in a BCI capable of real-time performance, two important factors must be taken into account. First, the rate estimation method must be fast. Every method considered, except the regression splines, can be rewritten as a convolution of a window filter with the spike train, so it is possible to take advantage of fast convolution methods. The computational resources needed to compute spline basis coefficients is relatively low, so spline based methods are at least theoretically feasible for real-time use. However, the mean shift adaptive KDE, though a Gaussian convolution by definition, cannot operate in real time due to the computational time needed to perform the mode finding step in mean shift trajectory calculation for bandwidth acquisition. There are likely differences in computing time for calculating the different windows (i.e. Gaussian, rectangular, exponential, etc) though these differences should be minimal as efficient methods exist for their numerical solution.

The second BCI factor restricts any method from using future information as real-time systems do not have access to future time events. Immediately this eliminates both the KDE methods for direct rate estimation as the Gaussian window spans all past *and* future times. The only methods capable of solely operating on past events are regression splines and the ISI-convolution filter methods as each window, even the ISI KDE, are causal filters. The four static methods, exponential, gamma, inverse

Gaussian and ISI KDE must be trained offline and their shapes are held constant which, though computationally efficient, will not be able to reflect changes in spiking behavior of a neural unit over time. Only an adaptive method is capable of learning new firing characteristics of neural units as they evolve over time. Currently, the only adaptive method available that provides normally distributed firing rates, has minimal computational time, is causal and is capable of learning new firing preferences is the adaptive exponential ISI-convolution filter. Additional adaptive estimators may be developed in the same class as the adaptive exponential and will be the subject of future work.

4.6 Conclusion

The instantaneous firing rate of single neural units was modeled as the conditional intensity function of stochastic point processes. Under this framework it was possible to identify two main types of estimators, direct rate and ISI-convolution filters. A variety of rate estimators were applied to both simulated and real spike data and were subjected to tests for both GOF and normality. The condition of Gaussian distributed firing rates was introduced and its importance to subsequent analysis made clear. Additional constraints were discussed that are unique to real-time systems. Of all methods tested, only one met all conditions necessary for the BCI design considered in this dissertation. The adaptive exponential ISI-convolution filter yielded statistically valid conditional intensities, was capable of producing a normal distribution of firing rates, compute updated rate estimates with computational overhead under one millisecond and is causal, meaning it relies only on past events at each timestep. Future study will delve into the relationship between the adaptive exponential method and the class of penalized linear splines. This type of regression spline

is quite flexible and will likely provide the best avenue for possible improvements to the adaptive exponential estimator.

CHAPTER 5

NEURAL DECODING

5.1 Introduction

The purpose of this dissertation research is to construct a neural prosthesis for direct control of a real-time formant-based speech synthesizer (see Chapter 2) by a locked-in patient. In general, such control requires spike train decoding methods capable of making predictions about the system state (e.g. speech signal) given a set of noisy observations (e.g. neural activity). While prediction of speech signals has never been attempted before, a straightforward analogy can be made between the current attempt and previous decoding efforts of the motor system by other researchers. For example, a simplified look at a typical motor action, hand grasping, involves activation of the motor cortex, transmission of information to the periphery and stimulation of appropriate groups of muscles to execute the particular grasp. Similarly, to produce a simple speech sound (i.e. the vowel /A/) involves activation of the motor cortex specifying the vocal tract configuration (i.e. muscles[1]) needed to form an acoustic filter that provides spectral shape to a source waveform provided by excitation of the voluntary respiration system (i.e. more muscles[2]).

Previous research has explored the relationships between end-effector behavior and motor cortex activity. In a seminal work, Georgopoulos and colleagues found a simple relationship between activation of individual motor cortex neurons and hand move-

[1] Vocal tract configuration is taken as mostly tongue position and jaw height.
[2] Airflow through the larynx is achieved by voluntary control of the diaphragm and glottis.

ment directions made by a monkey moving a manipulandum on a 2D plane. Specifically, these neurons exhibited a cosine tuning curve in which they fired maximally for preferred movement directions and minimally for anti-preferred (Georgopoulos et al., 1982, 1986). From this relationship, they were able to predict observed movement direction from linear combinations of the firing rates for each recorded neuron modulated by their preferred directions. This type of analysis formed an estimate of system state from an ensemble of neurons or *population code*. The vast majority of motor system neural decoding techniques attempt to predict external behavior or system state from populations of single units. For instance, the previously mentioned work by Georgopoulos and colleagues, the population vector algorithm (PVA; Georgopoulos et al., 1982, 1986; Moran and Schwartz, 1999a,1999b; Taylor et al., 2002; Velliste et al., 2008), has been used to predict arm and hand kinematics, as have optimal linear filters (Wessberg et al., 2000; Carmena et al., 2003; Paninski et al., 2004; Hochberg et al., 2006), maximum likelihood methods (ML; Truccolo et al., 2005) and Bayesian inference (Kalman filters (Wu et al., 2003, 2006; Kim et al., 2007) and Particle filters (Brockwell et al., 2004)).

This chapter will focus on detailing the methodology of three decoding techniques: the PVA, optimal linear filter and Bayesian inference. The effectiveness of each method toward the research problem of this dissertation will be discussed and some data analysis presented.

5.2 Population decoders

The PVA (Georgopoulos et al., 1986) and optimal linear filter (Warland et al., 1997) are examples of linear methods for direct state decoding from neural spike trains. Both methods obtain estimates of external states by linear combinations of each

unit's instantaneous firing rates weighted by their preferred state. However, neither method explicitly takes advantage, or is constrained by, previous neural and external states nor knowledge of external states in general, although some amount of temporal dependence is implicitly included by the linear filter method. Alternatively, Bayesian inference methods (Wu et al., 2003; Brockwell et al., 2004; Truccolo et al., 2005; Wu et al., 2006; Kim et al., 2007) estimate the *a posteriori* probability of system state conditioned on both measurements (i.e. neural firing rates) *and* external state evolution (i.e. statistical prior). The Kalman filter is a special case of Bayesian inference in which the posterior probability is obtained by solution of Bayes Law with likelihood, or encoding model, and prior probability distributions approximated by linear Gaussian systems (Roweis and Ghahramani, 1999; Wu et al., 2006). In addition, recent studies have shown Bayesian inference and linear filtering to be more accurate than the PVA (Brockwell et al., 2004; Wu et al., 2006) and to be capable of real-time performance in movement restricted patients (Hochberg et al., 2006; Kim et al., 2007).

5.2.1 Population Vector Analysis

The population vector algorithm (PVA) developed by Georgopoulos and colleagues (Georgopoulos et al., 1982, 1986) was one of the earliest proven methods for accurate prediction of monkey arm kinematics. In its classic formulation arm movement direction is predicted from the neural activity among populations of primary motor cortex neurons. First, directional tuning curves are obtained for each of the recorded neurons,

$$D(t) = b_0 + b_x \sin\theta(t) + b_y \cos\theta(t) \tag{5.1}$$

where D is the instantaneous firing rate associated with a movement in direction θ. The coefficients $b = (b_0, b_x, b_y)$ are obtained via sinusoidal regression.[3] The preferred direction of each cell is encoded in regression coefficients b_x and b_y while the baseline activity is characterized by b_0. The directional population vector, PV, is then calculated,

$$PV_x = \sum_{i=1}^{n} \frac{D_i(t) - A}{M} \cdot \frac{b_{x,i}}{M} \tag{5.2}$$

where $b_{x,i}$ is the horizontal component of the preferred direction of cell i, A is a centering parameter (typically set to b_0) and M is a rate normalizing factor[4]. This method is popular because of its simplicity; only binned firing rates (i.e. PSTH) and observed (but uniformly distributed) kinematic states during a training period are needed to predict future states.

The earliest formulations of the PVA were concerned only with movement direction; however, alternative versions have investigated the differential response for the magnitude (i.e. speed) and direction components of the population vector (Moran and Schwartz, 1999b), position encoding (Kettner et al., 1988) and torque encoding (Kalaska et al., 1989). All three types of kinematic models (direction, velocity and position) are capable of fitting behavioral data, but it is likely that a combination of position and velocity (and perhaps other movement kinematics derivatives) will provide the most accurate results (Wu et al., 2003, 2006).

5.2.2 Linear filter

An alternative linear method to the PVA is the linear filter (Warland et al., 1997), also termed reverse regression or reverse correlation (Kass et al., 2005). Though Warland et al. (1997) first used the convolution of an optimal linear filter with discretized

[3]This is related to standard linear regression if the x and y components of Eq. 5.1 are taken in Cartesian coordinates and unit-normalized rather than polar.

[4]For example if $A = b_0$, then $M = D_{\max} - D_{\min}$ (Georgopoulos et al., 1988, Method 10)

firing rates to reconstruct a visual stimulus from tiger salamander retinal ganglion cell activation, the method has since been adopted by other researchers for decoding of monkey and human hand kinematics from motor cortex activity (Wessberg et al., 2000; Serruya et al., 2002; Carmena et al., 2003; Paninski et al., 2004; Hochberg et al., 2006). In either case, the method is the same and begins with zero-mean, unit-variance normalization of the stimulus and construction of the response matrix, $\mathbf{R}$, consisting of the firing rates for each cell in a time interval specified by the desired length of the linear filter,

$$\mathbf{R} = \begin{bmatrix} 1 & r_0^1 & r_1^1 & \cdots & r_{N-1}^1 & \cdots & r_0^\nu & \cdots & r_{N-1}^\nu \\ 1 & r_1^1 & r_2^1 & \cdots & r_N^1 & \cdots & r_1^\nu & \cdots & r_N^\nu \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & r_{M-1}^1 & r_M^1 & \cdots & r_{N+M-2}^1 & \cdots & r_{M-1}^\nu & \cdots & r_{N+M-2}^\nu \end{bmatrix}$$

where N is the length of the filter, M is the length of the stimulus and r_i^ν are the firing rates of cell ν at time i. The filter coefficients are obtained via multiple linear regression,

$$\mathbf{f} = (\mathbf{R}^\mathsf{T}\mathbf{R})^{-1}\mathbf{R}^\mathsf{T}\mathbf{s} \tag{5.3}$$

$$\mathbf{f} = [a, f_0^1, f_1^1, \cdots, f_{N-1}^1, \cdots, f_0^\nu, \cdots, f_{N-1}^\nu]^\mathsf{T} \tag{5.4}$$

$$\mathbf{s} = [s_0, s_1, \cdots, s_{M-1}]^\mathsf{T} \tag{5.5}$$

where $\mathbf{s}$ is the stimulus, a is an offset term and $\mathbf{f}$ are the filter coefficients. The reconstructed stimulus, $\mathbf{u}$, is obtained according to,

$$\mathbf{u} = \mathbf{R} \cdot \mathbf{f}. \tag{5.6}$$

Decomposition of Equation 5.6, results in Equation 5.7, clearly illustrating the reconstruction, $\mathbf{u}$, is a result of summed convolution of the fitted filter with each cell's response.

$$u_i = a + \sum_{v} \sum_{j=0}^{N-1} r_{i-j}^{\nu} f_j^{\nu} \tag{5.7}$$

More generally, the optimal linear filter is a discrete Wiener filter (Gelb, 1974) where each prediction u_i is a linear function of the neural activity r_i^v for all cells over the last N time points (Wu et al., 2006). Specifically, a Wiener filter is a causal linear time-invariant (LTI) filter designed to remove the effects of additive noise from a linear stochastic signal. The optimal filter is found by minimization of residual error between the original signal and its noise-corrupted Wiener filtered reconstruction. Correspondingly, natural applications of the Wiener filter are denoising of audio and visual signals. In the context of neural decoding, one assumes that linear combinations of neural firing rates are simply noise-corrupted versions of the system state (i.e. stimulus). In this way, a filter can be constructed that optimally denoises the ensemble of firing rates.

5.2.3 Bayesian inference

Bayesian inference offers a powerful statistical framework for spike train decoding of a number of modalities, such as arm velocity in a center out task (Brockwell et al., 2004; Kim et al., 2007), pursuit tracking task (Truccolo et al., 2005; Wu et al., 2006) or representation of spatial position (Brown et al., 1998). In the context of neural decoding, Bayes theorem is used to estimate the the *a posteriori* probability of the system state (i.e. external stimulus: hand kinematics, spatial location, etc.) conditioned on spike activity given a probability model of the observed data obtained from training data. For example, Kass et al. (2005) describes a Bayesian inference

formulation of velocity prediction,

$$p(v_t|y_1, \cdots, y_t) \propto p(y_t|v_t)p(v_t|y_1, \cdots, y_{t-1}) \tag{5.8}$$

where the probability of observing a velocity v_t at time t is proportional to the product of the observed velocity-dependent firing rate, $p(y_t|v_t)$, and the velocity state prior probability distribution, $p(v_t|y_1, \cdots, y_{t-1})$. Estimation of the posterior density requires the solution of the unknown velocity state prior probability given by,

$$p(v_t|y_1, \cdots, y_{t-1}) = \int p(v_t|v_{t-1})p(v_{t-1}|y_1, \cdots, y_{t-1})dv_{t-1}, \tag{5.9}$$

where $p(v_t|v_{t-1})$[5] describes the probability of observing a velocity, v_t given previous velocity state, v_{t-1}, and $p(v_{t-1}|y_1, \cdots, y_{t-1})$ is the previously estimated posterior probability estimate from Equation 5.8. Equation 5.9 provides a convenient recursive solution to Equation 5.8 and has been solved using a number of techniques including a Bayes filter (Brown et al., 1998; Truccolo et al., 2005), Kalman filter (Wu et al., 2002, 2003, 2006; Kim et al., 2007) and Particle filter (Gao et al., 2002, 2003; Brockwell et al., 2004).

Kalman filter

In general, the Kalman filter (Kalman, 1960) is a member of the class of linear Gaussian models (LGM). That is, both the state and observation processes are assumed to be linear corrupted with additive zero-mean Gaussian noise. Further, the state model is assumed to be a first-order Gauss-Markov random process, meaning the state dynamics are linearly related (Roweis and Ghahramani, 1999). This relationship of

[5]Also known as the state transition probability.

observations, $\mathbf{z}_t$, and states, $\mathbf{x}_t$ is formalized by,

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathcal{N}(\mathbf{0}, \mathbf{Q}) \tag{5.10}$$

$$\mathbf{z}_t = \mathbf{C}\mathbf{x}_t + \mathcal{N}(\mathbf{0}, \mathbf{R}) \tag{5.11}$$

where the future states and observations are functions of the current state and state transition matrix, $\mathbf{A}$, or generative matrix, $\mathbf{C}$, respectively.[6] In addition, the LGM framework formulates the generative state dynamics as an "informative lower dimensional projection or explanation of the complicated observation sequence" (Roweis and Ghahramani, 1999). This property of LG models is uniquely suited to the problem of population decoding of neural activity where we are faced with the daunting task of interpreting the millisecond-by-millisecond activation of dozens of individual neurons. Effectively, the dimension reduction obtained via an LGM-framework results in the desired neural decoding.

Wu et al. (2002, 2003) first proposed the use of a Kalman filter to decode arm activity from over 40 neurons in monkey primary motor cortex into computer cursor control. They sought to develop a real-time method[7] that, among other properties, had a "sound probabilistic foundation" and incorporated a model of the noise inherent in the observations (Wu et al., 2003). Under the Kalman filter framework, the problem of neural decoding becomes one of inferring behavior (or world state) from noisy observations and prior knowledge about the *evolution* of world states (Wu et al., 2003). A similar approach is taken with modern speech recognition algorithms which employ hidden Markov model (HMM) techniques to predict speech units (e.g. words, phonemes, etc.) given noisy observations (i.e. speech waveforms transmitted over the phone line) and an *a priori* knowledge of the frequency or probability with which

[6] $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$ is a multivariate normal distribution with mean $\mathbf{0}$ and covariance $\mathbf{\Sigma}$.
[7] Capable of making instantaneous estimates of cursor position in under 200ms

certain speech units follow others.[8]

In their application of the Kalman filter to motor cortical control, Wu et al. (2002, 2003, 2006) first make two simplifying assumptions about Equation 5.8:

1. World state (i.e. hand kinematics) has a first-order Markov dependence (the state at time t depends only on the state at time $t-1$)

2. Observation at time t, conditioned on current state, is independent from all other observations.

These assumptions lead to an alternative formulation of Equation 5.8,

$$p(\mathbf{x}_k|\mathbf{z}_k) = \mathcal{K}p(\mathbf{z}_k|\mathbf{x}_k)\int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{z}_{k-1})d\mathbf{x}_{k-1} \tag{5.12}$$

where $p(\mathbf{z}_k|\mathbf{x}_k)$ is the likelihood or generative model and $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ is the state temporal prior (or state transition), both of which are approximated by the LGM generative model equations (Eqns. 5.11 and 5.10). In this way, Wu et al. (2002, 2003, 2006) were able to accurately decode monkey hand kinematics[9] in a pursuit tracking task from primary motor cortex activation. More recently, Kim et al. (2007) used the Kalman filter decoder in a neural prosthesis application for cursor control in a human with tetraplegia. The subject was without voluntary use of the motor periphery and through training with the Kalman filter decoder, learned to perform a multi-state center out task.[10] The research for this dissertation is inspired by the Kim et al. (2007) results and attempts to reconstruct speech sound trajectories in a center-out task in the vowel formant space from motor cortical activity.

[8]Interestingly, HMMs also fall into the category of linear Gaussian models and are intimately related to the Kalman filter (Roweis and Ghahramani, 1999).

[9]Position, velocity and acceleration

[10]Multi-state control consisted of: (1) continuous movements from neutral position to one of eight radial targets, (2) a binary "click" once the subject had moved the cursor to the correct location.

Particle filter

More general than the Kalman filter, the particle filter (Gao et al., 2002, 2003; Brockwell et al., 2004) is not limited by assumptions of linear Gaussian model dynamics. Rather, the posterior distribution of world states (i.e. hand kinematics) is found by iteratively sampling the likelihood model in a Monte Carlo simulation and computing the new state via application of the simulation results to the state transition model. This technique is advantageous since it makes no assumption about the observation or state characteristics.[11] However, as it requires Monte Carlo simulations to obtain its final result, the method is far too slow to be considered in a real-time neural prosthesis.

5.3 Methods

Three methods described above (PVA, linear filter and Kalman filter) were available for use in the neural prosthesis for speech production. However, previous study indicated the PVA would not likely obtain significantly improved results compared to the linear and Kalman filters (Brockwell et al., 2004; Wu et al., 2006) and was not considered for final neural prosthesis application. The linear and Kalman filters were tested offline using real neural data recorded from S1 via the Neurotrophic Electrode implant.

5.3.1 Preliminary study

Initial offline analysis of neural data used the rate histogram (50 ms time bins) approach for spiking activity estimation and the linear filter for decoding. Neural recordings were obtained from 41 units (since the initial study, several new spike clusters

[11] The generality of the Particle filter is especially nice when it is not appropriate to enforce linear normality.

have been identified and distinguished from the rest) on the Neurotrophic Electrode implant during an open-loop paradigm described below. S1 participated in a single vowel repetition task where he was asked to listen to a vowel sound (/A/, /OO/ or /IY/) in isolation, then to repeat the sound after hearing a GO signal (REPETITION task). Each vowel sound was 500 ms in duration and S1 was given 3000 ms to provide his response. Formant frequencies were predicted during response period after the GO signal utilizing the linear filter method. Linear discriminant analysis was used to assess formant frequency prediction error at each time bin (three vowels yielded a chance error rate of 67%).

The first two formant frequencies were taken as the two-dimensional stimulus, $\mathbf{s}$ from Equation 5.3, to obtain linear filter coefficients. The effect of filter length was determined by systematically varying length from 1 to 21 (i.e. 0 ms to 1000 ms spiking history) and computation of training and validation error from a 10-fold cross-validation paradigm. This was accomplished by fitting a linear filter to $\frac{9}{10}$ of the data, and testing on the remaining tenth for each filter length 1 to 21. The optimal filter length was taken as the point at which the training and validation error were minimal. For both training and validation data sets, formant frequencies were predicted by simple convolution of the firing rates by the trained filter.

5.3.2 Later study

Results from the initial study were promising but led to less than optimal error rates. Therefore, the offline experimental paradigm was modified to increase decoder performance and to calibrate the real-time neural prosthesis. The analysis described below was performed at the start of every recording session in which real-time speech synthesis was attempted.

Neural recordings were obtained from 56 units on the Neurotrophic Electrode im-

plant during an open-loop paradigm in which S1 was required to mentally speak-along with an acoustic stimulus (SPEAK-ALONG task). The stimulus was a continuous center-out-and-back design with the UH sound taken as the center point and the three corner vowels (/A/, /OO/ and /IY/) taken as endpoints. In this task, a one minute auditory stimulus with structure UH-V-UH-V-UH-..., where V were eight repetitions of the corner vowels, was presented acoustically to the subject. A GO cue indicating the start of the trial preceded the auditory stimulus. S1 was instructed to speak along with the stimulus as he heard it.

Normalized firing rate was found using the adaptive exponential rate estimator (Chapter 4), at a sampling rate of 67 Hz, or 15 ms intervals. The target formant frequencies were assumed as S1's true production since there is no observable behavior due to his locked-in condition.[12] The neural recordings and target formant frequencies (see Figure 5·1) were aligned to the beginning of the stimulus presentation as the subject was instructed to speak along with the stimulus exactly as he heard it.

The Kalman filter neural decoder was used to predict the first two formant frequencies and their velocities. Under this decoding paradigm, the formants and velocities were taken as the four dimensional Kalman filter system state (i.e. position and velocity in arm kinematic studies). The likelihood and temporal prior matrices were constructed according to Wu et al. (2002, 2003, 2006) utilizing the full covariance matrices for observation and state noise. The likelihood model is found by the multiple linear regression:

$$\mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{q} \tag{5.13}$$

where $\mathbf{H}$ is the likelihood matrix, $\mathbf{z}$ are the estimated firing rates, $\mathbf{x}$ are the formant frequencies and velocities and $\mathbf{q}$ is a multivariate normal random variable with zero

[12]This is similar to the Training Cursor (TC) in the paradigm employed by Kim et al. (2007).

mean and $\mathbf{Q}$ covariance. Linear fits passing a statistical validity threshold, $p \leq 0.05$ (multiple linear regression), were selected for further analysis (c.f. Kalaska et al., 1989). Similarly the temporal prior or state evolution model is found by the multiple linear regression:

$$\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{w} \tag{5.14}$$

where $\mathbf{A}$ is the state evolution matrix, $\mathbf{x}_{t+1}$ are one-step future states (formants and velocities), $\mathbf{x}_t$ are the current formant frequencies and velocities and $\mathbf{w}$ is a multivariate normal random variable with zero mean and $\mathbf{W}$ covariance. Formant frequencies and their velocities were predicted utilizing the Kalman filter decoder according to the following algorithm:

	Prediction stage	
1.	$\mathbf{x}_{t+1\|t} = \mathbf{A}\mathbf{x}$	Prior predicted state
2.	$\mathbf{P}_{t+1\|t} = \mathbf{A}\mathbf{P}$	Prior predicted estimate covariance
	Update stage	
3.	$\mathbf{y} = \mathbf{z}_{t+1} - \mathbf{H}\mathbf{x}_{t+1\|t}$	Innovation measurement
4.	$\mathbf{S} = \mathbf{H}\mathbf{P}_{t+1\|t}\mathbf{H}^{\mathsf{T}} + \mathbf{W}$	Innovation covariance
5.	$\mathbf{K}_{t+1} = \mathbf{P}_{t+1\|t}\mathbf{H}^{\mathsf{T}}\mathbf{S}^{-1}$	Kalman gain
6.	$\mathbf{x}_{t+1} = \mathbf{x}_{t+1\|t} + \mathbf{K}_{t+1}\mathbf{y}$	Updated state estimate
7.	$\mathbf{P}_{t+1} = (\mathcal{I} - \mathbf{K}_{t+1}\mathbf{H})\mathbf{P}_{t+1\|t}$	Updated estimate covariance

Table 5.1: Kalman filter prediction algorithm.

Goodness-of-fit for the Kalman filter model was evaluated utilizing the generative model R^2 values and by correlation coefficient analysis of the predicted formant trajectories vs. the target trajectories. Wu et al. (2002) noted that the correlation coefficient can be a good measure of GOF since, in their case, the Kalman filter algorithm predictions were often of the correct "shape" but simply translated from the target. Both measures were compared against chance levels. Validation of the Kalman filter model consisted of a closed-loop real-time evaluation described in detail

in Chapter 6.

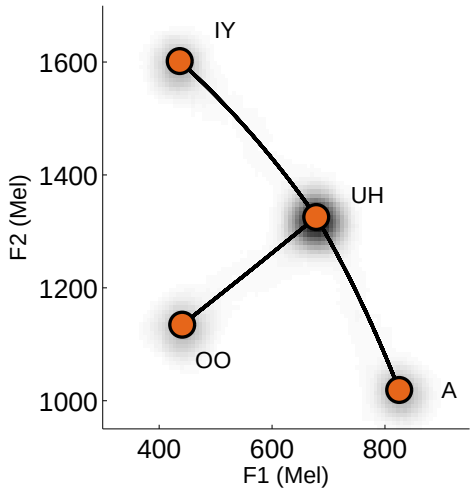


Figure 5·1: Target trajectories for training stimulus used in open-loop paradigm. The subject repeated each back-and-forth "leg" 8 times for a total of 63.4 s of production.

5.4 Results

5.4.1 Preliminary study

Filter coefficients for the linear filter method were estimated by multiple linear regression of the response matrix onto the training stimulus, with the number of filter coefficients systematically varied from one to twenty-one. Predicted formant frequencies were classified as one of the three vowel sounds /A/, /OO/ or /IY/ using linear discriminant analysis (LDA). Ten-fold cross-validation of vowel classification error was used to assess decoder performance. The data was divided into ten equal parts, trained on 9/10 and tested on the remaining tenth and repeated in order to test every left out portion. The per trial classification error was averaged across all training and validation sets and plotted for all filter lengths shown in Figure 5·2. In this figure, training error is shown in blue and validation error in red, both plotted with standard

error bars. It is immediately obvious that the training error monotonically decreases with filter length. Indeed, using a filter length equal to the length of the data set would yield zero classification error, however, the model would suffer from overfitting resulting in a very large validation error. Rather, the point with the lowest validation error is chosen as the optimal filter length. In this case, the optimal length is five coefficients, or 200 ms. This decoding study resulted in classification accuracy of approximately 60% for the training data predictins and near 45% for the cross-validated predictions. These results were supported by independent analysis of the same data using discrete classification methods. Miller et al. (2007) correctly classified 250-500ms windows of neural activity, with $\sim 80\%$ accuracy, utilizing a Bayesian analysis coupled with the ISOMAP[13] procedure for dimensionality reduction. A similar result of $\sim 75\%$ accuracy was achived by Wright et al. (2007) using a support vector machine neural network for discrete phoneme classification of 500 ms activity windows. Both results corroborate the findings of this decoding study and prove it is possible to decode speech from the neural responses recorded by the Neurotrophic Electrode implanted in the ventral precentral gyrus.

Later study

The Kalman filter likelihood and state model parameters were estimated by least squares linear regression using the entire set of training data. The system state was taken to be a four-dimensional vector of the first two formant frequencies and their velocities. The velocity components of the likelihood matrix, $\mathbf{H}$, are equivalent to preferred velocities (or directions) of the PVA and an example is shown in Figure 5·3 along with preferred positions, analogous to those found by Kettner et al. (1988). The estimated firing rate residual covariance, $\mathbf{Q}$, was found to contain values approx-

[13]ISOMAP is a nonlinear version of principal components analysis.

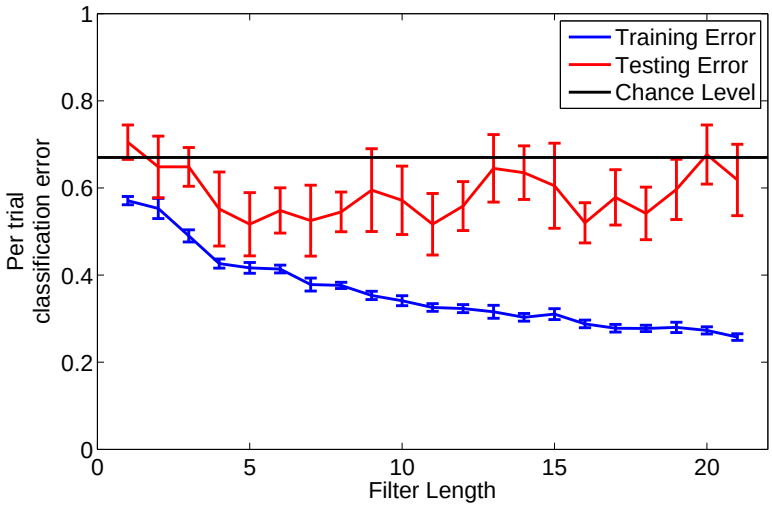


Figure 5·2: Training and cross-validation classification error for linear filter formant frequency predictions of initial offline decoding analysis. Training error shown in blue and cross-validation in red, both plotted with standard error bars. The chance level, 67% is shown in gray.

imately one order of magnitude smaller than the observed firing rate span.

The estimated state transition matrix approximated the identity matrix, though there were typically non-zero off-diagonal terms. This means that while no particular constraints were placed on state transitions[14] the Kalman filter learning procedure obtained a state evolution model where future states are equal to noise corrupted previous states. Off-diagonal terms additionally represent relationships due to the constrained state space (i.e. trajectories to and from the center vowel). The state transition matrix is expected to approximate the identity even further when more of the formant and velocity spaces are sampled during training. The residual covariance, $\mathbf{W}$, contained diagonal elements three orders of magnitude lower than the maximum span of formant frequency positions (i.e. 10^{-3}) but there was no difference for velocities.

A new model was fit at the beginning of most recording sessions over the span

[14]Brockwell et al. (2004) constrained velocities to the model $v_{t+1} = v_t + \epsilon$.

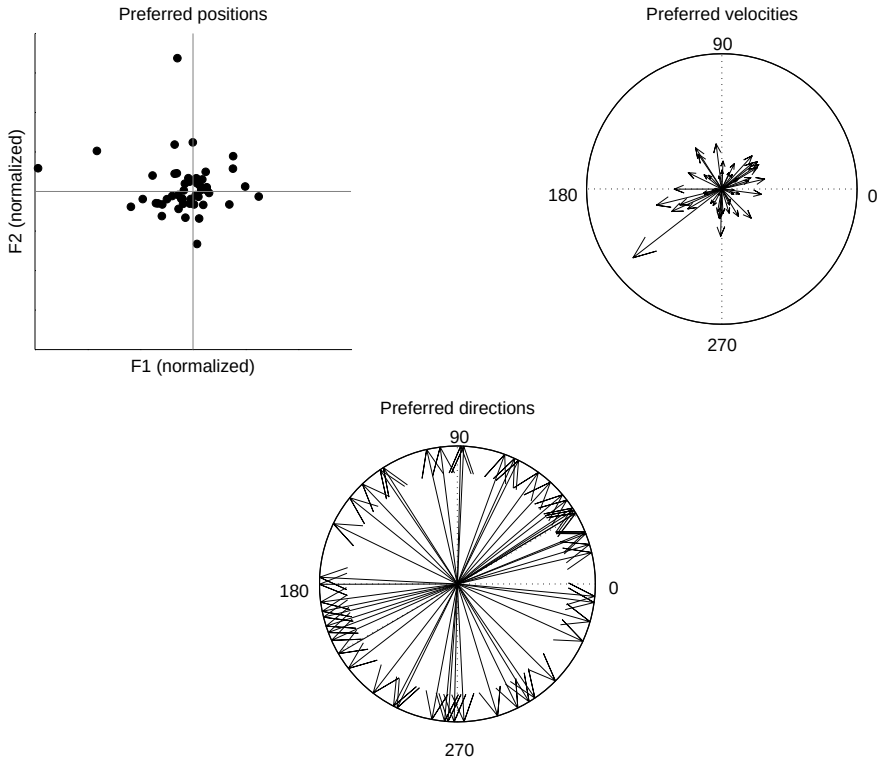


Figure 5·3: Preferred position, velocity and direction components of the likelihood model (generative matrix) for all neural units. Top left: preferred position. Top right: preferred velocity. Bottom: preferred direction.

of three months. Units were not considered for analysis on any particular day if the linear fit of formants and formant velocities on their firing rates was not significant ($p > 0.05$, multiple linear regression F-test) regardless of the R^2 value. Using this criteria approximately 51 units were chosen on average (see Table 5.2). The model fit R^2 ranged between 0.002 at minimum and 0.387 at maximum which falls within the range of values reported by Taylor et al. (2002). Fits of random data were performed to determine whether or not the model fits were simply due to chance. Normally distributed random noise with zero-mean and variance estimated from real data was fit by multiple linear regression. No noise controls were significant at $p < 0.05$ and all had R^2 approximately equal to zero. This evidence suggests the linear fits to real data are not simply due to chance.

Additional goodness of fit was determined by application of the previously fit linear models to the training data for each session. Formant frequencies and velocities were predicted and the correlation (R^2) of the predictions and training stimulus was obtained. These values were compared against those expected due to random input. In this case, the random data fits were permitted, despite the normal rejection criterion (i.e. $p > 0.05$ for the linear regression F-test). The random-fit models were applied to the random data for an estimate of the training prediction due to chance. The results for both data sets (real and random) are shown in Figure 5·4. In this figure, the real training data predictions are shown in blue and green (first and second formants respectively) for formant positions (circles) and formant velocities (triangles). The random data predictions are shown in black and gray (first and second formants respectively) utilizing the same markers for positions versus velocity. As indicated in the figure, the formant position correlations ($\bar{R}^2(F1, F2) = (0.56, 0.62)$) are much higher than chance. However, the velocity fits ($\bar{R}^2(F1, F2) = (0.16, 0.16)$) are much less so. In fact, according to these results it is not possible to distinguish

the velocity fits from random chance. Both results are illustrated in another fashion in Figure 5·5. In these figures the mean firing rates and linear fits found for example single units are shown with respect to 2D formant frequency position and velocity. These figures are typical for well fit single units. The linear fits for formant frequency position tend to agree with the observed values, as expected from the correlations previously found.[15] On the other hand, the firing rates with respect to velocity are certainly not linear. In fact, they appear to be symmetrically bimodal with increased activity observed along one velocity direction and its 180° reflection.

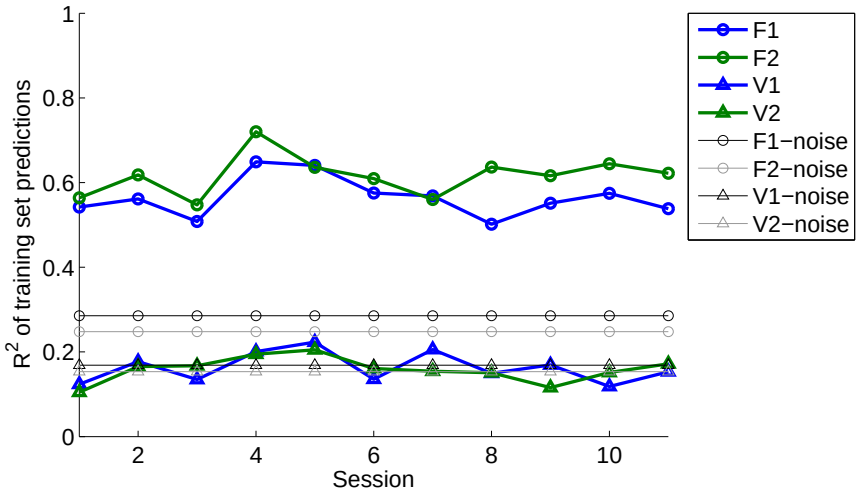


Figure 5·4: Open-loop training data prediction R^2 utilizing a Kalman filter decoder. Formant frequencies (circles) and velocities (triangles) are shown with the blue curve for the first formant and green for the second. Gray and black curves indicate the level expected due to chance for first and second formants using circles and triangles to indicate formant frequencies and velocities, respectively.

[15]This is seen more clearly between the linear fit figures and the empirical means interpolated across the stimulus space.

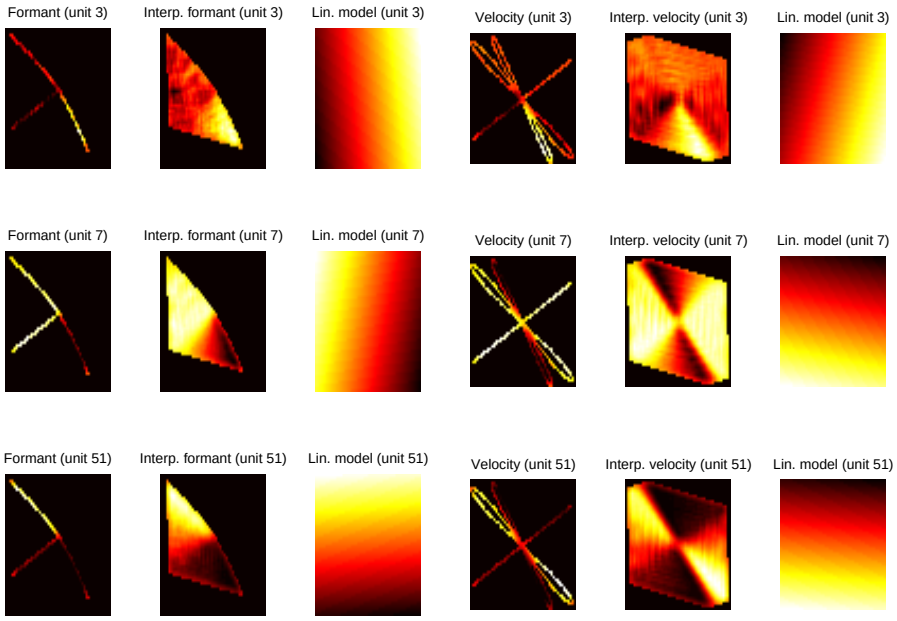


Figure 5·5: Empirical mean activity and linear fits for data with respect to formants (left) and formant velocities (right). Each row illustrates a different single unit. The left three columns indicate empirical mean firing rate (1^{st}), mean firing rate interpolated over the full formant space (2^{nd}) and linear model fit (3^{rd}). The same pattern is used for the right three columns for formant velocities.

.

Session	# units	Training prediction correlation R^2 (F1,F2)	R^2 (V1,V2)
1	51	(0.54, 0.56)	(0.12, 0.10)
2	52	(0.56, 0.62)	(0.18, 0.17)
3	50	(0.51, 0.55)	(0.13, 0.17)
4	55	(0.65, 0.72)	(0.20, 0.19)
5	54	(0.64, 0.64)	(0.22, 0.21)
6	52	(0.58, 0.61)	(0.14, 0.16)
7	55	(0.57, 0.64)	(0.21, 0.15)
8	54	(0.50, 0.64)	(0.15, 0.15)
9	54	(0.55, 0.62)	(0.17, 0.12)
10	53	(0.57, 0.64)	(0.12, 0.15)
11	53	(0.54, 0.62)	(0.15, 0.17)

Table 5.2: Statistical summary of open-loop decoding (closed-loop training paradigm) results.

5.5 Discussion

Two paradigms for offline decoding of neural signals obtained using the Neurotrophic Electrode implant were discussed. The first represented an initial attempt to predict formant frequency trajectories of intended speech productions using the linear filter method for decoding and the rate histogram for rate estimation. These results, and others (Brumberg et al., 2007; Miller et al., 2007; Wright et al., 2007), served as a proof-of-concept for continuous formant frequency trajectory prediction from neural signals in the speech motor cortex. The limitations of this initial analysis were readily apparent. First, the binned rate estimation method is suboptimal when compared against other estimation techniques, specifically the adaptive exponential method, as determined through the analysis in Chapter 4. Second, there was room for improvement in prediction of formant frequencies, as indicated by the poor, but better than chance, classification error using the linear filter method. Last, the experimental paradigm suffered from an "ambiguous response" problem. That is, S1 was instructed to speak immediately following the GO signal, but his reaction time is

not available. Therefore, it was necessary to make the assumption that he did begin immediately following the GO signal. In addition, it was not possible to determine when he had ended the vowel production, adding offset uncertainty. A new offline decoding paradigm was designed to attempt to account for these three issues.

The second, improved, paradigm was designed specifically for use as offline, or open-loop, training for the real-time neural prosthesis. The rate estimation was modified to use the adaptive exponential method with even finer resolution in time (15 ms intervals rather than 50 ms). Neural decoding of formant frequency trajectories was accomplished using the Kalman filter decoder where the world state included both formant frequencies and formant velocities. Last, the protocol was modified such that S1 was instructed to speak along with the acoustic stimulus as it occurred, beginning and ending with the stimulus. This last change allowed for reasonable assumptions that S1 began and ended his intended productions with the acoustic stimulus, alleviating the ambiguous response problem.

This offline decoder successfully fit linear models of firing rate with respect to formants and formant velocities. The linear fit R^2 values, though small, were on par with recent neural decoding efforts which reported good closed-loop performance. Goodness-of-fit, defined as the correlation between training data predictions and the target formant trajectories, was above chance for predicted formant frequencies but not for velocities which was surprising given the considerable amount of evidence for velocity or direction tuning in the motor cortex. There are currently two hypotheses for this result. First, it is possible that the implanted portion of motor cortex is indeed tuned for formant frequencies (or vocal tract positions) and not for velocities. The second hypothesis is that the poor velocity fits are an artifact of the velocity stimulus. The stimulus for this offline protocol utilized steady periods of 1000 ms duration and transitions of 300 ms. Under questioning, S1 indicated that he sometimes leads

(anticipates) or lags the acoustic stimulus in a non-systematic manner. If these leads and lags were large enough, then it would not be possible to accurately align the recorded neural data with the stimulus. The second hypothesis is currently under investigation utilizing a speak-along stimulus in which the steady vowels are presented randomly with 1 second duration steady periods and transitions. The slower stimulus should be more robust to any leads and lags in S1's production and its random nature should ensure that if any misalignments occur, they should all be lags as it will be more difficult to learn the sequence of vowel sounds.

5.6 Conclusion

Both offline neural decoding methodologies employed in this dissertation research indicated that it was possible to predict continuous formant frequency representations of speech sounds from speech motor cortex activations collected with the Neurotrophic Electrode implant. The evolution of decoding techniques described within this chapter was representative of the project in general; namely these methods will constantly be modified for improved decoder performance and to make the problem of control easier for the implant subject. Decoding model fits were shown to be quite poor, though within the ranges reported by recent primate studies conducted in brain computer interface environments (similar to human BCI conditions), as opposed to previous single unit recording environments (c.f. Georgopoulos et al., 1982). The shift in paradigm from repetition to speak-along tasks illustrated the importance of training stimulus choice. Finally, the results discussed in this chapter motivated the implementation of the real-time neural decoder for formant frequency trajectory prediction.

CHAPTER 6

NEURAL DECODING OF SPEECH IN REAL TIME

6.1 Introduction

This chapter represents the culmination of the research discussed in this dissertation. The following sections describe the design and implementation of a neural prosthesis for speech production via computerized speech synthesis and preliminary decoding results from closed-loop experimental trials with subject S1. The main consideration for the final design of the BCI for this dissertation required all methods to be capable of operation in a real-time environment with computational overhead time less than 50 ms.

The main requirement yielded choices of SpikeSort3D (manual cluster cutting) for spike sorting and the Klatt synthesizer for speech synthesis. SpikeSort3D, as mentioned in Chapter 3, has built-in real-time functionality when coupled with the Neuralynx Cheetah data acquisition system. A one way test of system delay was not possible, but a 50-70 ms round trip delay was determined from analysis computer to acquisition system and back. This round trip delay allows inference of a $\sim$35 ms delay due to data acquisition and spike sorting. Speech sound synthesis obtained from the Klatt synthesizer for a 10 ms sample used less than one millisecond of processing time, while at this time, the Maeda system requires more than 100 ms of computational time for the same waveform duration.

A secondary design consideration required estimation of firing rates from stochas-

tic point process spike trains capable of adapting to local and global changes in firing rate. This design requirement eliminated all firing rate estimators except the adaptive exponential ISI-convolution filter method (see Chapter 4 for details), which also requires little computational overhead.[1] Choice of neural decoding algorithm is the last component yet to be discussed. All three methods (PVA, linear filter and Kalman filter) have been shown to work in real-time (PVA: Taylor et al., 2002, Velliste et al., 2008; linear filter: Serruya et al., 2002, Paninski et al., 2004, Hochberg et al., 2006; Kalman filter: Wu et al., 2006, Kim et al., 2007). However, only the Kalman and linear filters have been shown to work in the case of a human subject (Hochberg et al., 2006; Kim et al., 2007) and provide the most accurate predictions (Brockwell et al., 2004; Wu et al., 2006), and only the Kalman filter requires only a single observation (c.f. linear filter) per prediction step. This last feature follows a general principle of simplicity, namely that the Kalman filter requires information from only a single time point to make a new prediction while the linear filter requires information from N time points in the past, where N is the convolution filter length. In addition, the previous offline results discussed in Chapter 5 showed somewhat poor results using the linear filter. Therefore, the Kalman filter was chosen for use in the real-time neural prosthesis.

Using this decoder, S1 participated in a series of closed-loop experimental trials of vowel sequence production. Early sessions with the closed-loop decoder consisted of mainly low-stress tasks designed to acclimate S1 to the decoding device while later sessions actively probed production accuracy. Statistical evaluation of decoded formant trajectories were used to assess production accuracy and decoder performance.

[1] Per timestep firing rate updates are performed in under 1 ms.

6.2 Materials and methods

6.2.1 Subject

A single 24 year old subject, S1, implanted with a three wire Neurotrophic Electrode device in the face area of the left precentral gyrus participated in this study. As previously noted in Chapter 1, S1 sustained a brain stem stroke as a result of an automobile accident rendering him locked-in, without the ability of voluntarily muscle control except for slow movements of his eyes. Neurological evaluation of S1 concluded he has no loss of cognitive function and brain imaging results (MRI and CT) suggest no neuroanatomical deficits other than the brain stem stroke. All procedures were approved by the US Food and Drug Administration, Neural Signals, Inc., Gwinnett Medical Center and Boston University Internal Review Boards.

6.2.2 Experimental protocol

S1 was administered a 200 mg dose of Provigil before every recording session. The desired effect was to ensure he was completely alert during the session and to establish an experimentally controlled baseline neural activity across recording sessions. The closed-loop experimental paradigm consisted of listening and attempted productions of vowel-vowel (V-V) sequences, UH-V where V could be /A/, /OO/, /IY/ or /I/. Taken as a two-dimensional representation of speech, the experimental paradigm closely resembles previous two dimensional movement tasks (Georgopoulos et al., 1982, 1986; Schwartz et al., 1988; Moran and Schwartz, 1999a,b). The target formant frequency trajectories were artificially created by cosine interpolation between the center vowel and endpoints given by the formant frequencies of a familiar vowel space and synthesized using the Klatt formant synthesizer.[2] The stimulus was presented to

[2]Vowel formant frequencies were obtained by spectral analysis of utterances by S1's father.

S1 acoustically as described above and visually as a two-dimensional representation of target vowel sequence formant frequency. After stimulus presentation an acoustic GO cue[3] indicated the beginning of the response period during which S1 was instructed to immediately attempt to produce the vowel sequence while the decoded speech sounds were fed back acoustically through computer speakers and visually on a two-dimensional representation of the decoded formant frequencies blown up and projected onto a wall within S1's visual field. An example of the visual display is shown in Figure 6·1.

During early, low-stress training, S1 was given between six and ten seconds to make a production attempt after the GO signal. In these cases, the stimuli were presented in a block design in which each stimulus type was repeated ten times, split into two blocks between which S1 was given a rest period. In later testing environment sessions, a total movement time of six seconds was allotted for S1 to complete his movement attempt. The stimuli were presented, again, in a block design which consisted of 8-9 trials where stimuli were presented randomly to S1.[4] Target regions were specified for the central /UH/ sound and each endpoint vowel defined as bivariate normal probability distributions with mean equal to the appropriate target formant frequency and diagonal covariance $\Sigma_e = \mathrm{diag}([\sigma_{1e}, \sigma_{2e}])$ with $\sigma_{1e} = 115$ Mel and $\sigma_{2e} = 220$ Mel for the endpoints and and diagonal covariance $\Sigma_c = \mathrm{diag}([\sigma_{1c}, \sigma_{2c}])$ with $\sigma_{1c} = 80$ Mel and $\sigma_{2c} = 150$ Mel for the center vowel. A production trial was considered a success and ended after S1 held the decoded formant frequencies in the endpoint target region for 500 ms and feedback of a correct trial was indicated by three "dings." A trial was considered a failure if he did not achieve the target vowel sound in the allotted movement time or hold the production steady for a complete 500

[3] An computer-generated "ding" was used as the acoustic GO cue.

[4] Distributions of stimulus occurrences were approximately equal.

ms. Early experimental trials did not end at endpoint target acquisition, therefore *post hoc* analysis was required to determine the acquisition (or movement) time.

In addition to specifying the acceptable endpoint region for the current trial, the target regions acted as attractors guiding the decoded formant frequencies. In each trial, the endpoint target regions were activated with gains not so large that every production trial resulted in success. Rather, the gain was chosen low enough in the endpoint regions to minimize the effort required by S1 to hold the production steady but not guarantee a successful trial on every attempt. The target region attraction gain factor was user defined and could be set at any point by the experimenter. An additional parameter, damping or smoothing, was also experimenter-controlled and could vary at any point during a recording session. This parameter regulated a weighting factor for smoothing new predictions. Setting this value to maximum resulted in new predictions that were identical to the previous time point prediction, while the minimum value yielded an estimate not constrained to any previous value (except, of course for the implicit constraints in the Kalman filter model). For these closed-loop sessions, the damping was typically set to one-half maximum and the target region gain was set to 0.35 of maximum.

6.2.3 Data acquisition

Extracellular voltage potentials were obtained using a three-wire (two channels, one reference) Neurotrophic Electrode implanted on the left precentral gyrus approximately in the face area near the premotor and primary motor cortex boundary. Neural signals were preamplified at 100x gain via implanted hardware, transmitted by transcutaneous FM radio to the recording hardware system. Then, the signal was amplified by external digital amplifiers at 100x gain and duplicated. One replicate was lowpass filtered (RAW; 9000 Hz) while the other was bandpass filtered (FILT;

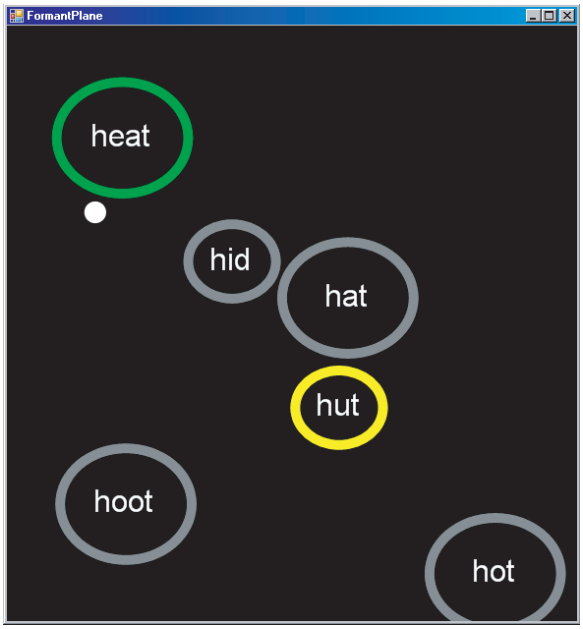


Figure 6·1: Example visual display used in closed-loop production trials. The white cursor represents the current production in terms of the first two formant frequencies. Gray ellipses are inactive target regions and the green ellipse is the active endpoint region. The white cursor moves according to the target trajectory in the listen period and to the decoded formant frequencies in the production period.

300-6000 Hz) then both were recorded onto digital tape. The signal was simultaneously sorted and sent to the decoding software over a local area network TCP connection to the analysis computer using the Neuralynx NetCom interface.

6.2.4 Data analysis

Spikes were sorted online using the Neuralynx, Inc. Cheetah data acquisition system and SpikeSort3D. Firing rates were obtained using the adaptive exponential convolution method and formant frequencies and velocities were predicted using the Kalman filter decoder, though only formant frequencies (i.e. positions) were used for real-time formant speech synthesis and subsequent performance analysis. Acoustic feedback sound waveforms were generated using the Klatt (1980) synthesizer. Audio feedback was achieved through real-time access and playback of the synthesized speech waveform through the analysis computer's sound buffer while visual feedback was presented to the subject as a two-dimensional representation of the decoded formant frequencies. All closed-loop decoding, synthesis and sound playback were implemented in the .Net framework using the Boost uBlas and MAPACK libraries for all matrix operations.

Firing rates, formant trajectories and velocities were written to file for each experimental trial. Early closed-loop experimental sessions used mean distance from production endpoint to target endpoint as a measure of performance while during subsequent sessions a record of hits and misses were used as the main performance indicator. Formant trajectories were normalized against the target trajectory template using the dynamic time warping (DTW) technique (Deller et al., 1993; Rabiner and Juang, 1993). Historically, this normalization method was commonly used for speech recognition before hidden Markov models were popularized. The path of least resistance though the space of penalties (i.e. L_2 distance between target and production) for all combinations of target and production trajectory points is obtained via

a dynamic programming algorithm, yielding the optimal signal warping taking into account prominent features of the two signals. Figure 6·2 illustrates an example path found by the DTW algorithm. Notice that the optimal path is curved while the linear time normalization is the straight-line path along the diagonal.

The mean trajectory for each stimulus type was computed from the single trial normalized productions. The correlation coefficient (squared, R^2) of the entire trajectory versus the target trajectory and movement time were used as measures of performance as was the end point accuracy defined as number of hits versus misses.

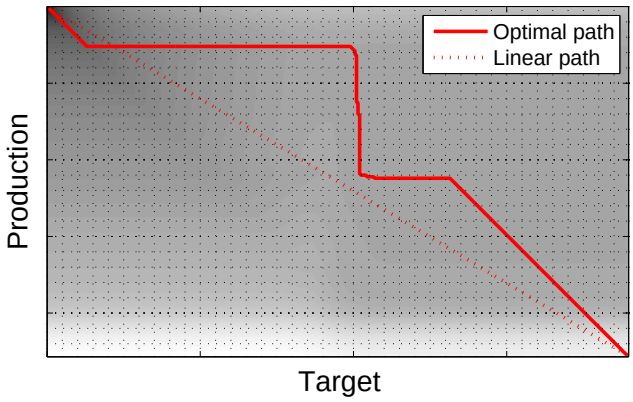


Figure 6·2: Example time warping path found by DTW procedure. Darker background shading indicates regions of relatively low error between the target and production trajectories. The solid red line shows the optimal DTW path while the dotted line represents the linear time warping path.

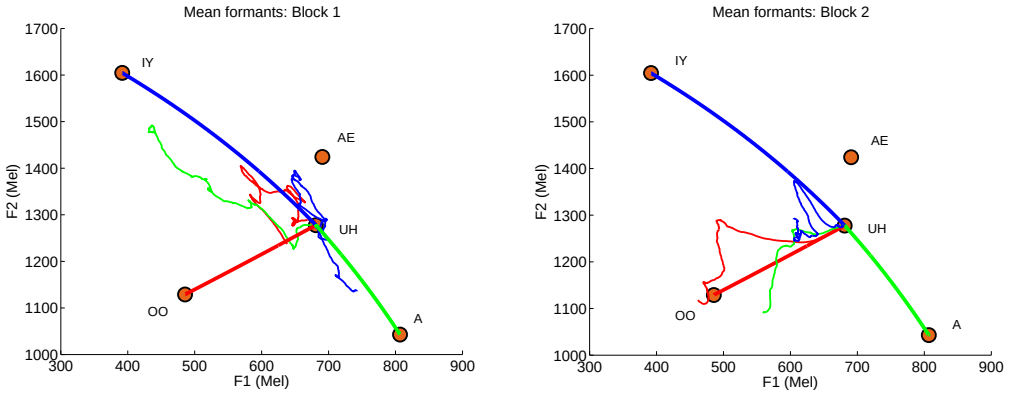
6.3 Results

S1 participated in many closed-loop real-time feedback experimental sessions over a three month period. Some sessions were excluded from analysis due to technical issues

related to the data acquisition hardware and decoding software. In addition, session availability was also determined by the subject's wakefulness. For instance, there was at least one session in which the subject was simply too tired to participate at all. In general, the length of each session was subject to S1's level of fatigue. Last, the subject suffers from severe muscle spasms; trials were eliminated from analysis if a spasm occurred at any point during a trial.

6.3.1 Initial trials

Analysis of decoded formant frequency trajectories for each stimulus indicated a statistically significant decrease in endpoint error between blocks of repeated stimuli. Figure 6·3 shows the target (bold) and mean predicted formant trajectories (normal weighting) plotted on the formant plane. Visually, it is easy to observe that the mean formant trajectories in the first block (left) are very poor. However, the mean formant trajectories are vastly improved during second block productions. In fact, the mean /UH OO/ trajectory is almost perfectly aligned with the target. The mean /UH A/ trajectory looks to have rotated counter-clockwise toward the ideal trajectory while the /UH IY/ remained approximately steady, but with reduced movement in the opposite direction to the target. These qualitative results were confirmed up by quantitative measures of performance and improvement. Specifically, a two-way analysis of variance (ANOVA) for block number and stimulus type was performed. The results indicated a statistically significant decrease in mean endpoint error between block 1 and 2 ($p < 0.005$, F-test) but did not reveal a global effect of stimulus. Individual effects for block number per stimulus type were obtained using two sample t-tests. This analysis indicated significant decreases in mean endpoint error for decoded /UH OO/ ($p < 0.05$) and /UH A/ trials ($p < 0.02$; see Table 6.1).



(A) Block 1 (B) Block 2

Figure 6·3: Target (bold) and predicted mean (normal weighting) trajectories for block 1 and 2 for an example early closed loop session. Trajectories are color coded (red: /UH OO/, green: /UH A/, blue: /UH IY/).

Prediction endpoint error (Mel)

	Block 1	Block 2	% Change
A	**571.34**	**267.81**	**-53**%
OO	**360.12**	**186.56**	**-48**%
IY	585.50	393.67	-33%
All	**505.65**	**282.68**	**-44**%

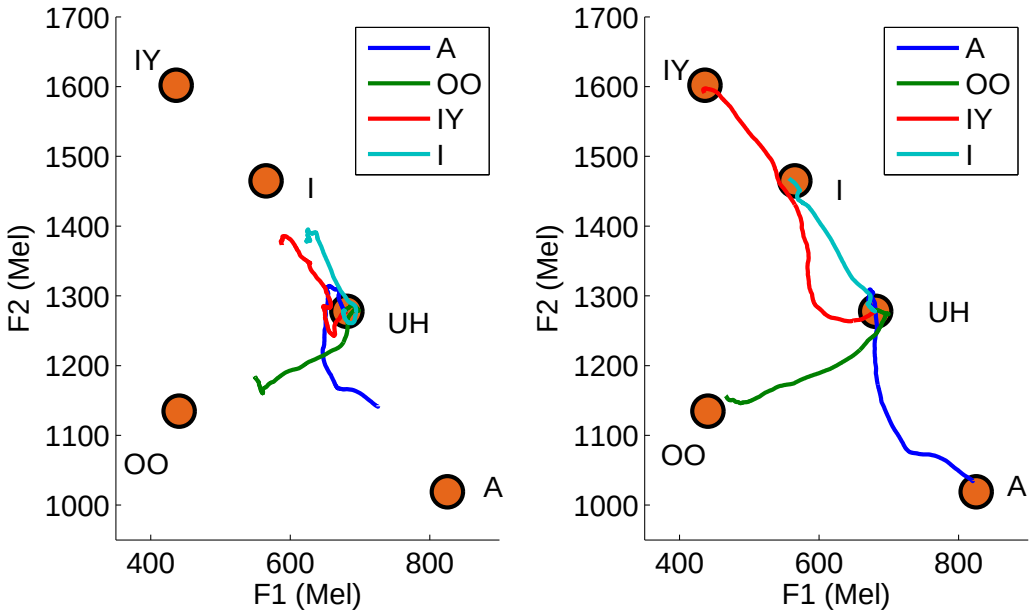
Table 6.1: Root-mean-square error between predicted formant trajectory endpoints and targets for an example early closed-loop session. Bold values indicate statistically significant changes in error between first and second stimulus blocks (all stimuli ANOVA, $p < 0.005$; individual stimuli: /A/ t-test, $p < 0.02$ and /OO/ t-test, $p < 0.05$).

6.3.2 Results across sessions

Overall performance during closed-loop study of the real-time neural prosthesis was quite good. S1 was able to achieve the correct endpoint vowel about 56% of the time averaged across all 19 sessions with a chance rate of approximately 25-30%. The trajectories used were similar to those indicated by the target acoustic stimulus (average r^2 for F1 and F2 was 0.51 and 0.54 respectively). Further analysis revealed the decoded formant trajectories from early experimental sessions were problematic indicating that S1 had difficulty using the neural prosthesis to control synthesized vowel productions. As detailed in Tables 6.3 and 6.4, only eight of 30 attempted productions were deemed "correct" on the first day of closed-loop study and all incurred substantial trajectory error. However, in the space of two months, decoding performance significantly improved.

A linear regression fit was used to determine improvement rate and significance of learning in terms of movement time and end-target acquisition rate as illustrated in Figures 6·5 and 6·6. The analysis showed statistically significant improvement in endpoint acquisition accuracy (F-test; $p < 0.02, F = 6.54$)) at a rate of approximately 1.3% over the closed-loop evaluation period (N=19 sessions) and a 45 ms decrease per session in movement time (F-test; $p = 0.05, F = 3.86$). Mean accuracy and movement time are shown as black squares plotted with 95% confidence intervals and the linear fit is shown in red. The chance level is plotted in gray and was determined by endpoint accuracy and movement time associated with random-walk simulated formant trajectories. The simulated random trajectories were generated according to Equation 5.14 using the state transition and state residual covariance matrices estimated from each session.

Qualitative observation of experimental trials suggested S1 was more capable of



(A) All trials (B) Correct trials

Figure 6·4: Mean closed-loop decoded formant trajectories over all sessions. Top left: average over all trials. Top right: average over correct trials. Traces are color coded, /UH IY/: red, /UH OO/: green, /UH A/: blue and /UH I/: cyan.

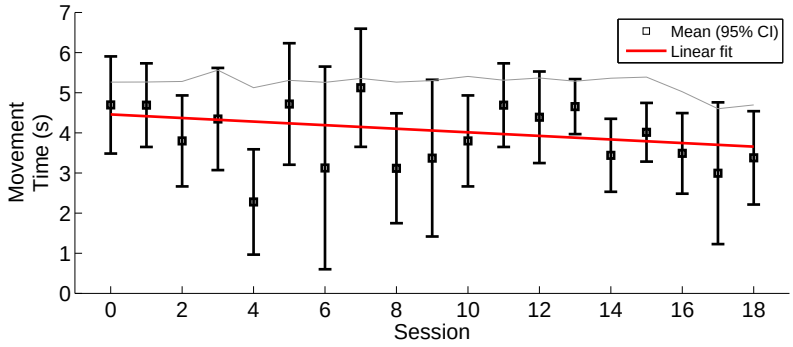


Figure 6·5: Linear fit to end-target acquisition movement time over all trials per closed-loop experimental session (slope=-0.045, $p = 0.05$). Individual points (black) indicate mean movement time per trial plotted with 95% confidence intervals. Linear fit is shown in red and movement times due to change shown in gray.

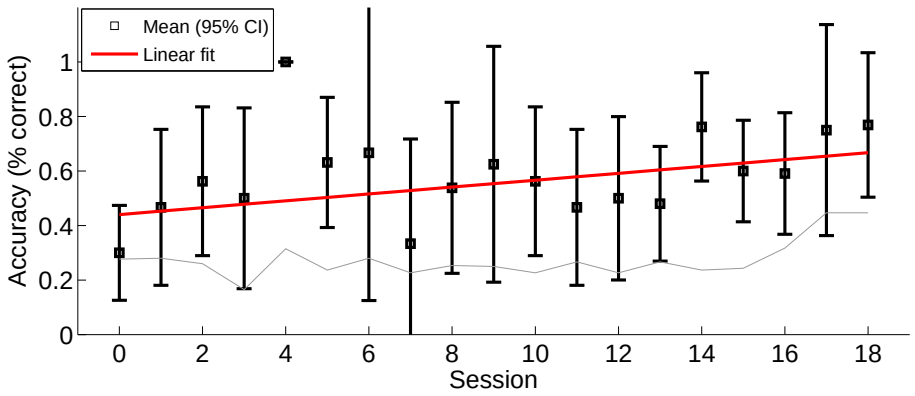


Figure 6·6: Linear fit to end-target acquisition accuracy (% correct) over all trials per closed-loop experimental session (slope=0.013, $p < 0.05$). Individual points (black) indicate mean accuracy per trial plotted with 95% confidence intervals. Linear fit is shown in red and accuracy rates due to chance shown in gray.

producing certain vowel sounds with the decoder than others. In particular, there seemed to be difficulty in production of the /IY/ sound relative to /A/, /OO/ and /I/. An unbalanced four-way analysis of variance was used to test this observation as well as to test for effects of other factors in the experimental design. The four factors were session number, stimulus type (/A/, /OO/, /IY/ and /I/), trial design (repeated vs. random blocks) and presence or absence of visual feedback. The analysis confirmed the previous result of significant main effect of learning across sessions ($F = 4.8, p < 0.03$). In addition, a main effect of stimulus type was found ($p < 0.002, F = 5.49$) though there were no significant main effects of trial design and visual feedback. A full ANOVA summary can be found in Table 6.2. Subsequent *post hoc* multiple comparisons (t-test) indicated specific significant differences between /IY/ endpoint accuracy vs. /A/, /OO/ and /I/. Specifically, /A/ (61%), /OO/ (65%) and /I/ (78%) targets were acquired with higher accuracy than /IY/ (36%), shown graphically in Figure 6·7.

Source	Sum Sq.	d.f.	Mean Sq.	F	Prob>F
Session	1.1027	1	1.1027	4.7980	**0.0293**
Stimulus	3.7827	3	1.2609	5.4863	**0.0011**
Trial design	0.3908	1	0.3908	1.7005	0.1933
Vis. Feedback	0.1411	1	0.1411	0.6140	0.4339
Error	66.8801	291	0.2298		
Total	73.5302	297			

Table 6.2: ANOVA summary for tests of experimental design factor significance with respect to end target acquisition accuracy in closed-loop task. Bold values represent significant main effects of Session and Stimulus but not for Trial design nor Visual feedback. Session number was taken as a continuous variable while the remaining factors were categorical.

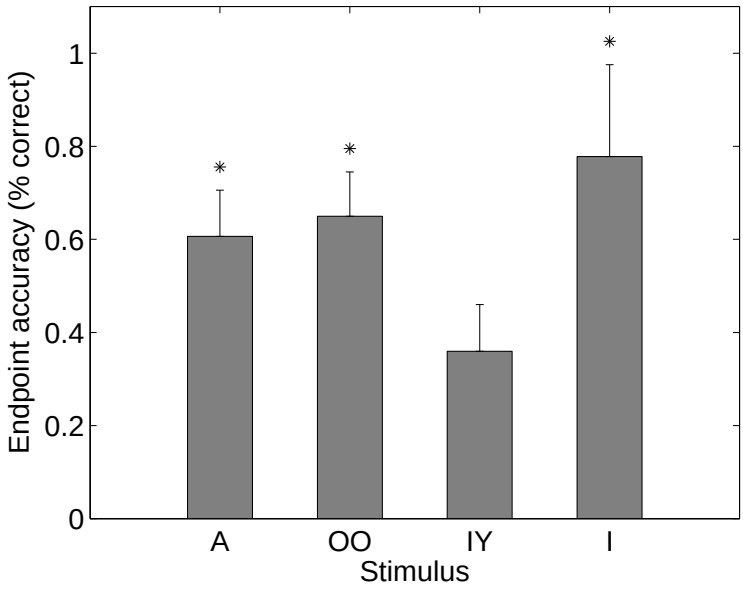


Figure 6·7: Endpoint accuracy (% correct) over all closed-loop trials per stimulus. Error bars indicate 95% confidence interval. *: significant ($p < 0.05$) *post-hoc* multiple t-test comparisons.

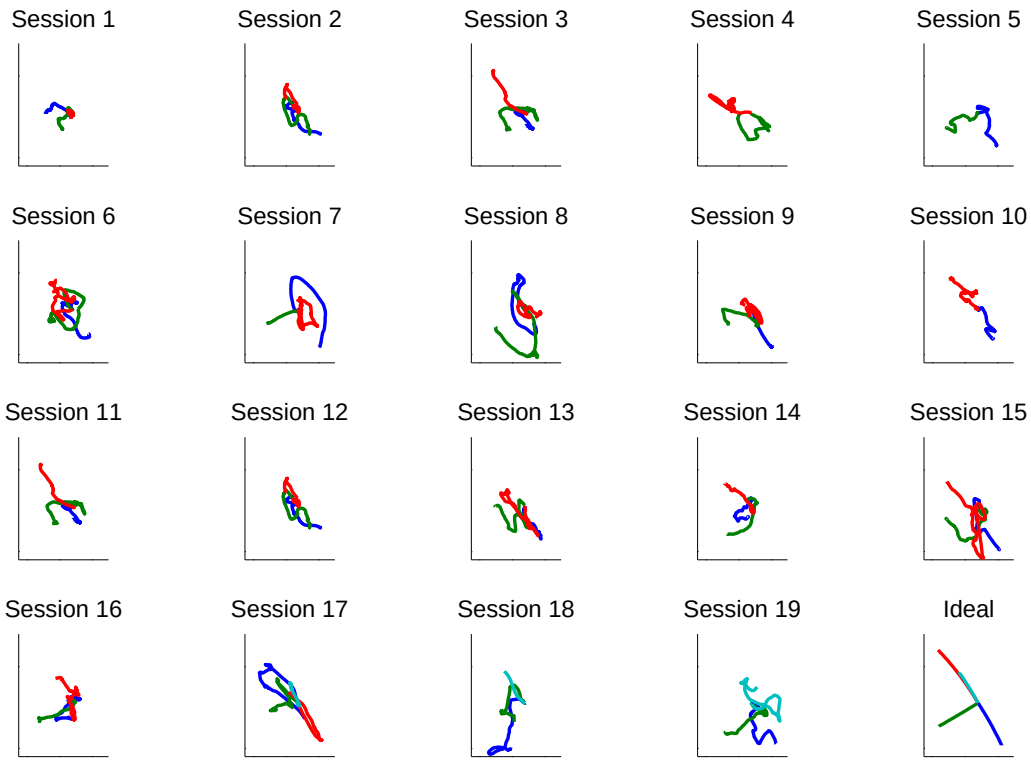


Figure 6·8: Mean trajectories of all production trials for formant frequencies decoded using the Kalman filter BCI for speech synthesis per session. The ideal trajectories are shown in the bottom right. Trajectories are color coded, /UH IY/: red, /UH OO/: green, /UH A/: blue and /UH I/: cyan.

	Stim	N	r^2 (F_1,F_2)	MT(s)
Sess. 1	A	2/10	(0.61, 0.48)	4.77
	OO	5/10	(0.76, 0.62)	3.43
	IY	2/10	(0.49, 0.28)	5.88
	All	9/30	(0.62, 0.46)	4.69
Sess. 2	A	4/6	(0.54, 0.69)	4.17
	OO	0/3	(0.41, 0.40)	6.00
	IY	3/6	(0.27, 0.38)	4.55
	All	7/15	(0.41, 0.51)	4.69
Sess. 3	A	2/5	(0.39, 0.43)	4.32
	OO	3/6	(0.55, 0.33)	3.94
	IY	4/5	(0.57, 0.66)	3.11
	All	9/16	(0.51, 0.47)	3.80
Sess. 4	OO	3/5	(0.41, 0.44)	3.66
	IY	3/7	(0.23, 0.45)	4.83
	All	6/12	(0.30, 0.45)	4.34
Sess. 5	A	2/2	(0.23, 0.75)	3.08
	OO	4/4	(0.62, 0.54)	1.88
	All	6/6	(0.49, 0.61)	2.28
Sess. 6	A	5/6	(0.18, 0.64)	3.47
	OO	4/5	(0.56, 0.64)	3.37
	IY	3/8	(0.25, 0.35)	6.50
	All	12/19	(0.31, 0.52)	4.72
Sess. 7	A	2/2	(0.70, 0.87)	2.39
	OO	2/2	(0.99, 0.99)	0.88
	IY	0/2	(0.48, 0.03)	6.10
	All	4/6	(0.72, 0.63)	3.13
Sess. 8	A	1/3	(0.26, 0.31)	4.44
	OO	2/2	(0.18, 0.62)	4.21
	IY	0/4	(0.38, 0.41)	6.10
	All	3/9	(0.30, 0.42)	5.12
Sess. 9	A	3/3	(0.91, 0.97)	0.86
	OO	3/3	(0.90, 0.65)	1.56
	IY	1/7	(0.21, 0.33)	4.75
	All	7/13	(0.53, 0.55)	3.12
Sess. 10	A	3/5	(0.40, 0.58)	3.30
	IY	2/3	(0.39, 0.50)	3.49
	All	5/8	(0.40, 0.55)	3.37
Sess. 11	A	2/5	(0.39, 0.43)	4.32
	OO	3/6	(0.55, 0.33)	3.94
	IY	4/5	(0.57, 0.66)	3.11
	All	9/16	(0.51, 0.47)	3.80
Sess. 12	A	4/6	(0.54, 0.69)	4.17
	OO	0/3	(0.41, 0.40)	6.00
	IY	3/6	(0.27, 0.38)	4.55
	All	7/15	(0.41, 0.51)	4.69
Sess. 13	A	4/5	(0.69, 0.65)	3.24
	OO	3/6	(0.64, 0.52)	4.54
	IY	0/3	(0.56, 0.38)	6.00
	All	7/14	(0.64, 0.53)	4.39
Sess. 14	A	4/9	(0.54, 0.57)	4.91
	OO	5/7	(0.65, 0.69)	4.12
	IY	3/9	(0.53, 0.47)	4.81
	All	12/25	(0.57, 0.57)	4.65
Sess. 15	A	9/10	(0.48, 0.72)	2.56
	OO	6/8	(0.53, 0.53)	3.78
	IY	1/3	(0.04, 0.47)	5.44
	All	16/21	(0.44, 0.61)	3.44
Sess. 16	A	3/7	(0.43, 0.47)	5.14
	OO	13/15	(0.83, 0.63)	2.83
	IY	2/8	(0.28, 0.35)	5.24
	All	18/30	(0.59, 0.52)	4.01
Sess. 17	A	1/3	(0.12, 0.26)	4.96
	OO	1/3	(0.37, 0.53)	5.52
	IY	1/3	(0.35, 0.45)	4.93
	I	10/13	(0.82, 0.85)	2.35
	All	13/22	(0.60, 0.67)	3.49
Sess. 18	A	1/2	(0.31, 0.68)	5.13
	OO	2/3	(0.61, 0.74)	3.38
	I	3/3	(0.84, 0.93)	1.18
	All	6/8	(0.62, 0.79)	2.99
Sess. 19	A	5/5	(0.55, 0.75)	2.81
	OO	4/6	(0.74, 0.65)	3.32
	I	1/2	(0.24, 0.40)	4.97
	All	10/13	(0.59, 0.65)	3.38

Table 6.3: Closed-loop performance for all predictions.

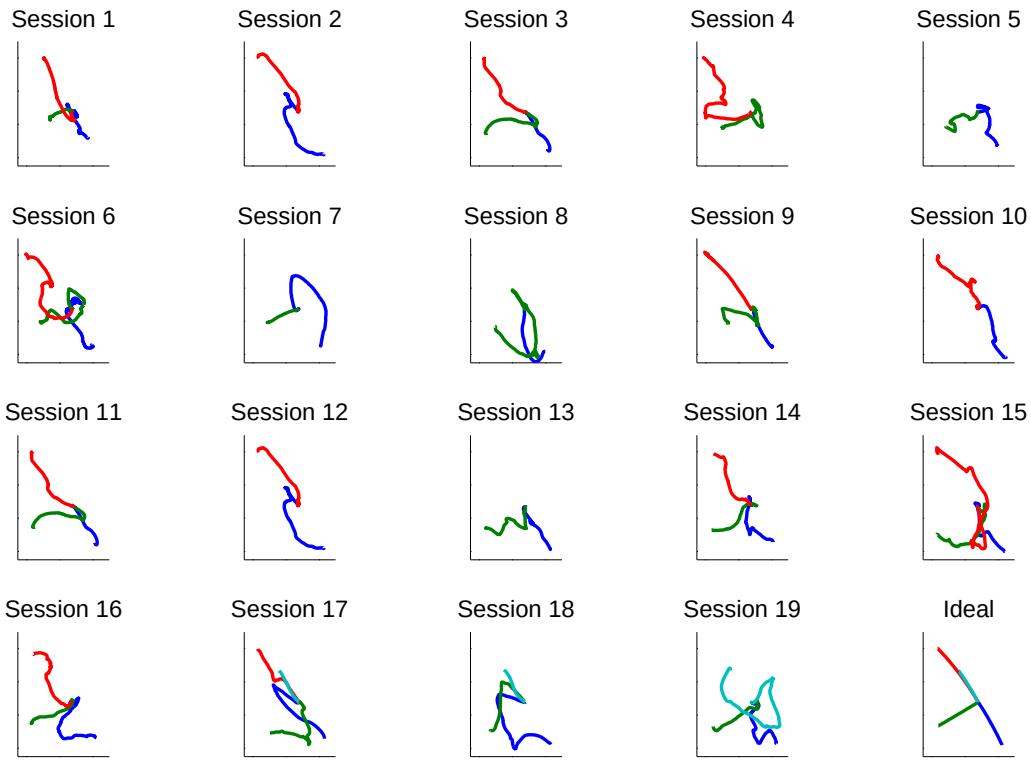


Figure 6·9: Mean trajectories of correct production trials for formant frequencies decoded using the Kalman filter BCI for speech synthesis per session. The ideal trajectories are shown in the bottom right. Trajectories are color coded, /UH IY/: red, /UH OO/: green, /UH A/: blue and /UH I/: cyan.

	Stim	N	r^2 (F_1,F_2)	MT(s)
Sess. 1	A	2/10	(0.34, 0.44)	4.13
	OO	5/10	(0.91, 0.61)	1.90
	IY	2/10	(0.84, 0.64)	3.41
	All	9/30	(0.77, 0.58)	2.73
Sess. 2	A	4/6	(0.51, 0.75)	3.25
	OO	0/3	(NaN, NaN)	NaN
	IY	3/6	(0.36, 0.56)	3.10
	All	7/15	(0.45, 0.67)	3.19
Sess. 3	A	2/5	(0.86, 0.83)	1.80
	OO	3/6	(0.83, 0.56)	1.88
	IY	4/5	(0.61, 0.81)	2.38
	All	9/16	(0.74, 0.73)	2.09
Sess. 4	OO	3/5	(0.66, 0.64)	2.04
	IY	3/7	(0.09, 0.68)	3.14
	All	6/12	(0.38, 0.66)	2.59
Sess. 5	A	2/2	(0.23, 0.75)	3.08
	OO	4/4	(0.62, 0.54)	1.88
	All	6/6	(0.49, 0.61)	2.28
Sess. 6	A	5/6	(0.21, 0.61)	2.93
	OO	4/5	(0.67, 0.74)	2.68
	IY	3/8	(0.34, 0.63)	3.03
	All	12/19	(0.40, 0.66)	2.87
Sess. 7	A	2/2	(0.70, 0.87)	2.39
	OO	2/2	(0.99, 0.99)	0.88
	IY	0/2	(NaN, NaN)	NaN
	All	4/6	(0.85, 0.93)	1.64
Sess. 8	A	1/3	(0.10, 0.66)	1.12
	OO	2/2	(0.18, 0.62)	4.21
	IY	0/4	(NaN, NaN)	NaN
	All	3/9	(0.16, 0.63)	3.18
Sess. 9	A	3/3	(0.91, 0.97)	0.86
	OO	3/3	(0.90, 0.65)	1.56
	IY	1/7	(0.63, 0.70)	0.88
	All	7/13	(0.87, 0.79)	1.16
Sess. 10	A	3/5	(0.60, 0.88)	1.50
	IY	2/3	(0.47, 0.56)	2.24
	All	5/8	(0.54, 0.75)	1.79
Sess. 11	A	2/5	(0.86, 0.83)	1.80
	OO	3/6	(0.83, 0.56)	1.88
	IY	4/5	(0.61, 0.81)	2.38
	All	9/16	(0.74, 0.73)	2.09
Sess. 12	A	4/6	(0.51, 0.75)	3.25
	OO	0/3	(NaN, NaN)	NaN
	IY	3/6	(0.36, 0.56)	3.10
	All	7/15	(0.45, 0.67)	3.19
Sess. 13	A	4/5	(0.70, 0.74)	2.55
	OO	3/6	(0.80, 0.54)	3.08
	IY	0/3	(NaN, NaN)	NaN
	All	7/14	(0.74, 0.66)	2.78
Sess. 14	A	4/9	(0.55, 0.77)	3.55
	OO	5/7	(0.71, 0.72)	3.38
	IY	3/9	(0.51, 0.80)	2.45
	All	12/25	(0.61, 0.75)	3.20
Sess. 15	A	9/10	(0.49, 0.79)	2.19
	OO	6/8	(0.59, 0.62)	3.04
	IY	1/3	(0.02, 0.71)	4.39
	All	16/21	(0.50, 0.72)	2.65
Sess. 16	A	3/7	(0.27, 0.71)	4.01
	OO	13/15	(0.88, 0.72)	2.35
	IY	2/8	(0.35, 0.73)	3.02
	All	18/30	(0.72, 0.72)	2.70
Sess. 17	A	1/3	(0.09, 0.18)	2.88
	OO	1/3	(0.69, 0.58)	4.59
	IY	1/3	(0.96, 0.96)	2.83
	I	10/13	(0.89, 0.96)	1.26
	All	13/22	(0.82, 0.87)	1.76
Sess. 18	A	1/2	(0.15, 0.61)	4.31
	OO	2/3	(0.79, 0.87)	2.09
	I	3/3	(0.84, 0.93)	1.18
	All	6/8	(0.71, 0.86)	2.01
Sess. 19	A	5/5	(0.55, 0.75)	2.81
	OO	4/6	(0.90, 0.80)	2.00
	I	1/2	(0.47, 0.14)	3.97
	All	10/13	(0.68, 0.71)	2.60

Table 6.4: Closed-loop performance for correct predictions only.

6.4 Discussion

A neural prosthesis for speech production was implemented and tested by a human subject in a closed-loop real-time environment. Early use of the device indicated the subject and decoder had little proficiency in obtaining correct predictions of intended speech productions. However, in a relatively short time, predictions of the subject's intended behavior significantly improved. In fact, the subject was capable of significantly improved performance within just one recording session. In general, endpoint accuracy measurements were greater than what could be expected due to chance though no session resulted in perfect (100%) formant prediction and the per session improvement rate was relatively modest (1.3%). Additionally, target endpoint acquisition success rates significantly differed between stimulus types, namely the subject consistently performed worse on /UH IY/ productions than any other. Such poor performance for /IY/ target acquisition allows speculation that S1, through the neural prosthesis, does not have access to the vocal tract representations responsible for /IY/ sound production. According to articulatory and acoustic analysis of speech, the /IY/ sound is characterized mainly by an extreme movement of the tongue dorsum to the top of the palate which corresponds directly to control of the second formant frequency. Therefore, it is possible that the electrode is implanted in a mostly jaw representation area of the precentral gyrus, a possibility supported by close anatomical inspection of the electrode location.

Correlation between decoded and target formant trajectories were similar to those obtained in previous studies of arm and hand kinematic trajectory prediction. In fact, using results obtained by Carmena et al. (2003) it is possible to hypothesize the likely *specific* location of the neural implant based on correlation values alone. Carmena et al. (2003) performed a neuron-dropping analysis of observed vs. pre-

dicted hand movement trajectory correlation for neurons recorded from the primary motor and dorsal premotor cortex (see Figure 6·10 for their results). The R^2 values found in the current study of formant frequency (i.e. position) prediction (R^2(F1, F2) = (0.51,0.54)) agree well with the dPMC values possibly indicating that the Neurotrophic Electrode is implanted in the premotor portion of the precentral gyrus (specifically the ventral premotor cortex). Formant velocities were not analyzed for the closed-loop task, therefore no information from the current study is shown in the velocity plot in Figure 6·10. However, velocity R^2 values from the open-loop task were $\sim$ 0.16, well below what is expected from Carmena et al.'s (2003) neuron-dropping analysis for 51 units in either primary or premotor cortex. A secondary interpretation of the Carmena et al. results is that multiple representations of the musculature exist across the primary and secondary motor cortex. Furthermore, it is advantageous for a motor cortical neural prosthesis to utilize information across all motor areas for precise and accurate prediction of intended behavior.

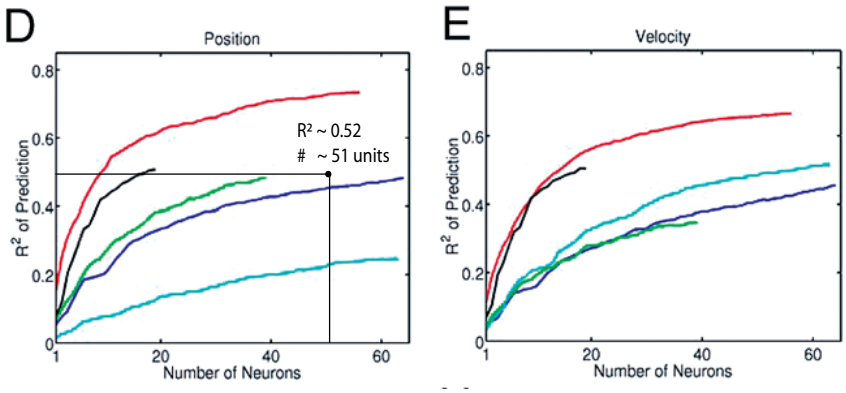


Figure 6·10: Neuron dropping analysis, Fig 2(D-E) from Carmena et al. (2003) with average formant frequency R^2 from the closed-loop task.. Dark blue trace indicates the average prediction R^2 for the dPMC while MI is represented in red. Average F1 and F2 R^2 ($\sim$ 0.52) of the closed-loop predictions shown as the single circle data point. No neuron-dropping analysis was performed in this research. Lines are provided for easier location of the average R^2 value. This value agrees well with the dPMC trace.

6.5 Conclusion

The results obtained during real-time closed-loop experiments of vowel-vowel utterance repetition prove that it is possible to decode formant frequencies from single units in the precentral gyrus. There was significant improvement in target acquisition over time, both within a session and across sessions. The analysis indicated that certain vowel sounds were obtained with much lower success rates, suggesting a relative inability for the subject to produce those sounds. Utilization of articulatory and acoustic analysis of the deficient vowel sound indicated that the electrode may be implanted in an area of motor cortex not functionally related to the vocal tract configurations necessary to produce the sound. Further, comparison of decoding performance measures against previous motor cortical decoding studies showed the results of the current investigation are with the range of expected values associated with "good" performance. Last, a specific comparison to results by Carmena et al. (2003) suggested the electrode may be implanted toward the ventral premotor cortex portion of the precentral gyrus.

The real-time, closed-loop performance was limited by a number of factors. First, relatively few experimental trials were acquired per session (no more than 30). This is quite low compared to other human and primate studies where typically hundreds of trials are performed. In addition, the closed-loop paradigm was confounded by the ambiguous response problem described in Chapter 5. That is, S1 was instructed to begin speaking after the hearing GO cue, but it was not possible to detect when he actually began attempting the utterance. Previously, this problem was alleviated through utilization of a speak-along task in which it was reasonable to assume S1 began speaking internally with the acoustic stimulus. Unfortunately, it is not feasible to alleviate the ambiguous response problem in the closed-loop paradigm with a

speak-along task.[5] Rather, ongoing studies of voicing onset and termination are being performed to detect S1's production epoch. The goal of such analysis is to include a voicing decoder and controller into the neural prosthesis design. Another limitation was that the decoder required offline recalibration before each session. This is unacceptable for two reasons: (1) learning from previous sessions is partially obliterated and (2) it is not practical for a permanent neural prosthesis. An adaptive Kalman filter decoder is currently under development which will allow for online, incremental calibration of the prosthesis. This decoder, similar in spirit to the decoder in Taylor et al. (2002), will eliminate loss-of-learning as the adaptation will only allow small changes to the decoder. Last, the decoder did not provide any control over naturalness parameters, such as fundamental frequency. As such, all synthesized speech utilizing the real-time neural decoder sounded flat and artificial. Further development of the prosthesis device will incorporate fundamental frequency modulation, among other parameters, to increase naturalness.

Nonetheless, the totality of evidence presented in this chapter proves the concept of decoding speech as a continuously varying motor control task where articulatory gestures are replaced by formant frequency trajectories. The subject's success, despite the limitations summarized above, in the control of a real-time speech synthesizer is remarkable and provides the basis for future work on speech restoration by neural prosthesis.

[5]Having two audio streams: (1) stimulus and (2) decoded feedback would likely confuse the subject and it could make interpretation of results more difficult, as it would not be possible to dissociate the effect of the stimulus from the decoded feedback on neural activity.

CHAPTER 7

CONCLUSION

This dissertation presented novel research that led to development and implementation of the first brain computer interface for control of an artificial speech synthesizer by a locked-in patient in real-time. The underlying tenet that served as a basis for all research performed for this dissertation was the perspective that speech production, at the level of the motor cortex, is a continuously varying motor task, rather than one of discrete classification and concatenation of lexical units. In this way, the studies presented within were able to take advantage of over two decades of hand and arm motor cortical electrophysiological investigations without any loss of generalization. The final neural prosthesis design was dependent on efficient integration of research from many disparate fields of study including articulatory and acoustic phonetics, discrete-time signal processing, electrophysiology, and software engineering.

Chapter 1 introduced the problem of speech restoration for a paralyzed individual. Brief reviews of articulatory and acoustic analyses of speech illustrated the motoric nature of speech production and the relationships between vocal tract configuration and prominent acoustic features of the speech waveform, namely the formant frequencies. Understanding these relationships made it possible to reduce the dimensionality of decoded speech output from a multidimensional articulatory domain to a much more tractable two-dimensional formant frequency representation. While this reduction is advantageous in terms of decoding and ease of user control and relates the problem of speech production to previous studies of 2D cursor movement, it suffers a signif-

icant limitation. Utilization of formant frequencies for speech synthesis, described in Chapter 2 restricts "intelligible" production to vowel sounds and some vowel-like consonants known as liquids and glides (e.g. /Y/, /R/, /L/ and /W/). That is, fluent speech requires production of vowels, liquids and consonants (stop, fricative and affricate). Production of these latter consonants is dependent on complete closure of the vocal tract yielding zero airflow and consequently zero energy, passing through the upper vocal tract making it impossible to estimate formant frequencies. Due to the success using the first two formant frequencies alone, future decoding efforts will be performed in higher dimensional domains (i.e. first three formants) and eventually switch to articulatory modality for complete, direct control of a simulated vocal tract enabling production of all sound types. Work to improve the computational overhead of an articulatory synthesizer (Maeda model, Chapter 2) for speech production is ongoing and will be integrated into future prosthesis designs.

In Chapter 3 the important task of identification and classification of single distinct spike waveforms was discussed. Manual selection of voltage thresholds (for identification) and definition of spike cluster boundaries (for classification) were used for the current prosthesis design. Manual clustering of spike waveforms has been shown to be prone to human error, though it remains very popular in the field of experimental neuroscience and is the only method available capable of performing in real-time with the Neuralynx Cheetah data acquisition system. Obvious avenues for future work include systematic analysis of various spike detection and clustering techniques.

A novel method for single unit spiking rate estimation was presented in Chapter 4, called an *adaptive exponential convolution filter*. It is essentially a modified kernel density estimation procedure utilizing a convolution window appropriate for analysis of neural spike trains. In this case, the window used was an exponential fit to the interspike interval distribution describing the dependence of each rate estimation

point on previous spiking history. Further, the rate parameter of the exponential was allowed to vary in time with underlying changes in the firing behavior of the single units. This estimator surpassed all other considered techniques in a battery of performance analyses. This method, though kernel-based, was found to be related to density estimation by penalized linear splines. Though the cubic splines investigated in this dissertation research did not perform overly well, it is possible that improvements to the rate estimation procedure may be gained by further analysis of the class of penalized linear regression splines.

Proof-of-concept for neural decoding of speech utilizing formant frequencies was established in Chapter 5. Initial studies using binned rate estimation and an optimal linear filter for decoding indicated that formant prediction from neural activity in the ventral precentral gyrus was possible, though performance was sub-par. Improvements to both the experimental design and decoding methodology led to the final neural prosthesis utilizing the adaptive exponential rate estimator joined with a Kalman filter for formant frequency (and formant velocity) prediction. Training performance and model goodness-of-fit was within the ranges reported by other studies involving decoding of arm and hand kinematics in monkeys. Last, an *offline* open-loop training paradigm was established for calibration of the real-time neural prosthesis for closed-loop usage.

The real-time prosthesis for speech production was tested using a closed-loop experimental protocol. This task, spanning over three months, required the subject to listen to and repeat vowel-vowel acoustic stimuli artificially generated by a computer. The vowel stimuli were chosen to mimic the well known 2D center-out studies initially conducted by Georgopoulos and colleagues. The vowel sound /UH/ was taken as the center point, and the four radial endpoints used were /A/, /OO/, /IY/ and /I/. Both /A/ and /OO/ are low F2 vowel sounds related primarily to movements of

the jaw (though lip pursing is required for /OO/ and both include some tongue movement) while /IY/ and /I/ are relatively high F2 sounds characterized tongue movements up and down in the oral cavity. The closed-loop study revealed decoder performance greater than expectations due to chance and a modest improvement over the course of the study. Additionally, analysis of average accuracy per stimulus type showed the subject/decoder had great difficulty in successful productions of formant trajectories ending with the /IY/ sound, suggesting the electrode is not implanted in an area representing up and down movements of the tongue. Similarly, the relative success of /A/ and /OO/ sound production implies an implant location in an area representing jaw movements and possibly tongue movements backward and forward. Last, analysis of predicted vs. target formant trajectories compared against recent studies by Carmena et al. (2003) suggested that the implant may be implanted primarily in the premotor cortex portion of the ventral precentral gyrus, although this is fairly speculative and requires further investigation.

The closed-loop study was limited in that the decoder was recalibrated, offline, before most recording sessions. This meant that the subject would have to relearn a slightly different decoder every session, eliminating some learning effects gained from previous sessions. Design of an adaptive decoder (specifically adaptive Kalman filter decoder) is ongoing and should account for this significant limitation. Taylor et al. (2002) similarly implemented and tested an adaptive population vector algorithm decoder for control of a robotic arm by a monkey subject and achieved much better performance compared to its static counterpart. Therefore, utilization of an adaptive decoder for speech production is expected to outperform the static decoder described in this dissertation.

REFERENCES

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

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Bankman, I. N., Johnson, K. O., and Schneider, W. (1993). Optimal detection, classification, and superposition resolution in neural waveform recordings. *IEEE Transactions on Biomedical Engineering*, 40(8):836–841.

Barbieri, R., Quirk, M. E., Frank, L. M., Wilson, M. A., and Brown, E. N. (2001). Construction and analysis of non-poisson stimulus-response models of neural spiking activity. *Journal of Neuroscience Methods*, 105:25–37.

Bartels, J. L., Andreasen, D. S., Kennedy, P. R., Siebert, S. A., and Wright, E. J. (2007). Human speech cortex long-term recordings [2]: inputs, outputs and signal stability. In *Neuroscience Meeting Planner 2007*, Program No. 517.19, San Diego, CA. Society for Neuroscience.

Bishop, P. O., Levick, W. R., and Williams, W. O. (1964). Statistical analysis of the dark discharge of lateral geniculate neurones. *Journal of Physiology*, 170(3):598–612.

Blatt, M., Wiseman, S., and Domany, E. (1996). Superparamagnetic clustering of data. *Physical Review Letters*, 76(18):3251–3254.

Bohland, J. W. and Guenther, F. H. (2006). An fmri investigation of syllable sequence production. *Neuroimage*, 32(2):821–841.

Breiman, L., Meisel, W., and Purcell, E. (1977). Variable kernel estimates of multivariate densities. *Technometrics*, 19(2):135–144.

Brockwell, A. E., Rojas, A. L., and Kass, R. E. (2004). Recursive bayesian decoding of motor cortical signals by particle filtering. *Journal of Neurophysiology*, 91:1899–1907.

Brown, E. N., Barbieri, R., Eden, U. T., and Frank, L. M. (2004). Likelihood methods for neural spike train data analysis. In Feng, J., editor, *Computational Neuroscience: A Comprehensive Approach*, chapter 9, pages 253–286. CRC Press.

Brown, E. N., Barbieri, R., Ventura, V., Kass, R. E., and Frank, L. M. (2001). The time-rescaling theorem and its application to neural spike train data analysis. *Neural Computation*, 14:325–346.

Brown, E. N., Frank, L. M., Tang, D., Quirk, M. C., and Wilson, M. A. (1998). A statistical paradigm for neural spike train decoding applied to position prediction from ensemble firing patterns of rat hippocampal place cells. *Journal of Neuroscience*, 18(18):7411–7425.

Brumberg, J. S., Andreasen, D. S. Bartels, J. L., Guenther, F. H., Kennedy, P. R., Siebert, S. A., Schwartz, A. B., Velliste, M., and Wright, E. J. (2007). Human speech cortex long-term recordings [5]: formant frequency analyses. In *Neuroscience Meeting Planner 2007*, Program No. 517.17, San Diego, CA. Society for Neuroscience.

Cai, T., Hyndman, R., and Wand, M. P. (2002). Mixed model-based hazard estimation. *Journal of Computational and Graphical Statistics*, 11(4):784–798.

Carmena, J. M., Lebedev, M. A., Crist, R. E., O'Doherty, J. E., Santucci, D. M., Dimitrov, D. F., Patil, P. G., Henriquez, C. S., and Nicolelis, M. A. L. (2003). Learning to control a brain-machine interface for reaching and grasping by primates. *PloS Biology*, 1(2):193–208.

Cheeseman, P. and Stutz, J. (1996). Bayesian classification (AutoClass): Theory and results. In Fayyad, U. M., Piatestsky-Shapiro, G., Smyth, P., and Uthurusamy, R., editors, *Advances in Knowledge Discovery and Data Mining*, chapter 6, pages 61–83. AAAI Press/MIT Press.

Comaniciu, D. (2003). An algorithm for data-driven bandwidth selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):281–288.

Comaniciu, D. and Meer, P. (2002). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):1–18.

Csicsvari, J., Hirase, H., Czurko, A., and Buzsáki, G. (1998). Reliability and state dependence of pyramidal cell-interneuron synapses in the hippocampus: an ensemble approach in the behaving rat. *Neuron*, 21:179–189.

Daley, D. and Vere-Jones, D. (2003). *An introduction to the theory of point process.* SpringerVerlag, New York, 2nd edition.

Deller, J. R., Proakis, J. G., and Hansen, J. H. (1993). *Discrete Time Processing of Speech Signals.* Prentice Hall PTR, Upper Saddle River, NJ, USA.

Doupe, Allison, J. and Kuhl, P. K. (1999). Birdsong and human speech: common themes and mechanisms. *Annual Review of Neuroscience*, 22:567–631.

Dudley, H., Reisz, R. R., and Watkins (1939). A synthetic speaker. *Journal of the Franklin Institute*, 227:739–764.

Dudley, H. and Tarnoczy, T. H. (1950). The Speaking Machine of Wolfgang von Kempelen. *The Journal of the Acoustical Society of America*, 22:151.

Fant, G. (1960). *Acoustic theory of speech production*. Mouton & Co. N.V., The Hague.

Fee, M. S., Mitra, P. P., and Kleinfeld, D. (1996). Automatic sorting of multiple unit neuronal signals in the presence of anisotropic and non-gaussian variability. *Journal of Neuroscience Methods*, 69:175–188.

Ferrari, P. F., Gallese, V., Rizzolatti, G., and Fogassi, L. (2003). Mirror neurons responding to the observation of ingestive and communicative mouth actions in the monkey ventral premotor cortex. *European Journal of Neuroscience*, 17:1703–1714.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188.

Flanagan, J. L. (1972). Voices of Men and Machines. *The Journal of the Acoustical Society of America*, 51:1375.

Fraley, C. and Raftery, A. E. (2006). Mclust version 3 for r: Normal mixture modeling and model-based clustering. Technical report.

Gao, Y., Black, M., Bienenstock, E., Shoham, S., and Donoghue, J. (2002). Probabilistic inference of hand motion from neural activity in motor cortex. *Advances in Neural Information Processing Systems*, 14:213–20.

Gao, Y., Black, M. J., Bienenstock, E., Wu, W., and Donoghue, J. P. (2003). A quantitative comparison of linear and non-linear models of motor cortical activity for the encoding and decoding of arm motions. *Neural Engineering, 2003. Conference Proceedings. First International IEEE EMBS Conference on*, pages 189–192.

Gelb, A. (1974). *Applied Optimal Estimation*. The MIT Press.

Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., and Massey, J. T. (1982). On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *Journal of Neuroscience*, 2(11):1527–1537.

Georgopoulos, A. P., Kettner, R. E., and Schwartz, A. B. (1988). 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*, 8(8):2928–37.

Georgopoulos, A. P., Schwartz, A. B., and Kettner, R. E. (1986). Neuronal population coding of movement direction. *Science*, 233(4771):1416–1419.

Ghosh, S. S., Tourville, J. A., and Guenther, F. H. (InPress). An fmri study of the overt production of simple sounds. *Journal of Speech, Language and Hearing Research*.

Grün, S., Diesmann, M., and Aertsen, A. (2001a). Unitary events in multiple single-neuron spiking activity: I. detection and significance. *Neural Computation*, 14:43–80.

Grün, S., Diesmann, M., and Aertsen, A. (2001b). Unitary events in multiple single-neuron spiking activity: Ii. nonstationary data. *Neural Computation*, 14:81–119.

Guenther, F. (1994). A neural network model of speech acquisition and motor equivalent speech production. *Biological Cybernetics*, 72(1):43–53.

Guenther, F. H. (1995). Speech sound acquisition, coarticulation, and rate effects in a neural network model of speech production. *Psychological Review*, 102(3):594–621.

Guenther, F. H., Ghosh, S. S., and Tourville, J. A. (2006). Neural modeling and imaging of the cortical interactions underlying syllable production. *Brain and Language*, 96(3):280–301.

Guenther, F. H., Hampson, M., and Johnson, D. (1998). A theoretical investigation of reference frames for the planning of speech movements. *Psychological Review*, 105(4):611–633.

Harris, K. D., Henze, D. A., Csicsvari, J., Hirase, H., and Buzsáki, G. (2000). Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. *Journal of Neurophysiology*, 84:401–414.

Hatsopoulos, N. G., Xu, Q., and Amit, Y. (2007). Encoding of movement fragments in the motor cortex. *Journal of Neuroscience*, 27(19):5105–5114.

Hochberg, L. R., Serruya, M. D., Friehs, G. M., Mukand, J. A., Saleh, M., Caplan, A. H., Branner, A., Chen, D., Penn, R. D., and Donoghue, J. P. (2006). Neuronal ensemble control of prosthetic devices by a human with tetraplegia. *Nature*, 442(7099):164–171.

Horsley, V. A. (1886). Brain surgery. *British Medical Journal*, 2:670–675.

Horsley, V. A. (1890). Remarks on the surgery of the central nervous system. *British Medical Journal*, 2:1286–1292.

Iyengar, S. and Liao, Q. (1997). Modeling neural activity using the generalized inverse Gaussian distribution. *Biological Cybernetics*, 77(4):289–295.

Kalaska, J., Cohen, D., Hyde, M., and Prud'homme, M. (1989). A comparison of movement direction-related versus load direction-related activity in primate motor cortex, using a two-dimensional reaching task. *Journal of Neuroscience*, 9(6):2080.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45.

Kass, R. E. and Ventura, V. (2001). A spike-train probability model. *Neural Computation*, 13:1713–1720.

Kass, R. E., Ventura, V., and Brown, E. N. (2005). Statistical issues in the analysis of neuronal data. *Journal of Neurophysiology*, 94:8–25.

Katkovnik, V. and Shmulevich, I. (2002). Kernel density estimation with adaptive varying window size. *Pattern Recognition Letters*, 23:1641–1648.

Kennedy, P., Andreasen, D., Bartels, J., Ehirim, P., Mao, H., Seibert, S., and Wright, E. J. (submitted). The longevity and stability of multiple neural signals recorded from human motor speech cortex using the neurotrophic electrode.

Kennedy, P. R. (1989). The cone electrode: a long-term electrode that records from neurites grown onto its recording surface. *Journal of Neuroscience Methods*, 29:181–193.

Kennedy, P. R. (2006). Comparing electrodes for use as cortical control signals: Tiny tines, tiny wires or tiny cones on wires: Which is best? In *The biomedical Engineering Handbook*, volume 1 of *The electrical engineering handbook series*. CRS/Taylor and Francis, Boca Raton, 3rd edition.

Kennedy, P. R., Andreasen, D., Bartels, J., Ehirim, P., Mao, H., Seibert, S., and Wright, E. J. (in press). Neurotrophic electrode: method of assembly and implantatin into human motor speech cortex. *Journal of Neuroscience Methods*.

Kennedy, P. R. and Bakay, R. A. E. (1998). Restoration of neural output from a paralyzed patient by direct brain connection. *NeuroReport*, 9:1707–1711.

Kennedy, P. R., Bakay, R. A. E., Moore, M. M., Adams, K., and Goldwaithe, J. (2000). Direct control of a computer from the human central nervous system. *IEEE Transactions on Rehabilitation Engineering*, 8(2):198–202.

Kennedy, P. R., Bakay, R. A. E., and Sharpe, S. M. (1992a). Behavioral correlates of action potentials recorded chronically inside the Cone Electrode. *NeuroReport*, 3:605–608.

Kennedy, P. R., Kirby, T., Moore, M. M., King, B., and Mallory, A. (2004). Computer control using human intracortical local field potentials. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 12(3):339–344.

Kennedy, P. R., Mirra, S. S., and Bakay, R. A. E. (1992b). The cone electrode: ultrastructural studies following long-term recording in rat and monkey cortex. *Neuroscience Letters*, 142:89–94.

Kennedy, P. R., Wright, E. J., Siebert, S., and Andreasen, D. (2006). Detecting patterns of neural signals from broca's area to produce speech in a locked-in subject. In *Neuroscience Meeting Planner 2006*, Program No. 256.14, Atlanta, GA. Society for Neuroscience.

Kettner, R. E., Schwartz, A. B., and Georgopoulos, A. P. (1988). Primate motor cortex and free arm movements to visual targets in three- dimensional space. III. Positional gradients and population coding of movement direction from various movement origins. *Journal of Neuroscience*, 8(8):2938–2947.

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

Kim, S. P., Simeral, J. D., Hochberg, L. R., Donoghue, J. P., Friehs, G. M., and Black, M. J. (2007). Multi-state decoding of point-and-click control signals from motor cortical activity in a human with tetraplegia. *Neural Engineering, 2007. CNE'07. 3rd International IEEE/EMBS Conference*, pages 486–489.

Klatt, D. H. (1980). Software for a cascade/parallel formant synthesizer. *Journal of the Acoustical Society of America*, 67(3):971–995.

Klatt, D. H. (1987). Review of text-to-speech conversion for English. *Journal of the Acoustical Society of America*, 82(3):737–793.

Kohler, E., Keysers, Christian, U., Fogassi, L., Gallese, V., and Rizzolatti, G. (2002). Hearing sounds, understanding actions: Action representation in mirror neurons. *Science*, 297:297–299.

Lewicki, M. S. (1998). A review of methods for spike sorting: the detection and classification of neural action potentials. *Network: Computation in Neural Systems*, 9:R53–R78.

Loftsgaarden, D. O. and Quesenberry, C. P. (1965). A nonparametric estimate of a multivariate density function. *The Annals of Mathematical Statistics*, 36(3):1049–1051.

Maeda, S. (1990). Compensatory articulation during speech: Evidence from the analysis and sythesis of vocal tract shapes using an articulatory model. In *Speech Production and Speech Modeling*. Kluwer Academic Publishers, Boston.

Miller, L. E., Andreasen, D. S., Bartels, J. L., Kennedy, P. R., Robesco, J., Siebert, S. A., and Wright, E. J. (2007). Human speech cortex long-term recordings [4]: bayesian analyses. In *Neuroscience Meeting Planner 2007*, Program No. 517.20, San Diego, CA. Society for Neuroscience.

Mitzdorf, U. (1985). Current source-density method and application in cat cerebral cortex: investigation of evoked potentials and EEG phenomena. *Physiological Reviews*, 65(1):37–100.

Moran, D. W. and Schwartz, A. B. (1999a). Motor cortical activity during drawing movements: population representation during spiral tracing. *Journal of Neurophysiology*, 82:2693–2704.

Moran, D. W. and Schwartz, A. B. (1999b). Motor cortical representation of speed and direction during reaching. *Journal of Neurophysiology*, 82:2676–2692.

Mukhopadhyay, S. and Ray, G. C. (1998). A new interpretation of nonlinear energy operator and its efficacy in spike detection. *IEEE Transactions on Biomedical Engineering*, 45(2):180–187.

National Institute of Neurological Disorders and Stroke (2008). Amyotrophic lateral sclerosis fact sheet. Retrieved June 9, 2008 from National Institutes of Health: National Institute of Neurological Disorders and Stroke: http://www.ninds.nih.gov/disorders/amyotrophiclateralsclerosis.

Neyman, J. and Pearson, E. S. (1933). On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231:289–337.

Obeid, I. and Wolf, P. D. (2004). Evaluation of spike-detection algorithms for a brain-machine interface application. *IEEE Transactions on Biomedical Engineering*, 51(6):905–911.

Olson, C. R., Gettner, S. N., Ventura, V., Carta, R., and Kass, R. E. (2000). Neuronal activity in macaque supplementary eye field during planning of saccades in response to pattern and spatial cues. *Journal of Neurophysiology*, 84:1369–1384.

Paninski, L., Fellows, M. R., Hatsopoulos, N. G., and Donoghue, J. P. (2004). Spatiotemporal tuning of motor cortical neurons for hand position and velocity. *Journal of Neurophysiology*, 91:515–532.

Park, Byeong, U. and Marron, J. S. (1990). Comparison of data-driven bandwidth selectors. *Journal of the American Statistical Association*, 85(409):66–72.

Parzen, E. (1962). On the estimation of a probability density function and the mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076.

Pazienti, A. and Grün, S. (2006). Robustness of the significance of spike synchrony with respect to sorting errors. *Journal of Computational Neuroscience*, 21(3):329–342.

Penfield, W. and Roberts, L. (1959). *Speech and brain-mechanisms*. Princeton University Press, Princeton, NJ.

Perkel, D. H., Gerstein, G. L., and Moore, G. P. (1967). Neuronal spike trains and stochastic point processes I: the single spike train. *Biophysical Journal*, 7(4):391–418.

Peterson, G. E. and Barney, H. L. (1952). Control methods used in a study of the identification of vowels. *Journal of the Acoustical Society of America*, 24:175–184.

Pipa, G. and Grün, S. (2003). Non-parametric significance estimation ofjoint-spike events by shuffling and resampling. *Neurocomputing*, 52(54):31–37.

Quiroga, R. Q., Nadasdy, Z., and Ben-Shaul, Y. (2004). Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. *Neural Computation*, 16:1661–1687.

Rabiner, L. and Juang, B. H. (1993). *Fundamentals of speech recognition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Raykar, V. C. and Duraiswami, R. (2006). Fast optimal bandwidth selection for kernel density estimation. *Proceedings of the sixth SIAM International Conference on Data Mining*, pages 524–528.

Roweis, S. and Ghahramani, Z. (1999). A unifying review of linear gaussian models. *Neural Computation*, 11(2):305–345.

Rutishauser, U., Schumann, E. M., and Mamelak, A. N. (2006). Online detection and sorting of extracellularly recorded action potentials in human medial temporal lobe recordings, in vivo. *Journal of Neuroscience Methods*, 154:204–224.

Schmidt, E. M. (1984). Computer separation of multi-unit neuroelectric data: a review. *Journal of Neuroscience Methods*, 12:95–111.

Schmitzer-Torbert, N., Jackson, J., Henze, D., Harris, K., and Redish, A. D. (2005). Quantitative measures of cluster quality for use in extracellular recordings. *Neuroscience*, 131:1–11.

Schwartz, A. B., Kettner, R. E., and Georgopoulos, A. P. (1988). Primate motor cortex and free arm movements to visual targets in three-dimensional space. I. relations between single cell discharge and direction of movement. *Journal of Neuroscience*, 8(8):2913–27.

Schwartz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(4):461–464.

Serruya, M. D., Hatsopoulos, N. G., Paninski, L., Fellows, M. R., and Donoghue, J. P. (2002). Instant neural control of a movement signal. *Nature*, 416(6877):141–2.

Sheather, S. J. and Jones, M. C. (1991). A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society, Seiries B (Methodological)*, 53(3):683–690.

Shenoy, K., Churchland, M., Santhanam, G., Yu, B., and Ryu, S. (2003). Influence of movement speed on plan activity in monkey pre-motor cortex and implications for high-performance neural prosthetic system design. *Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE*, 2:1897–1900.

Stark, E. and Abeles, M. (2007). Predicting movement from multiunit activity. *Journal of Neuroscience*, 27(31):8387–8394.

Stevens, K. N. (2000). *Acoustic Phonetics*. MIT Press.

Suner, S., Fellows, M. R., Irwin, V. C., Nakata, G. K., and Donoghue, J. P. (2005). Reliability of signals from a chronically implanted, silicon-based electrode array in non-human primate primary motor cortex. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 13(4):524–541.

Tan, C. O., Taylor, J. A., Ler, A. S. H., and Cohen, M. A. (in press). Detection of multifiber neuronal firings: a mixture separation model applied to sympathetic recordings. *IEEE Transactions on Biomedical Engineering*.

Taylor, D. M., Tillery, S. I. H., and Schwartz, A. B. (2002). Direct cortical control of 3D neuroprosthetic devices. *Science*, 296(5574):1829–1832.

Terrell, G. R. and Scott, D. W. (1992). Variable kernel density estimation. *The Annals of Statistics*, 20(3):1236–1265.

Thorpe, S., Delorme, A., and Van Rullen, R. (2001). Spike-based strategies for rapid processing. *Neural Networks*, 14(6-7):715–725.

Tibshirani, R., Walther, G., and Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423.

Tourville, J. A., Reilly, K. J., and Guenther, F. H. (2008). Neural mechanisms underlying auditory feedback control of speech. *Neuroimage*, 39(3):1429–1443.

Truccolo, W., Eden, U. T., Fellows, M. R., Donoghue, J. P., and Brown, E. N. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble and extrinsic covariate effects. *Journal of Neurophysiology*, 93:1074–1089.

Tuckwell, H. C. (1988). *Introduction to Theoretical Neurobiology*, volume 2. Cambridge University Press, New York.

Velliste, M., Perel, S., Spalding, M. C., Whitford, A. S., and Schwartz, A. B. (2008). Cortical control of a prosthetic arm for self-feeding. *Nature*, 453(7198):1098–1101.

von Kempelen, W. R. (1791). Mechanismus der menschlichen Sprache nebst Beschreibungeiner sprechenden Maschine. *Stuttgart-Bad Cannstatt: Friedrich Frommann Verlag.(Facsimile print, 1970)*.

Warland, D. K., Reinagel, P., and Meister, M. (1997). Decoding visual information from a population of retinal ganglion cells. *Journal of Neurophysiology*, 78:2336–2350.

Wessberg, J., Stambaugh, C. R., Kralik, J. D., Beck, P. D., Laubach, M., Chapin, J. K., Kim, J., Biggs, S. J., Srinivasan, M. A., and Nicolelis, M. A. (2000). Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature*, 408(6810):361–5.

Wood, F., Black, M. J., Vargas-Irwin, C., Fellows, M., and Donoghue, J. P. (2004). On the variability of manual spike sorting. *IEEE Transactions on Biomedical Engineering*, 51(6):912–918.

Wright, E. J., Andreasen, D. S., Bartels, J. L., Brumberg, J. S., Guenther, F. H., Kennedy, P. R., Miller, L. E., Robesco, J., Schwartz, A. B., Siebert, S. A., and Velliste, M. (2007). Human speech cortex long-term recordings [3]: neural net analyses. In *Neuroscience Meeting Planner 2007*, Program No. 517.18, San Diego, CA. Society for Neuroscience.

Wu, W., Black, M. J., Gao, Y., Bienenstock, E., Serruya, M., and Donoghue, J. P. (2002). Inferring hand motion from multi-cell recordings in motor cortex using a kalman filter. *Workshop on Motor Control in Humans and Robots: On the Interplay of Real Brains and Artificial Devices*, pages 66–73.

Wu, W., Black, M. J., Gao, Y., Bienenstock, E., Serruya, M., Shaikhouni, A., and Donoghue, J. P. (2003). Neural decoding of cursor motion using a Kalman filter. *Advances in Neural Information Processing Systems*, 15:133–140.

Wu, W., Gao, Y., Bienenstock, E., Donoghue, J. P., and Black, M. J. (2006). Bayesian population decoding of motor cortical activity using a kalman filter. *Neural Computation*, 18:80–118.

CURRICULUM VITÆ

Jonathan S. Brumberg

Department of Cognitive and Neural Systems
Boston University
Boston, MA 02215
(617) 353-6181
email: brumberg@cns.bu.edu
web: http://cns.bu.edu/~brumberg

EDUCATION

Ph.D. Cognitive and Neural Systems
Boston University, Boston, Massachusetts, 2003–2008
Dr. Frank Guenther, advisor
Thesis entitled: "An electrophysiological investigation of human motor cortex and its application to speech restoration"

Bachelor of Science in Computer Science
University of Delaware, Newark, Delaware, 2002

Bachelor of Arts in Philosophy
University of Delaware, Newark, Delaware, 2002

RESEARCH EXPERIENCE

2003–present **Research Assistant, Department of Cognitive and Neural Systems**
Boston University, Boston, Massachusetts

- Investigation of the neural correlates of speech production and applications toward development of neural prostheses.
- Signal processing and analysis of neurobiological signals including extracellular voltage potentials and magnetoencephelography.
- Extensive computational modeling of a neural network for simulation of speech production.

2002–2003 **Research Assistant, Psychology Department**
Temple University, Philadelphia, Pennsylvania
- Supervised by Dr. Thomas F. Shipley
- Performed and analyzed experiments pertaining to perception of Biological Motion and Illusory Contours

Summer 2002 **Research Assistant, Psychology Department**
University of Delaware, Newark, Delaware
- Supervised by Dr. John Whalen and Dr. James Hoffman
- Performed and analyzed ERP studies with human subjects using EGI Geodosic Sensor Net

Fall 2000 **Research Assistant, Department of Computer and Information Sciences**
University of Delaware, Newark, Delaware
- Created new course manual for Computer and Information Sciences Department, Introduction to Computer and Information Sciences

TEACHING EXPERIENCE

Fall 2004 **Teaching Fellow, Department of Cognitive and Neural Systems**
Boston University, Boston, Massachusetts
- Lab instructor for graduate-level introduction to mathematical methods for computational sumlation and signal processing. Instructed students on methods for solving differential equations (ODE).

2000-2002 **Teaching Assistant, Computer and Information Sciences Department**
University of Delaware, Newark, Delaware
- Instructed class as Teaching Assistant for undergraduate-level courses, ranging from beginner to intermediate level, including: Introduction to Computer and Information Science, General Computer Science (C programming), Introduction to Computer Science (C++ programming), Machine Organization and Assembly Language (Assembly Language, SPARC architecture)

PUBLICATIONS

1. Brumberg, J. S., Andreasen, D. S., Bartels, J. L., Guenther, F. H., Kennedy, P. R., Siebert, S. A., Schwartz, A. B., Velliste, M. and Wright, E. J. Human speech cortex long-term recordings [5]: formant frequency analyses. Program No. 517.17. 2007 Neuroscience Meeting Planner San Diego, CA: Society for Neuroscience, 2007. Online.

2. Wright, E. J., Andreasen, D. S., Bartels, J. L., Brumberg, J. S., Guenther, F. H., Kennedy, P. R, Miller, L., Robesco, J., Schwartz, A. B., Siebert, S. A. and Velliste, M. Human speech cortex long-term recordings [3]: neral net analyses. Program No. 517.18. 2007 Neuroscience Meeting Planner San Diego, CA: Society for Neuroscience, 2007. Online.

3. Siebert, S. A. Andreasen, D. S., Bartels, J. L., Brumberg, J. S., Guenther, F. H., Kennedy, P. R. and Wright, E. J. Human speech cortex long-term recordings [1]: spike sorting and noise reduction. Program No. 728.14. 2007 Neuroscience Meeting Planner San Diego, CA: Society for Neuroscience, 2007. Online.

4. Terband, H., Maassen, B. and Brumberg, J. Motor speech in adults and children: computational-neurological modeling of childhood apraxia of speech (CAS). (2007) Poster presented at American Speech-Language Association Conference 2007. Boston, MA.

5. Shipley, T. F., Maguire, M. J. and Brumberg, J. (2004) Segmentation of event paths. *Journal of Vision.* 4(8), 562.

6. Shipley, T. F., Maguire, M. J. and Brumberg, J. S. (2003) Top down effects on search for biological motion. *Abstracts of the Psychonomics Society*, 8, 51.

SKILLS

- Advanced knowledge of computer and web programming languages including C, C++, Matlab, Pascal, Java, Perl/CGI, PHP, HTML, LaTeX, SQL and Microsoft .NET (C# and VB)
- Administration of Linux server clusters and Microsoft Windows Server
- Extensive experience using Microsoft Office suite and Adobe Photoshop
- Proficiency with statistical analysis software packages S/R

ACTIVITIES

- Student Member, Society for Neuroscience, 2007–2008
- Member, National Student Speech Language Hearing Association, 2007–2008
- Member, University of Delaware Honors Program, 1998–2002
- Member, Theta Chi Fraternity, Alpha Xi Chapter, 1999–present