2020

Experimental demonstration of single neuron specificity during underactuated neurocontrol

BOSTON UNIVERSITY

COLLEGE OF ENGINEERING

Dissertation

EXPERIMENTAL DEMONSTRATION OF SINGLE NEURON SPECIFICITY DURING UNDERACTUATED NEUROCONTROL

by

SAMUEL GARRETT BROWN

B.S., University of Rochester, 2014

Submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

2020

Approved by

First Reader

Jason Ritt, Ph.D.
Scientific Director of Quantitative Neuroscience
Robert J. & Nancy D. Carney Institute for Brain Science

Assistant Professor of Neuroscience
Brown University

Second Reader

John A. White, Ph.D.
Professor and Chair of Biomedical Engineering

Third Reader

Uri T. Eden, Ph.D.
Professor of Mathematics and Statistics

Fourth Reader

Kamal Sen, Ph.D.
Associate Professor of Biomedical Engineering

Fifth Reader

ShiNung Ching, Ph.D.
Associate Professor of Electrical and Systems Engineering
Washington University in St. Louis

We are all aware that the senses can be deceived, the eyes fooled. But how can we be sure our senses are not being deceived at any particular time, or even all the time? Might I just be a brain in a tank somewhere, tricked all my life into believing in the events of this world by some insane computer? And does my life gain or lose meaning based on my reaction to such solipsism?

Project PYRRHO, Specimen 46, Vat 7. Activity recorded M.Y. 2302.22467.

(TERMINATION OF SPECIMEN ADVISED)

Sid Meier's Alpha Centauri, 1999

DEDICATION

I would like to dedicate this work to my lovely partner, Cassandra, without whose help and support this wouldn't have been finished, to my parents Gwen and James, without whose love and inspiration this would never have been started, and to my cat Huxley, without whose constant interruption and indifference this would have been done 3 months sooner.

ACKNOWLEDGMENTS

The past 5 years have been a whirlwind of people, labs, and free food, and I imagine I could fill the space up to Chapter 2 with people who have helped me and supported me through this adventure. Given my space limit and your attention span, however, I would like to single a few people out. First, I must thank my advisor Jason Ritt, and my adopted advisor John White, who not only took great care of me and ensured that I turned into a much better scientist than I was when I began, but who also made sure I never had to be stressed about money or space, both of which were at a premium during my time at BU. I must also thank the past members of the Ritt lab, Mike Palmiere, Su Kim, Shuo Huang, and Smrithi Sunil, all of whom helped keep the lab running smoothly and would answer even the most ridiculous of my questions. Further, my gratitude goes to the members of my adopted lab, Bahar Rahsepar, Jad Noueihed, Jacob Norman, Kevin Ghaemi, and Fernando Fernandez, whose support kept me sane, and whose antics kept me saner. I would like to thank my Committee, consisting of my advisors, as well as Uri Eden, Kamal Sen, and ShiNung Ching, who have provided me wonderful feedback on my work and who are greatly appreciated. Thank you to my partner Cassie, to my parents Gwen and James, and to the rest of my family and friends who helped me through this work, have helped me through many things before, and I get the feeling will help me through many more things in the future, too. Finally, I would like to thank the mice that I used in this study, without whose sacrifice this work would not have been possible, to whom gratitude is meaningless, but whose acknowledgement keeps us human.

EXPERIMENTAL DEMONSTRATION OF SINGLE NEURON SPECIFICITY DURING UNDERACTUATED NEUROCONTROL

SAMUEL GARRETT BROWN

Boston University, College of Engineering, 2020

Major Professor: Jason Ritt, Ph.D., Scientific Director of Quantitative Neuroscience, Robert J. & Nancy D. Carney Institute for Brain Science; Assistant Professor of Neuroscience, Brown University

ABSTRACT

Population-level neurocontrol has been advanced predominately through the miniaturization of hardware, such as MEMS-based electrodes. However, miniaturization alone may not be viable as a method for single-neuron resolution control within large ensembles, as it is typically infeasible to create electrode densities approaching 1:1 ratios with the neurons whose control is desired. That is, even advanced neural interfaces will likely remain underactuated, in that there will be fewer inputs (electrodes) within a given area than there are outputs (neurons). A complementary "software" approach could allow individual electrodes to independently control multiple neurons simultaneously, to improve performance beyond naïve hardware limits. An underactuated control schema, demonstrated in theoretical analysis and simulation (Ching & Ritt, 2013), uses stimulus strength-duration tradeoffs to activate a target neuron while leaving non-targets inactive. Here I experimentally test this schema in vivo, by independently controlling pairs of cortical neurons receiving common optogenetic input, in anesthetized mice. With this approach, neurons could be specifically and independently controlled following a short (~3 min) identification procedure. However, drift in neural responsiveness limited the

performance over time. I developed an adaptive control procedure that fits stochastic Integrate and Fire (IAF) models to blocks of neural recordings, based on the deviation of expected from observed spiking, and selects optimal stimulation parameters from the updated models for subsequent blocks. I find the adaptive approach can maintain control over long time periods (>20 minutes) in about 30% of tested candidate neuron pairs. Because stimulation distorts the observation of neural activity, I further analyzed the influence of various forms of spike sorting corruption, and proposed methods to compensate for their effects on neural control systems. Overall, these results demonstrate the feasibility of underactuated neurocontrol for in vivo applications as a method for increasing the controllable population of high density neural interfaces.

TABLE OF CONTENTS

DEDICATION v

ACKNOWLEDGMENTS vi

ABSTRACT vii

TABLE OF CONTENTS ix

LIST OF TABLES xii

LIST OF FIGURES xiii

LIST OF ABBREVIATIONS xv

1 INTRODUCTION 1

1.1 Neurocontrol is an Important Aspect of Clinical Neuro-engineering 1

1.2 Existing Neurocontrol Methods do not Address Underactuation 2

1.3 Contributions of this Dissertation 4

2 IN VIVO APPLICATION OF UNDERACTUATED CONTROL 5

2.1 Introduction 5

2.1.1 An Underactuated Control Schema Motivated by Neural Dynamics 5

2.1.2 Requirements for Mutual Controllability 9

2.1.3 Considerations for in vivo Implementation 11

2.1.4 Pilot Studies 14

2.2 Experimental Preparation 17

2.2.1 Hardware Setup 17

2.2.2 Surgical Preparation and Search for Units 19

2.3 Computational Methods 21

2.3.1 Table of IAF spike probabilities 21
2.3.2 Error Function (θ Optimization) 25
2.3.3 Cost Function (GT Optimization) 25
2.4 System Identification and Control 27
2.4.1 Initial Characterization 27
2.4.2 Adaptive Control 30
2.5 Data Analysis Methods 32
2.5.1 The Response Fraction 32
2.5.2 The Control Quality Metric 34
2.5.3 Confirmation of Driving Responses Towards Targets 36
2.6 Results 36
2.6.1 Summary of Results 36
2.6.2 Example of a Controllable Pair 38
2.6.3 Common Failure Modes During the Experiment 42
2.7 Discussion 45
2.7.1 Comparison to Other Neurocontrol Studies 45
3 CONTROL WITH CORRUPTION 48
3.1 Introduction 48
3.1.1 The Importance of Observability During Neurocontrol 48
3.1.2 Types of Spike Sorting Corruption 49
3.2 Methods 51
3.2.1 Rescaling of the Integrate and Fire Model to Explore Corruption 51

3.2.2 A Probability-Based Framework for Modeling Corruption 52
3.2.3 Spike Exclusion 53
3.2.4 Spike Addition 54
3.2.5 Spike Deletion 55
3.2.6 Neuron and Hash Models 56
3.3 Results 58
3.3.1 Exploration of Corruption Paradigms 58
3.3.2 Exploration of Spike Exclusion 60
3.3.3 Exploration of Spike Addition 61
3.3.4 Exploration of Spike Deletion 65
3.3.5 Boundaries in neural parameter space 71
3.3.6 Parameter Boundaries of Spike Exclusion 76
3.3.7 Parameter Boundaries of Spike Addition 78
3.3.8 Parameter Boundaries of Spike Deletion 81
3.4 Discussion 83
3.4.1 The Use of Spike Sorting in Neurocontrol 83
3.4.2 Detecting and Compensating for Corruption 85
4 DISCUSSION 89
4.1 Limitations of the Current Study 89
4.2 Implications for Clinical Neuro-control 93
BIBLIOGRAPHY 94
CURRICULUM VITAE 101

LIST OF TABLES

Table 1: Parameters used during Fokker-Planck integration and associated database calculation 23

Table 2: Definitions of the Response Fractions 33

Table 3: IAF parameters for the corrupting neuron NC 59

Table 4: Logistic parameters for the corrupting hash 59

LIST OF FIGURES

Figure 1: Stimulation in an underactuated system 5

Figure 2: A path of the Integrate and Fire (IAF) neuron model 7

Figure 3: The SD curves of a mutually controllable neuron pair, with stimuli designed to bias the neurons' activity 9

Figure 4: Mutual controllability criterion for deterministic neurons 10

Figure 5: Nonstationarity leads to SD Curve drift 15

Figure 6: A schematic of the experimental setup 18

Figure 7: Shape of the cost function in SD space 26

Figure 8: An outline of the characterization and optimization protocol 32

Figure 9: $\boldsymbol{CQ}$ for each tested pair (n=29) 37

Figure 10: Out of the n=29 pairs tested, 27 responded with high selectivity to the stimulus. 38

Figure 11: The results of an example controllable pair 39

Figure 12: Spike waveforms of an example controllable pair 40

Figure 13: The SD curves of a neuron under exclusion corruption, with lines colored by their value of $\boldsymbol{P}_{\boldsymbol{E}_{\boldsymbol{Ex}}}$ 61

Figure 14: The SD curves of a neuron under addition corruption from another neuron with identical parameters, with lines colored by their value of $\boldsymbol{P}_{\boldsymbol{E}_{\boldsymbol{Inc}}}$ 62

Figure 15: The SD curves of a neuron under addition corruption from another neuron that has higher sensitivity, with lines colored by their value of $\boldsymbol{P}_{\boldsymbol{E}_{\boldsymbol{Inc}}}$. 64

Figure 16: The SD curves of a neuron under addition corruption from another neuron that has lower sensitivity, with lines colored by their value of $\boldsymbol{P}_{\boldsymbol{E}_{\boldsymbol{Inc}}}$. 65

Figure 17: The SD curve of a neuron under deletion corruption by hash with comparable sensitivity to the neuron, with lines colored by their value of $\boldsymbol{P}_{\boldsymbol{E}_{\boldsymbol{Del}}}$ 66

Figure 18: The SD curve of a neuron under deletion corruption by hash with high sensitivity, with lines are colored by their value of $\boldsymbol{P}_{\boldsymbol{E}_{\boldsymbol{Del}}}$. No SD curve is defined for high levels of corruption 68

Figure 19: The SD curve of a neuron under deletion corruption by hash with low sensitivity, with lines colored by their value of $\boldsymbol{P}_{\boldsymbol{E}_{\boldsymbol{Del}}}$. No SD curves (or only partial SD curves) are defined for high levels of corruption. 69

Figure 20: The SD curve of a neuron under deletion corruption by hash with high sensitivity, including the "doubled" SD curves. 70

Figure 21: The thresholds of losing controllability due to corruption affecting either the fast (blue) or slow (red) neuron. 72

Figure 22: A demonstration of the two control failure modes during corruption..... 73

Figure 23: The maximum $\boldsymbol{P_{Exc}}$, indicated by color, that can be tolerated by a neuron parameterized by $\boldsymbol{r\theta T} = [\boldsymbol{\alpha T}, \boldsymbol{\beta T}, \mathbf{1}]$ when being controlled with a standard bearer neuron. 77

Figure 24: The maximum $\boldsymbol{P_{Inc}}$ that can be tolerated by a non-target neuron, across parameterizations of the corruptor neuron, when being controlled with a standard bearer neuron. 80

Figure 25: The maximum $\boldsymbol{P_{Hash}}$, indicated by color, that can be tolerated by a target neuron, controlled with the standard bearer neuron, at a given GT, with different parameterizations of the hash..... 83

Figure 26: A set of three simulated neurons that are mutual controllable 91

Figure 27: A system which uses spatial encoding to address multiple cells in an underactuated system 92

LIST OF ABBREVIATIONS

CDF – Cumulative Distribution Function

CQ – Control Quality (Metric)

DBS – Deep Brain Stimulation

FP – Fokker Planck

IAF – Integrate and Fire

PDF – Probability Distribution Function

RF – Response Fraction

RFD – Response Fraction Difference

SD – Strength-Duration (Curve)

1 INTRODUCTION

1.1 Neurocontrol is an Important Aspect of Clinical Neuro-engineering

Direct electrical interaction with neural tissue underlies a broad array of research and clinical applications. Current technology allows insight into brain function by "reading out" information with increasingly high specificity (Burridge & Ladouceur, 2001; Hatsopoulos & Donoghue, 2009; Vidal et al., 2016; Wang et al., 2019). However, methods for inducing specific, complex neural activity through stimulation, or "writing in" information, lag behind (Wolff & Ölveczky, 2018). Emulating natural activity patterns requires the ability to address small populations within larger ensembles, perhaps even down to the individual cell level, which is not achievable with current neural stimulation methods. Technology underlying techniques such as deep brain stimulation (DBS) are probably insufficient for applications such as creating realistic artificial percepts, due to the complex activity patterns that may be required to mimic sensory stimulation.

A major approach to increasing neural stimulation selectivity is to increase the density of electrodes in the region of interest, in the hopes of gaining finer control over which neurons are activated. Unlike purely recording electrode arrays, such as Neuropixels (Jun et al., 2017), bi-directional high density micro electrode arrays, or HDMEAs, feature circuitry for both stimulation and recording, and allow high resolution control over a neural population (Eversmann et al., 2011; Frey et al., 2010). However, barring a major innovation in electrode technology, a close to 1:1 ratio of electrodes to neurons is unlikely to be achieved in dense neural tissue. For the immediate future,

controlling neural populations with cell-level specificity remains an under-actuated problem, meaning that fewer inputs (electrodes or optical fibers) exist in the system than dynamical elements in the plant (neurons) to be controlled.

1.2 Existing Neurocontrol Methods do not Address Underactuation

Inducing desired activity in the brain is a key step towards generating artificial percepts. It has previously been shown that stimulating sensory areas of the brain leads to percepts associated with that region's function, localized to the area on the body represented by the somatic mapping of that region (Ostrowsky et al., 2002; Schmidt et al., 1996). However, the application of artificial percepts in these areas is imprecise due to the limitations of modern stimulation technologies (Palanker et al., 2005), such as their electrode contact density (Zeng, 2017).

Despite these limitations, several neural control strategies have been proposed and successfully implemented, such as a single-cell resolution, activity-guided system using two-photon stimulation (Rickgauer et al., 2014), and model-free control systems for enforcing both static (Newman et al., 2015) and dynamic (Bolus et al., 2018) firing rate targets. Some studies also address control of larger populations, such as attempting to synchronize (Mitchell & Petzold, 2018) or desynchronize (Nabi & Moehlis, 2011) the activity of large neuron ensembles using input to only a single neuron in the population. Further, closed-loop DBS systems increase the efficacy of Parkinson's and essential tremor treatments, both in simulation (Santaniello et al., 2011) and in vivo (Rosin et al., 2011), when compared to their open-loop counterparts.

This small selection from the large body of the existing neurocontrol literature demonstrates the great strides that neuro-stimulation has taken in recent decades. However, none of these control strategies were designed to address the under-actuation problem due to the limited electrode density of modern stimulation hardware. While they offer various methods to increase the precision of the induced activity level, they do not address the specificity of neuron selection. There is little crossover between works that address high resolution stimulation and those that use hardware that is viable for use in wearable medical devices.

Generally, each electrode in an array or probe is able to control a single neuron, or the global activity of a single population. Therefore, the most common approach to increase stimulation specificity is to use hardware with higher stimulating electrode density, thereby increasing the number of neurons that can be targeted. However, there is a limit to the density with which we can manufacture electrode arrays using modern methods, and this limit falls far below the threshold of full actuation, or a 1:1 electrode-to-neuron ratio. This motivates a new way of approaching neural control for the purposes of inducing complex activity, such as that required for delivering artificial percepts, in these underactuated conditions.

While it has not been a primary focus in the field, some studies directly address the issue of underactuated stimulation, by considering oscillating phase models (Li et al., 2012) or IAF neurons (Ching & Ritt, 2013; Nandi et al., 2017) coupled by a common input. However, translation of these methods from computational to in vivo work has not yet been demonstrated. This transition introduces complications, such as the imprecision

of applying a linear neural model to a non-linear biological system, in addition to noise and other challenges in electrophysiology recordings. Therefore, before this method can be used in translational applications, issues related to its implementation in vivo must first be solved.

1.3 Contributions of this Dissertation

In this dissertation, I address the need to individually control a number of cells in a population beyond limitations in stimulation hardware. In Chapter 2, I present a method for performing underactuated control in vivo, adapting the control scheme proposed by Ching & Ritt (2013) from its previous in silico implementation. In Chapter 3, I explore some of the observability concerns encountered when performing single unit isolation following stimulation, and model worst-case corruption of measured neural responses compared to true responses. Together, these results are a step towards feasibility of underactuated control as a method to increase the effective dimensionality of high channel-count stimulating neural interfaces.

2 IN VIVO APPLICATION OF UNDERACTUATED CONTROL

2.1 Introduction

2.1.1 An Underactuated Control Schema Motivated by Neural Dynamics

To achieve underactuated control on some population of neurons, it must be possible to modulate the input(s) to the system in such a way to individually address the cells in the population. This modulation could be spatial, such as illuminating multiple areas in different combinations, or temporal, in which the timing of each stimulus or the modulation of its amplitude encodes the identity of the neuron to be activated.

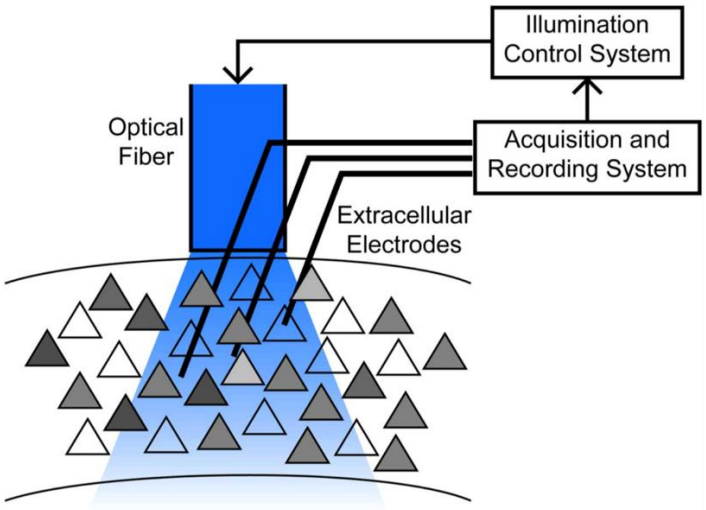

Figure 1: Stimulation in an underactuated system

A common single input (the light from the optical fiber) is shared by each neuron within the blue light cone. Multiple extra cellular electrodes allow the control system to record from multiple neurons within the cone simultaneously. The control system will attempt to modulate the single light source to move multiple neurons into the desired states simultaneously.

The control schema proposed by Ching & Ritt (2013) uses temporal encoding, using trade-offs between the power and duration of a laser pulse to address a targeted

neuron in a population using only a single input, as illustrated in Figure 1. The mechanism behind this encoding relies on the dynamical responses of neurons with different membrane properties undergoing identical stimulation. The control technique was devised based on the leaky, noisy, integrate and fire (IAF) neuron model (Dayan & Abbott, 2001).

Suppose V is the membrane potential of a neuron, $S(t)$ is some stimulus, α is the leakiness of the neuron's membrane, β is the neuron's sensitivity to the stimulus, σ is the intensity of the intrinsic noise in the membrane potential, dW is a standard Weiner process, and V_T is some membrane potential threshold. The IAF neuron model is described by

$$\frac{dV}{dt} = -\alpha V + \beta S(t) + \sigma dW \qquad (1)$$

$$\text{When } V = V_T, V \to 0, \text{ defined as a spike}$$

A sample path of the IAF model is demonstrated in Figure 2.

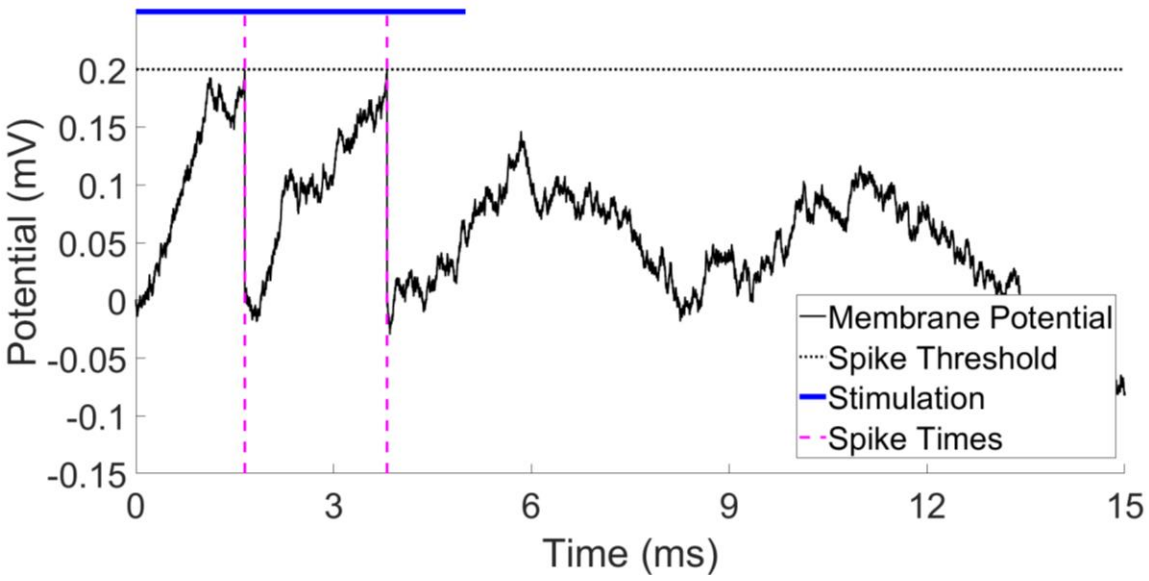

Figure 2: A path of the Integrate and Fire (IAF) neuron model

An example path of an IAF neuron is shown in black. The neuron is subject to stimulation for the first 5 seconds, indicated by the blue line. When the path crosses the spike threshold at 0.2 mV, the neuron is assumed to have spiked, indicated by the vertical purple dotted line.

Consider an IAF neuron N_A parameterized by $\theta_A = [\alpha_A, \beta_A, \sigma_A]$. Because the IAF model is a linear model, the optimal input $S(t)$ to cause the neuron to spike in the shortest amount of time is an impulse function. However, when considering a physiologically useful model for the input, in which our laser power is limited so that it does not damage the tissue, a more reasonable choice of $S(t)$ is a pulse, parameterized by the strength-duration pair $[G, T]$. Due to IAF's linearity, pulse inputs (or a "bang-bang control" input) are time optimal (Dorato et al., 1967; Nandi et al., 2017).

Consider then the situation in which we would like to cause N_A to spike with some probability P. For short durations T, relatively large strengths G will be required to achieve firing probability P. Conversely, for long durations T, relatively small strengths

G will be required. There exists a trade-off between the strength and the duration of a stimulus that can cause the same firing probability, and this behavior naturally leads to the concept of the strength-duration (SD) curve, which I define as the set of all points in strength-duration space that cause the neuron to fire with given probability P. Throughout, I will consider the curve generated by choosing $P = 0.5$.

Suppose now that we have a second neuron N_B parameterized by $\theta_B = [\alpha_B, \beta_B, \sigma_B]$ that we would like to control simultaneously with N_A using the common input $S(t)$. Such control is possible according to the proposed schema if the SD curves of the two neurons cross each other, as explained by the following.

As shown in Figure 3, such a crossing cuts the SD plane into four distinct regions. The top region, above both curves, contains stimuli which cause both neurons to spike with high probability. The bottom region, below both curves, contains stimuli which induce low spike probability in both neurons. The interesting pair of regions is between the curves, on either side of the intersection point. One region is “high duration”, and the other “low duration”. A stimulus in either of these regions will cause a spike in one (target) neuron with probability $P_T > .5$, while the other (non-target) neuron spike probability satisfies $P_{NT} < .5$. In this way, stimuli may be chosen that can bias activity toward either neuron as a target, while the non-target neuron has a lower probability of firing.

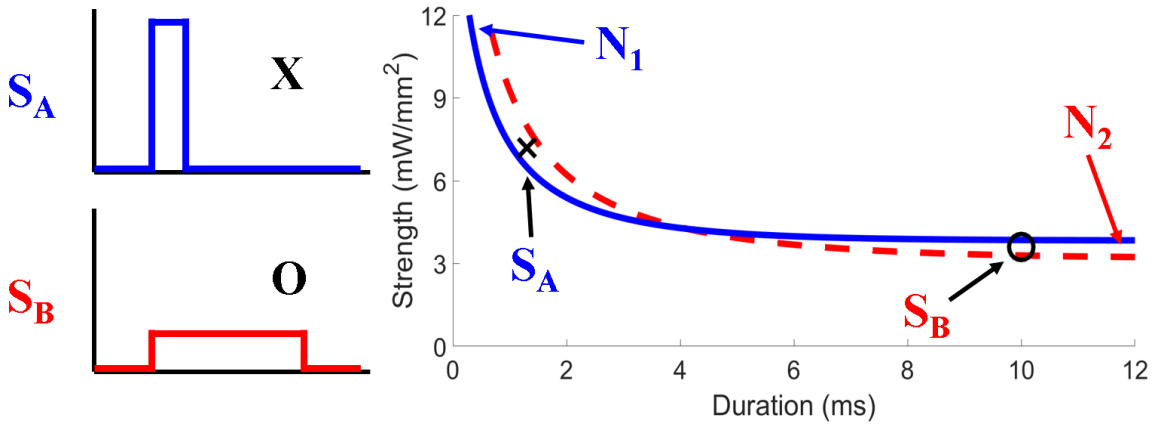

Figure 3: The SD curves of a mutually controllable neuron pair, with stimuli designed to bias the neurons' activity

Stimuli chosen from the regions between the two curves, to the left or right of the intersection point, will bias activity towards one neuron in the pair. One neuron fires more often in response to low duration input (N_A), while the other responds more often to high duration input (N_B).

2.1.2 Requirements for Mutual Controllability

This graphical intuition can also be represented in terms of the IAF neural parameters. I assume the neurons have non-identical parameters (i.e. they are heterogeneous), and I choose the indices so that neuron N_A is the unit whose α is largest. With this convention, Ching & Ritt (2013) showed in the deterministic case ($\sigma = 0$) that the SD curves of both neurons N_A and N_B will cross if and only if

$$\beta_A > \beta_B \tag{2}$$

$$\frac{\alpha_A}{\beta_A} > \frac{\alpha_B}{\beta_B} \tag{3}$$

In α-β space, this means that, given some deterministic neuron described by $[\alpha_S, \beta_S]$, the values of a mutually controllable deterministic neuron $[\alpha, \beta]$ can be at any point represented in blue in Figure 4. The mutually controllable region for the stochastic case is more limited (Huang, 2019), but the deterministic case is useful as a general rule.

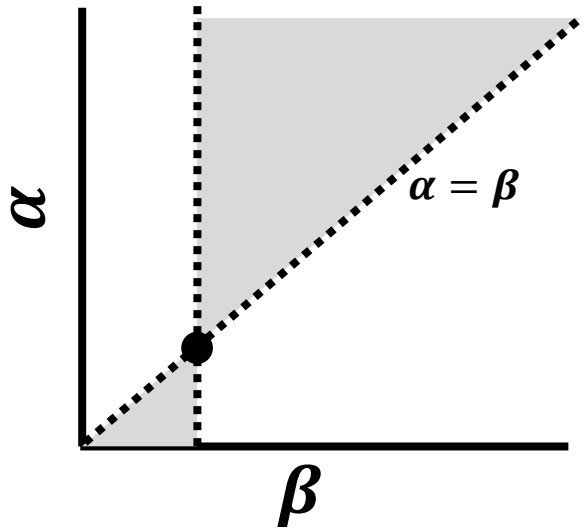

Figure 4: Mutual controllability criterion for deterministic neurons

A deterministic neuron parameterized by $[\alpha, \beta]$ drawn from the gray region is mutually controllable with the deterministic neuron parameterized by $[\alpha_S, \beta_S]$, the black dot.

In SD space, we may draw a stimulation from one of the regions between the stochastic neurons' SD curves to bias neural activity. If a stimulation is drawn from the left inter-curve region, it will bias activity towards N_A, and I will call that stimulation S_A. If a stimulation is drawn from the right inter-curve region, it will bias activity towards N_B, and I will call that stimulation S_B. I will therefore consider the two neurons, N_A and N_B, as well as the two stimuli used to bias their activity, S_A and S_B.

Due to the convention on the identities of N_A and N_B, S_A will always take the form of a short but strong stimulation, and S_B will always take the form of a long but weak stimulation. Throughout this dissertation, N_A will be represented by blue, and N_B will be represented by red. Additionally, where appropriate, S_A will be represented by crosses ("X"), while S_B will be represented by circles ("O").

2.1.3 Considerations for in vivo Implementation

Transitioning this schema for underactuated neurocontrol to an in vivo application presents a number of challenges that did not need to be considered when testing in silico, despite previous efforts incorporating noise, limited controllability, and using only spike times as observations (Ching & Ritt, 2013). Aside from normal instrumentation concerns, such as electrical noise in electrophysiology recordings, the primary problems faced during the transition to an animal model involved the intrinsic noise of the recorded neuron, and the limited observability of its state.

Observability and controllability are classic cornerstones to effective control (Kalman, 1959). The observability of a system describes the ability to determine the system's intrinsic state using only the outputs of the system in combination with any controllable inputs. It describes how easily the system's state can be understood based on its behavior. The controllability of a system (or more specifically, the state controllability) describes the ability of the system's inputs to drive the system between states. It describes how effective the inputs to the system are at moving it between states.

The parameters that underlie the behavior of the system we are trying to control are the IAF model parameters, $\theta = [\alpha, \beta, \sigma]$, which govern a linear approximation of the non-linear behavior of a real neuron.

For a neural system to be observable, it must be possible to infer the system's intrinsic state based on its input and output. There exist electrophysiology methods that support continuous measurement of membrane potential for recorded neurons, such as patch clamping (Sakmann & Neher, 1984), but such intracellular methods are infeasible

in most in vivo settings, particularly in high density electrode arrays and clinical applications. I use extracellular recording methods that, while comparatively easier to implement, give very limited information about the neuron. These methods effectively yield a binary observation of the neuron's state: whether in any time window it emits an action potential or not. Using methods that will be covered later, it is possible to make reasonable estimates for the IAF model parameters, based on how the neuron behaves when subjected to various stimuli. However, it is difficult to make a precise estimate of the membrane potential due to the noise intrinsic to the neural system (Meng et al., 2011). It is impossible to get a direct reading of the membrane potential of a cell using extracellular electrodes. However, spike times can be recorded. Therefore, I eliminate the membrane potential as a parameter, making the assumption that the neuron is near rest potential at the initial condition.

This noisiness and unpredictability have implications for the controllability of the neuron. The input to the system that I will be considering is the optogenetic light-driven input $S(t)$, as a current across the membrane. I assume that, over short timespans (over one second), the model parameters describing the neuron do not change. The illumination is, however, able to influence the membrane potential, though only in a positive direction given the positive reversal potential of the input conductance for ChR2 (Boyden et al., 2005).

During single unit control, it is generally desirable to maximize the change in the membrane potential $\frac{dV}{dt}$ by applying a large laser input $S(t)$. This increases the influence of the deterministic part of the system relative to the stochastic behavior, thereby

decreasing both the time to spike and the variance in spike times. The strength of the laser input, and therefore the change in the membrane potential, is generally limited only by the laser power that the neural system can tolerate without tissue damage.

When controlling two neurons simultaneously, information about the membrane potential is required to ensure that the target neuron reaches action potential first. Because the value of V is not observable given the spike times, the more strict output requirements in the two-neuron case (requiring one neuron to spike before the other) mean that more precise choices for inputs are needed in the system than in a comparable one-neuron system.

Extracellular recording is vulnerable to corruption by background activity, which distorts the waveforms of the neuron of interest, making it difficult to observe. Spike sorting is generally used to separate the neuron of interest from other neural activity, but spike sorting the responses to broad stimulation presents a specific challenge. Non-targeted stimulation, such as electrical stimulation or optogenetic stimulation with a broad promotor, tends to activate large volumes of neural tissue simultaneously. When this occurs, the combined activity of the activated tissue sums to produce "hash": amorphous, unpredictable activity that, when of a large enough amplitude, obscures the action potential from the neuron of interest. I will explore some of these issues of spike sorting corruption, both from hash interference and from other sources, more fully in Chapter 3.

Hash is a significant obstacle when recording neural activity during stimulation. For optogenetic stimulation, genetic methods can be tailored to express opsins in fewer

cells. Alternatively, smaller volumes may be stimulated by using optical focusing. However, a simple way to reduce hash is to simply stimulate at lower power. Control may still function normally under these conditions, but using low power inputs biases the controlled population towards higher sensitivity neurons, those that will still be active when subject to low amplitude input.

Additionally, neurons with low spontaneous firing rates are easier to work with, as they are easier to manually find, characterize, and analyze, because spikes can be inferred to have been induced from stimulation, rather than from internal mechanisms.

For the above reasons, neurons in this study tended to have high light sensitivity and low spontaneous firing rate. When viewed as IAF models, neurons tended to have large β's, and relatively large ratios $\frac{\alpha}{\sigma}$.

2.1.4 Pilot Studies

A natural starting place to test underactuated neurocontrol is to define static stimuli S_A and S_B, apply them to activate the neurons in chosen sequences, and compare desired and observed spike responses. In pilot experiments, S_A and S_B were defined manually, by fixing stimulus durations $T_A = 1$ ms and $T_B = 10$ ms, and trying different amplitude G for both stimuli until a performant choice for both S_A and S_B was found (different T were used if needed). This approach yielded controls that tended to perform well for a short period of time, but then decayed in quality. This decay was likely due to nonstationarity in the tested neurons.

Non-stationarity is a significant factor in studies of neuronal spiking activity, and can lead to errors for many analysis techniques that assume stationarity (Grün et al., 2003). This issue is of particular importance in control systems, where nonstationarity can lead to significant deviations between the neural systems state and the controller's estimate. Possible causes of nonstationarity in neural populations are natural drift or overstimulation of the target neurons. As the parameters change over time, the SD curves of both neurons will shift, as in Figure 5. This means that the outcome of stimulation will change, usually decreasing performance. To deal with this nonstationarity, I developed an adaptive approach, that updated S_A and S_B periodically over the course of the experiment.

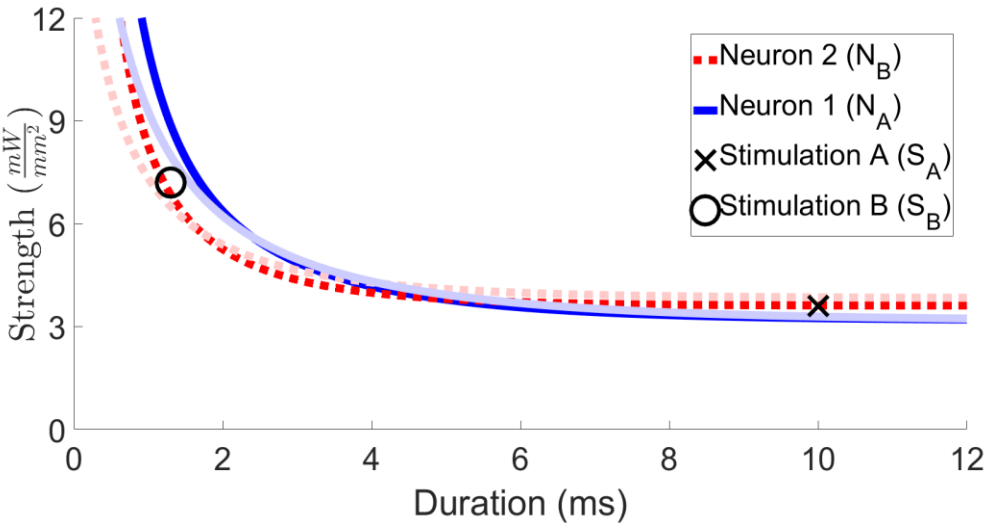

Figure 5: Nonstationarity leads to SD Curve drift

As time goes on, neurons that were once mutually controllable may drift, such as from the light shaded curves to the darker curves. In this case, the neurons are still controllable, though performance has degraded. If continued, neurons may continue to drift until they lose controllability entirely.

The first step of such an adaptive approach is to define a cost function to optimize through choice of S_A and S_B. Using this cost function, an algorithm could produce and maintain the values of S_A and S_B. Next, an optimization algorithm must be chosen. Because the stimulation results are stochastic, deterministic gradient-based search methods, such as interior-point optimization (Byrd et al., 1997, 2000), are not suitable. Therefore, I used a direct search for optimization, based on MATLAB's `patternsearch` function (adapted from the Global Optimization Toolbox for MATLAB, MathWorks Inc., Natick, MA), in which a small number of stimuli were tested directly, without calculating derivatives in cost space. This non-model based approach performed reasonably well, but, because of the design of the optimizer, many delivered stimuli were suboptimal due to the explore-exploit tradeoff. Exploration of the stimulation space left only about half of the stimuli for exploiting each point that was thought to be optimal. To decrease the effects of this tradeoff, as well as to leverage some prior knowledge about the system, I switched to a model-based approach.

In the final revision of the adaptive optimizer, the parameters of each of the two neurons were found by fitting IAF models to the neural responses. Using the θ calculated from the fits, the cost function was optimized for each stimulus with simulated results. Because both the fit and the cost optimizations were deterministic, core MATLAB gradient-descent-based optimization functions could be used. By updating S_A and S_B periodically over the course of the experiment, I was able to compensate for nonstationarity of the parameters of the controlled neurons.

2.2 Experimental Preparation

2.2.1 Hardware Setup

I performed underactuated control experiments using adult (>8 week) Thy1-ChR2-YFP (Jax 007612, Jackson Labs, Inc, Bar Harbor ME) mice. I used vaporized isoflurane (0.5% – 2.0% in O_2) as anesthesia, flow rate was kept near 600–700 $\frac{mL}{min}$. Body temperature was maintained using homoeothermic heating system (Harvard Apparatus, Holliston MA) (37°C). The experiment was controlled using a custom MATLAB script, responsible for high-level protocol flow and saving information, including stimulus parameters and trail metadata. The script interfaced with a RZ5 digital acquisition and signal processing system (Tucker Davis Technologies, Alachua FL) via the system's software server, OpenEx. The RZ5 hardware was responsible for reading, processing, and recording electrophysiology data; controlling the laser; performing all low-level task-related processing, such as trial timing randomization; and saving all task-related information. A 473 nm, 100 mW laser (Omicron PhoxX 473-100), guided through an optical fiber, provided optogenetic stimulation. A TDT 32-channel PZ5 Neuro-digitizer Amplifier and headstage was used to read neural data from a silicon probe with 8 tetrodes across 4 shanks (A4x2-tet-5mm-150-200-121-Z32, NeuroNexus, Ann Arbor MI). Figure 6 shows a simple schematic of the experimental setup, as well as the desired responses to each stimulation.

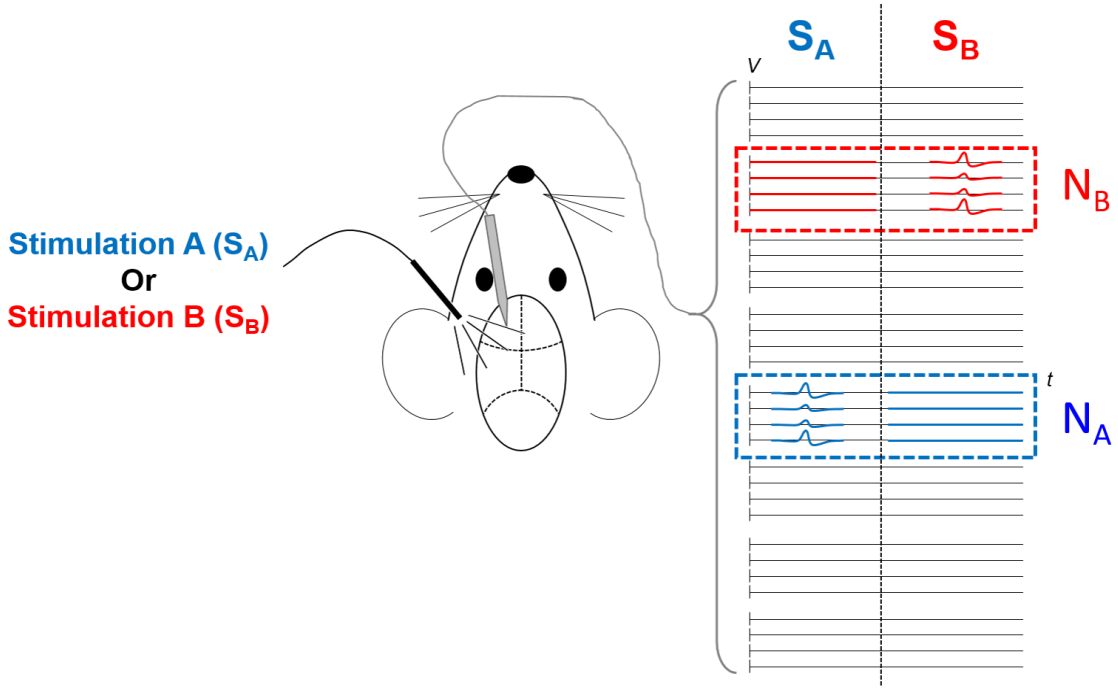

Figure 6: A schematic of the experimental setup

This schematic shows an optical fiber illuminating an exposed region of cortex. Each stimulation, S S_A or S_B, will ideally be able to bias activity towards N_A or N_B, respectively.

The laser housing includes a shutter, and the beam was guided through a 9:1 beamsplitter; the 10% beam was directed towards a photodiode used to measure the laser power online, while the remaining 90% of the power was transmitted through a 200 μm optical fiber (Thorlabs Inc, Newton, NJ) determined to the brain surface. At the beginning of each experiment, a calibration procedure determined the relationship between the control voltage $V_C \in (0,5)V$ and the output laser power P: while a series of control voltages were applied to the laser, a light meter at the terminal end of the optical fiber recorded the output powers. During the experiment, the control voltage for each desired laser power was found by inverting the 4th order polynomial $P = f(V_C)$ that best

fit the calibration pulses. Laser powers were converted to irradiances by dividing by the area of the fiber optic cable.

2.2.2 Surgical Preparation and Search for Units

The surgical procedure is as follows. After induction in a chamber using isoflurane, mice were transferred to a nose cone with bite bar on a homoeothermic heating pad (Harvard Apparatus, Holliston MA). The Matrx isoflurane vaporizer (Midmark) was set initially to 2% in O_2, and gradually reduced to about 1% over the course of the surgical preparation, guided by breathing rate and other vital signs. The fur on the top of the head was removed using scissors and hair removal cream (Nair). The skull was stabilized in ear bars, and the scalp resected at the midline. A craniotomy (~1 5mm diameter) was formed over barrel cortex (0.5 mm posterior and 3.5 mm lateral of bregma). The dura was removed with Vetbond cyanoacrylate glue (3M, Saint Paul MN). A saline well made from a cut section of a 0.5 ml plastic centrifuge tube was glued to the skull around the craniotomy. A ground wire was placed into a burr hole, contralateral to the craniotomy.

The optical fiber and silicon probe were advanced on separate stereotactic arms into the well. The optical fiber was placed at a 45° angle, such that the light was directed posteriorly. The probe was placed at a 20° angle, such that the probe inserted approximately perpendicular to the brain's surface. Special care was taken to ensure the optical fiber was parallel with the probe surface, and on the side opposite to the electrode contacts, to minimize light artifacts. The probe was advanced into the brain about 500

μm, and allowed to rest for 5 minutes. The position of the optical fiber was readjusted as needed to ensure that light artifacts had very low amplitude related to the noise floor.

To find candidate controllable pairs, two preset stimuli were chosen, one at 1 ms duration and the other at 10 ms duration. The strengths were set such that they evoked a local field potential deflection on most contacts, and multi-unit activity on some contacts, with the amount of activity evoked by both stimuli approximately equal. These stimuli were alternately presented during manual search for putative single units that react to one or both stimuli.

If no responsive single units were found, the probe was advanced ~50 μm – 100 μm, followed by a new search. If a responsive single unit was found, the two test stimuli were adjusted such that the candidate unit spiked in response to about 70%–90% of both stimuli. The remaining tetrodes were then searched using these adjusted stimuli for any candidate units that had a high firing probability in response to one stimulus, and a low firing probability in response to the other. The stimulus strengths were then adjusted until it was found that either there exists a set of stimuli such that each neuron could be biased to be more active than the other, in which case full testing began, or no such set of stimuli was found, and remaining contacts were searched for controllable units. If no candidate pair of units were found, the remaining contacts were searched for identifiable units, or if no more were found, the probe was advanced ~50 μm – 100 μm.

Once a candidate pair was found, the testing of the pair proceeded as follows. Candidate units were isolated online on the RZ5 using SpikePac sorting software (Tucker Davis Technologies, Alachua FL). A characterization step was performed, to first fit

each unit to an IAF neuron model, and then calculate a set of optimal stimuli S_A and S_B. The process is described in more detail in 2.4.1 Initial Characterization. Once the optimal stimuli were found, the pair was tested to determine the effectiveness and stability of control. The process is described in more detail in 2.4.2 Adaptive Control.

2.3 Computational Methods

2.3.1 Table of IAF spike probabilities

To characterize in vivo neurons, responses were compared to simulated integrate and fire (IAF) neurons (Dayan & Abbott, 2001). Pulsatile stimuli were described by their strengths G and durations T, and their responses were coded as either 1 (at least one spike) or 0 (no spikes). The probability of firing was determined by numerical solution of a Fokker-Planck (FP) equation, given the neural parameterization θ and stimulus $S \equiv [G, T]$ (Iolov et al., 2017).

Suppose $P(V, t)$ is the probability density function over membrane potential V and time t for an IAF neuron. The evolution of $P(V, t)$ as described by the FP equation is

$$P(V,t) = -\frac{\partial}{\partial V}\big(-\alpha V + \beta S(t)\big)P(V,t) + \frac{\sigma^2}{2}\frac{\partial^2}{\partial V^2}P(V,t) \qquad (4)$$

This equation was solved using Crank-Nicolson numerical integration with an absorbing boundary at the threshold V_T, and a reflecting boundary at a lower boundary $V_L < 0$ (chosen so that a negligible portion of the probability mass would touch the lower

boundary). The initial condition was found by finding the membrane potential distribution of an unstimulated simulation to come to near steady-state (negligible flow out of the absorbing boundary), and multiplying this distribution so that the total mass in the domain is 1. The firing probability $P_{Spike}(T)$ was calculated by finding the fraction of the original probability mass that left the domain through the absorbing boundary V_T between times $t = 0$ and $t = T$. For simplification, any simulation for which less than P_{cutoff} mass remained within the boundary was coded as 100% spike probability. All parameter values used in the FP calculation are shown in Table 1.

Parameter	Description	Value(s)
V_T	Membrane potential threshold for a spike	.2
V_L	Lower boundary for membrane potential domain	-1.5
n_T	Number of divisions in the time domain (from 0 to t_{Upper})	5001
n_V	Number of divisions in the membrane potential domain (from V_L to V_T)	301
t_{Upper}	Upper limit for the time domain	15
P_{cutoff}	Threshold for assuming zero firing probability	10^{-4}
Strength Bounds	Boundaries of tested stimulation strengths	[0 5]
Duration Bounds	Boundaries of tested stimulation durations	[0 15]
α Bounds	Boundaries of tested α values	[0 .5]
σ Bounds	Boundaries of tested σ values	[.001 .3]
n_{divs}	Number of values tested within each set of bounds	45

Table 1: Parameters used during Fokker-Planck integration and associated database calculation

Note that the definition of the firing probability P_{Spike} in this case is the probability that the neuron will fire at least once. This is represented in the model by removing any probability mass from the domain that has crossed the V_T threshold. The probability of any spike occurring is used (as opposed to the probability of exactly one spike) because the stimuli applied to the neurons during the experiment have a natural

constraint, that the non-target neuron should not spike. Because of this constraint, stimuli will have relatively low power and will be unlikely to induce multiple independent spikes in the target neuron. Thus the probability of any spike occurring is approximately equal to the probability of a single spike occurring, but is easier to calculate.

Numerical integration of the FP equation is too computationally expensive for online use. Because of this, numerical solution was performed offline over a mesh of neural parameters α and σ, and stimulation parameters G and T (for IAF neurons, β acts only as a scale for stimulation strength G). This produced a large table of firing probabilities across a variety of parameter values. The boundaries of each parameter are given in Table 1, and $n_{divs} = 45$ equally spaced values of each parameter were used. To predict the firing probability of an experimentally recorded neuron with fitted parameters $\theta = [\alpha, \beta, \sigma]$ and stimulus $S = [G, T]$, the table was linearly interpolated.

Due to the near-linearity of the IAF model, it was assumed that there would be very little cumulative increase in firing probability after stimulus offset. In other words, if 5 ms stimulus is applied to the FP model, I assumed that very little probability mass would leave through the threshold boundary between $t = 5$ ms and a reasonable upper boundary for the post-stimulation window. I simplified the FP calculation by performing integration for each set of $[G, \alpha, \sigma]$, up to the maximum considered stimulation time $T_{max} \equiv 15$ ms. To find the firing probability for any stimulus with duration $T < T_{max}$, the firing probability was calculated as above by finding the instantaneous $P(V, t)$ that has left the domain at the given time point t. This leads to a slight underestimation of firing probability.

2.3.2 Error Function (θ Optimization)

To estimate θ of an in vivo neuron, an error function was used to compare the neuron's stimulation results with predicted firing probabilities found via table interpolation.

A summed squared error was used as the error function. Specifically, suppose $\{x_1, x_2, \dots, x_N\}$ are the measured responses of the in vivo neuron to stimuli $\{[G,T]_1, [G,T]_2, \dots, [G,T]_N\}$, where $x_i \in \{0,1\}$ and 0 means "no-spike" and 1 means "spike", and $\{y_1, y_2, \dots, y_N\}$ are the firing probabilities of an IAF neuron with parameters θ, given the same stimuli, where $y_i \in [0,1]$. The error function f_{err} is given by

$$f_{err} = \sum_{i=1}^{N} (x_i - y_i)^2 \tag{5}$$

2.3.3 Cost Function (GT Optimization)

The choice of stimulation parameters during the characterization and adaptation phases was guided by the optimization of a cost function. Suppose P_T is the predicted firing probability of the target neuron, P_{NT} is the predicted firing probability of the non-target neuron, λ is a penalty factor for laser power, and G is the power in $\frac{mW}{mm^2}$ of the chosen stimulus. The cost f_{cost} of a stimulation parameter choice was given by

$$\mathrm{f_{cost}} = -P_T(1 - P_{NT}) + \lambda G^2 \tag{6}$$

Note that P_T and P_{NT} are complicated functions of (G,T) computed through table

lookup, as described above. Minimizing this function over $S = (G, T)$ yielded a stimulus S_T for each neuron to maximize that neuron's firing probability P_T, balanced against minimizing the firing probability of the non-target neuron P_{NT}. A demonstration of the shape of the cost function is in Figure 7, which shows how the cost changes in SD space for each stimulus S_A and S_B given a pair of IAF neurons that satisfy the mutual controllability condition set forth by Ching & Ritt (2013).

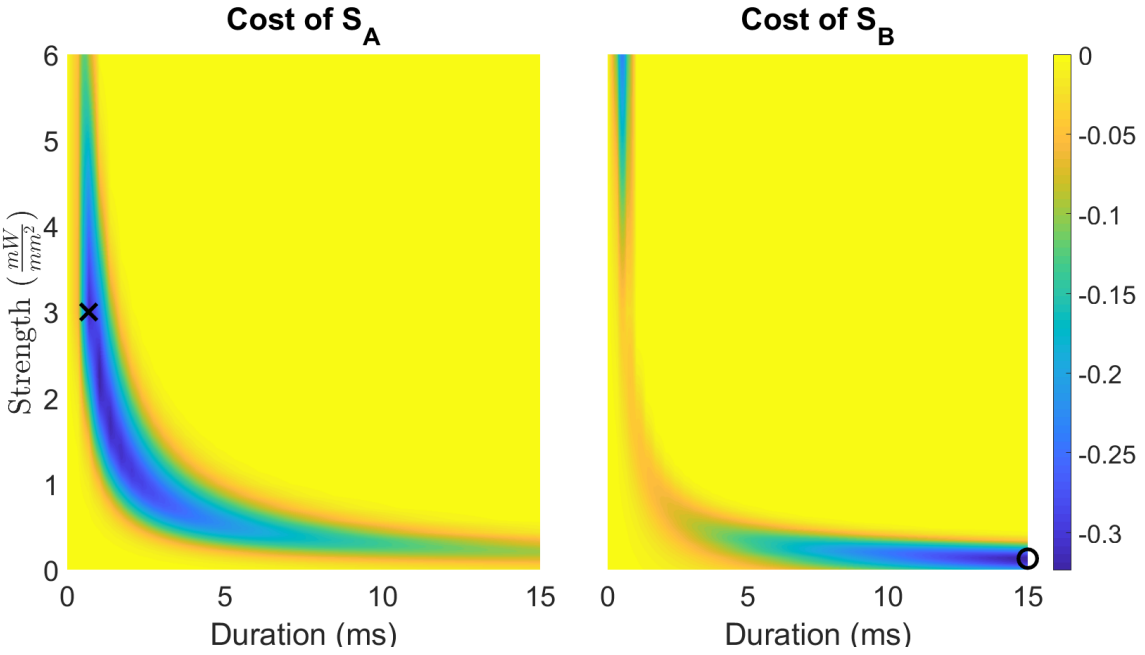

Figure 7: Shape of the cost function in SD space

Evaluation of the cost function at various locations in SD space for each stimulus, when using a pair of mutually controllable IAF neurons ($\boldsymbol{\theta_1} = [.\mathbf{3}\ .\mathbf{125}\ .\mathbf{05}]$, $\boldsymbol{\theta_2} = [.\mathbf{05}\ .\mathbf{06}\ .\mathbf{05}]$). The "X" and "O" show the optimal S_A and S_B for this pair neurons, respectively.

The penalty term is included for two reasons: The first reason is that the optimal solution S_T for most neuron pairs lies on the boundaries of the strength-duration space, due to the fact that the unpenalized cost function is concave when P_T and P_{NT} are neither

0 nor 1, so a soft boundary in G was implemented. The second reason is that the unpenalized cost is flat across most of SD space (most stimuli either make both neurons or neither neuron spike, and the firing probability is monotonic with both G and T), which means that the optimization function (MATLAB's `fmincon`) cannot calculate a gradient in these areas, and therefore cannot continue the optimization. The addition of a penalty introduces a gradient towards $G = 0$, which allows the optimization function to move from very large G's back into a space where at least one neuron's firing probability is not 1. Due to the fact that multiple initial conditions were used as a global optimization method, it was unlikely that the optimization function would spend much time in the space of low (G, T)'s, and therefore the space in which neither neuron spikes was not a significant issue. Throughout, $\lambda = 10^{-5}$.

2.4 System Identification and Control

2.4.1 Initial Characterization

After each unit was manually identified, they were stimulated until a sufficient quantity of waveforms was recorded so that online tetrode sorting could be performed to isolate the units of interest. Then, a characterization step was performed on both units.

This characterization procedure generated a large buffer of data for each unit by stimulating at various positions in SD space. This buffer facilitated calculation of an initial estimate for each unit's θ, and therefore an initial optimal stimulus S_T for each neuron.

To produce stimuli that are maximally informative for estimating θ, the characterization procedure consisted of cycling through a series of predetermined stimulus durations, and, for each duration, attempting to find the strength G_{P50}that evokes a 50% firing probability. Durations were selected from the set (1, 2, 5, 10, 15) ms. The procedure for the characterization is as follows:

1. The strength upper boundaries were manually chosen for each duration so as not to overstimulate the units during characterization. This is done by manually choosing a strength (at which both units were found to have a near 100% probability of firing) at the lowest and highest durations, 1 ms and 15 ms. The strength upper boundary for the remaining intermediary durations was linearly interpolated between these two extremes. The strength lower boundaries for all durations was 0 $\frac{mW}{mm^2}$.
2. The G_{P50} strength was then found for each duration sequentially.
    a. The duration $\tilde{T}$ for this round was chosen from the list, starting for the lowest duration.
    b. The lower and upper strength boundaries for this duration (G_{min} and G_{max}, respectively) were tested. If a spike occurred on the lower boundary, or no spike occurred for the upper boundary, then the characterization was paused so that the boundaries could be manually readjusted. Alternatively, if the user believed that the result from any of the boundaries was uncharacteristic of normal behavior, the procedure could be continued under the assumption that the lower boundary did not produce any spike and the upper boundary did (this assumption is only used for the purposes of finding G_{P50}, and not later

for calculating θ).

c. A set of "allowable strengths" were calculated for this duration: a mesh of strengths were generated between G_{min} and G_{max} with a step of $\Delta G = .01$.

d. The procedure alternated between each unit whose G_{P50} had not yet been found, starting with unit 1. A linear regression of spike response (0 or 1) on strength was calculated using all previous stimuli at this duration.

e. Using the resulting linear regression, a next stimulation strength was found as the strength whose firing probability was 0.5 according to the linear regression. This value was rounded to the nearest allowable strength, becoming the next estimate, $\tilde{G}_{P50}$.

f. Two conditions were checked. The first condition was whether both a minimum number of stimuli ($n = 15$) had been given, and this iteration's $\tilde{G}_{P50}$ was less than $x_{thresh}G_{max}$ away from the previous iteration's value (where $x_{thresh} = .05$). The second condition was whether a maximum number of stimuli ($n = 30$) had been given.

    i. If neither of the conditions were met, then the stimulus $[\tilde{G}_{P50}, \tilde{T}]$ was applied, and the responses of both neurons recorded. The process then returned back to d, switching which unit was being characterized if needed.

    ii. If either of the conditions were met, then G_{P50} for this duration and unit was set to $\tilde{G}_{P50}$. If the other unit's G_{P50} for this duration was not yet found, the stimulus $[\tilde{G}_{P50}, \tilde{T}]$ was applied, and the results of both

units recorded. The process then returned back to d (to find a new $\tilde{G}_{P50}$ for the other unit). Otherwise, the next duration was selected, and the process returned to c, unless all durations were completed.

An initial estimate of θ for both units was calculated from f_{err} using this stimulation data, and then an initial S_T for each unit was found by minimizing f_{cost}.

2.4.2 Adaptive Control

The structure of stimuli sent during the adaptive control tests were organized into a hierarchy:

1. Stimulus – A single rectangular pulse of the laser, labeled by a given $[G, T]$, classified as either an S_A or an S_B stimulation (intended to selectively increase the firing probability of N_A or N_B, respectively).
2. Sequence – A series of 5 Stimuli administered in a burst, with an inter-stimulation interval of 100 ms. The stimuli in a Sequence were composed of either $3S_A + 2S_B$ stimuli, or $2S_A + 3S_B$ stimuli, for a total of 20 possible unique Sequences.
3. Run – A collection of 20 Sequences. A Run consisted of one of each possible unique Sequence that can be constructed from $3S_A + 2S_B$ and $2S_A + 3S_B$ Stimuli. Each unit pair was tested using a session of 20 Runs. Across each run, Latin Squares randomization was used to ensure that each unique Sequence was found at the beginning, middle, and end of a run an equal number of times, to randomize any Run-ordering effects.

4. Block – Used exclusively for the update step of the adaptation, a block was composed of 10 Sequences (one half of a Run). Because of this, there was no guaranteed distribution of S_A and S_B Stimuli.

The two optimal stimuli found during the initial characterization step (one for each unit) were used during the first block of the adaptive control period. The adaptation algorithm, described below, found updated θ's for both unit using only information from the first block (not from the initial characterization), and two updated optimal stimuli. Stimulation resumed using these new stimuli, until they were reevaluated after completing the following block. The adaptation protocol maintained a buffer of stimuli and responses from the previous (at most) 4 blocks ($N_{Buf} = 200$ stimuli). The buffer started filling at the onset of the adaptation control procedure (i.e. no initial characterization stimuli were used). Adaptive control iteratively alternated two steps:

1. Using the procedure in Section 2.3.2 Error Function (θ Optimization), the parameters for each unit were estimated.

   Using the procedure in Section

2. 2.3.3 Cost Function (GT Optimization), the optimal stimulus was found for each unit, given their current estimated parameters.

This procedure continued throughout the adaptive control procedure. A summary of the procedure, include all steps taken before adaptive control began, is shown in Figure 8.

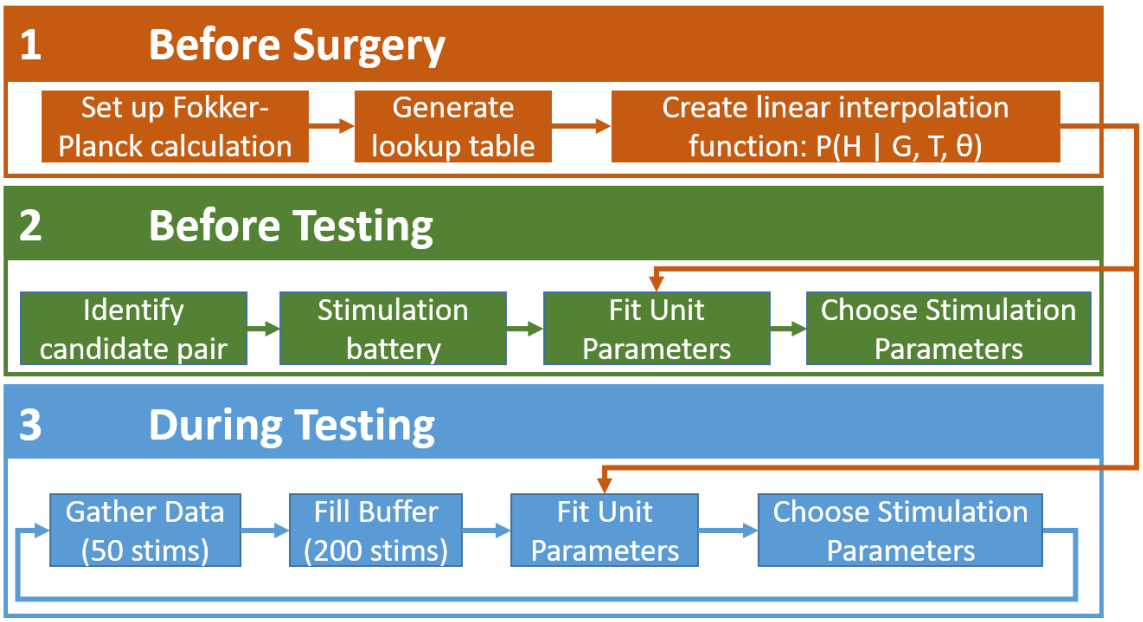

Figure 8: An outline of the characterization and optimization protocol

Once testing concluded, the depth of the silicon probe was recorded, and the probe was advanced to begin searching for the next candidate pair.

2.5 Data Analysis Methods

2.5.1 The Response Fraction

Efficacy of control was evaluated according to a number of related metrics. The first metric was the response fraction (RF), or the fraction of stimulus presentations that evoked at least one spike. The RF serves as an estimate of the firing probability of each unit. The true positive (TP) response fraction was defined as the response fraction limited to stimulus presentations targeting that unit:

$$TP = mean(x_T | S_T) \tag{7}$$

where x_T is the target unit response coded at 0 for no spikes and 1 otherwise. In contrast, the false alarm (FA) response fraction is the response fraction of a unit when subject to stimulus presentations that do not target it:

$$FA = mean(x_{NT}|S_T) \tag{8}$$

TP is an estimate of the probability of target unit firing, $P(N_T, |S_T) \equiv P_T$, while FA is an estimate of the probability of the non-target unit firing, $P(N_{NT}|S_T) \equiv P_{NT}$. The response fractions can also be thought of as the elements of a confusion matrix, as in Table 2.

		Stimulus Type	
		S_A	S_B
Spike Response	N_A	TP_A	FA_B
	N_B	FA_A	TP_B

Table 2: Definitions of the Response Fractions

By definition, stimuli that produce large TP and small FA are more effective. However, each pair of stimuli (S_A and S_B) will produce four response fractions (TP_A, TP_B, FA_A, FA_B), that must be further compared to produce a single metric by which overall control efficiency can be evaluated.

The cost function defined in section

2.3.3 Cost Function (GT Optimization) is a natural evaluation metric. However, it does not capture all features one might desire, such as a direct interpretation in terms of biasing towards the target unit. For example, suppose a certain stimulus produces [TP,

FA] $= [1, 0.75]$ and another stimulus produces [TP, FA] $= [0.5,0.5]$. Both of these stimuli have a cost $f_{cost}(1, 0.75) = f_{cost}(0.5, 0.5) = -0.25$ (ignoring the penalty term), but the first stimulus induces some biasing towards the target unit while the second does not. An alternative evaluation metric is a simple difference of the response fractions for each stimulus. For each stimulus (S_A and S_B), this response fraction difference (RFD) is defined as

$$RFD = TP_T - FA_{NT} \tag{9}$$

where TP_T is the true positive response fraction of the stimulus's target neuron, and FA_{NT} is the false alarm response fraction of the stimulus's non-target neuron. It has the beneficial property that $RFD = 0$ is a natural boundary between biasing toward or away from the target unit.

The RFD (or rather, $-RFD$) was not used as the cost function online because, being a simple difference, it does not incorporate a penalty for extreme spike probabilities. The cost f_{cost} punishes any stimulus which has FA close to 1 or TP close to 0, and has a gradient that points towards the $TP = 1 - FA$ line in those regions. However, the $-RFD$ function has a gradient $\nabla RFD = [1, -1]$ at all points. This has the undesirable property of encouraging movement towards the boundaries of the TP-FA space.

2.5.2 The Control Quality Metric

A control quality metric (CQ) was defined as the minimum of the mean RFD's of the two stimuli over a session

$$CQ = \min(\overline{RFD_A}, \overline{RFD_B}) \tag{10}$$

$CQ \in [-1,1]$, and $CQ = 1$ represents perfect control. From this definition naturally follows a minimum controllability criterion for each session. I defined any unit pair such that $CQ > 0$ as having met the minimum controllability criterion. Additionally, I define any pair in which at least one stimulus has $RFD > 0$ as having met the one-way controllability criterion, meaning that activity can be biased towards at least one neuron. To determine confidence intervals for the metric, I considered the 95% high density region (HDR) (Hyndman, 1996) of the distribution of CQ. Assuming that RFD_A and RFD_B are Gaussian distributed random variables, we can derive the distribution of CQ and its HDR.

Suppose two Gaussian random variables X_1 and X_2 are parameterized by means (μ_1, μ_2) and variances (σ_1^2, σ_2^2), with correlation coefficient ρ. The PDF of M is

$$PDF_M(x) = \frac{1}{\sigma_1}\phi\left(\frac{x-\mu_1}{\sigma_1}\right) \times \Phi\left(\left(\frac{\rho(x-\mu_1)}{\sigma_1\sqrt{1-\rho^2}}\right) - \left(\frac{x-\mu_2}{\sigma_2\sqrt{1-\rho^2}}\right)\right)$$

$$+\frac{1}{\sigma_2}\phi\left(\frac{x-\mu_2}{\sigma_2}\right) \times \Phi\left(\left(\frac{\rho(x-\mu_2)}{\sigma_2\sqrt{1-\rho^2}}\right) - \left(\frac{x-\mu_1}{\sigma_1\sqrt{1-\rho^2}}\right)\right) \tag{11}$$

where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of the standard normal distribution (Nadarajah & Kotz, 2008).

I assumed that the correlation coefficient between the two RFD's was $\rho = 0$, corresponding to the independence of the two stimuli. I found the HDR of PDF_{CQ} using a numerical watershed method: the boundaries of the HDR (x_l, x_u) were found such that

they intersect PDF_{CQ} at the same y-value, and $\int_{x_l}^{x_u} PDF_{CQ} = .95$. The convexity of PDF_{CQ} ensured that the HDR is continuous. I considered any session whose CQ HDR was entirely greater than zero to have met the minimum controllability condition.

2.5.3 Confirmation of Driving Responses Towards Targets

I used a shuffling approach to test the selectivity of the stimulus pair, defined as a tendency of observed response fractions towards their desired values that cannot be explained by chance.

I conducted $N = 20{,}000$ random shuffles of the stimulus labels (S_A, S_B) for each session, and calculated CQ for each shuffle. This produced a distribution of CQ's for each session. The selectiveness metric was calculated as the z-score of the session's true CQ relative to the shuffled CQ distribution. Z-scores that are far to the right (higher CQ) from the shuffled distribution indicate that the control designed performed as well as it did because the stimulation did in fact push target firing probabilities up and non-target probabilities down.

2.6 Results

2.6.1 Summary of Results

24 mice were tested, and candidate unit pairs were found in 15 of the mice. In these 15 mice, 29 total candidate pairs were found. 8 of these pairs met the minimum controllability criterion. Every pair that was tested met the one-way controllability

threshold. The CQ distribution across pairs is shown in Figure 9. All results shown are for online sorted units.

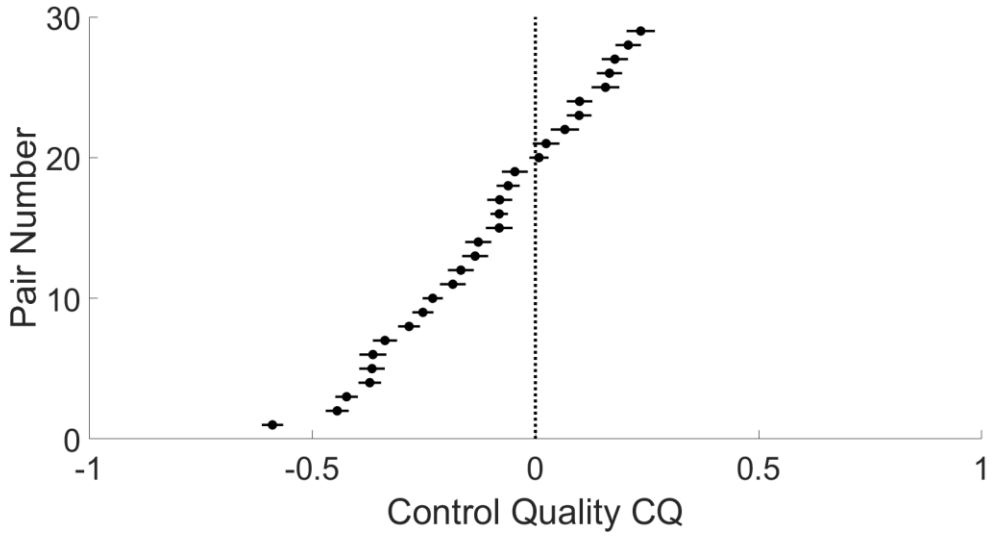

Figure 9: CQ for each tested pair (n=29)

8 pairs were found whose CQ is larger than 0. This means they met the minimum controllability criterion, such that both stimuli induced more activity in their target unit than their non-target unit.

Out of the 29 candidate pairs, in 27 cases units responded highly selectively to the two stimuli. In these cases, the stimuli were able to drive the units towards their targets, if not all the way to meet minimum controllability. As seen in Figure 10, most pairs responded to stimuli with selectivity well above chance.

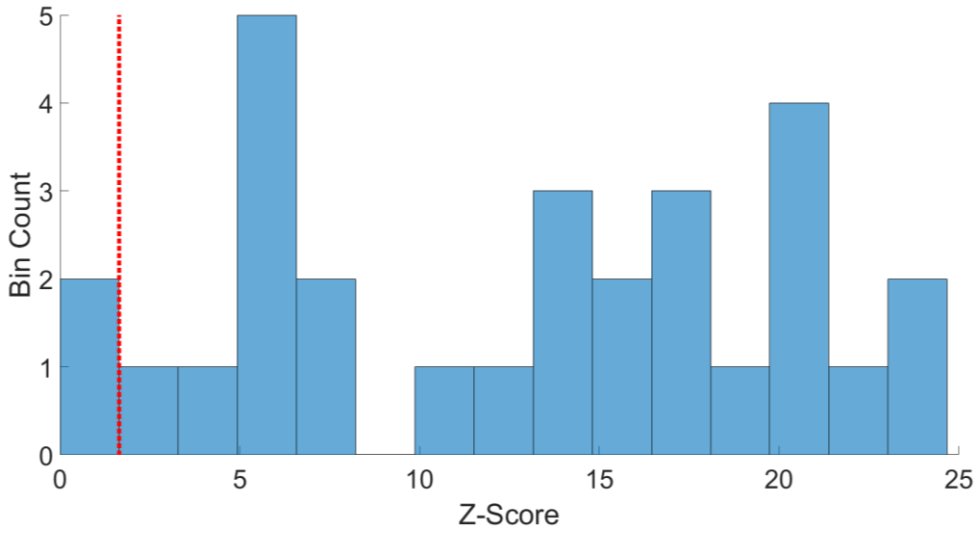

Figure 10: Out of the n=29 pairs tested, 27 responded with high selectivity to the stimulus.

A histogram of the selectivity of each tested pair, in terms of z-score. Bin widths are $w = 1.645$.

2.6.2 Example of a Controllable Pair

An example of a controllable pair can be seen in Figure 11. The RF's, CQ, cost, stimulation parameters $[G, T]$, and both sets of neural parameters $\theta = [\alpha, \beta, \sigma]$ are shown. The mean CQ for the pair shown in Figures 11/12 is 0.1780, and its selectivity has a z-score of 23.10.

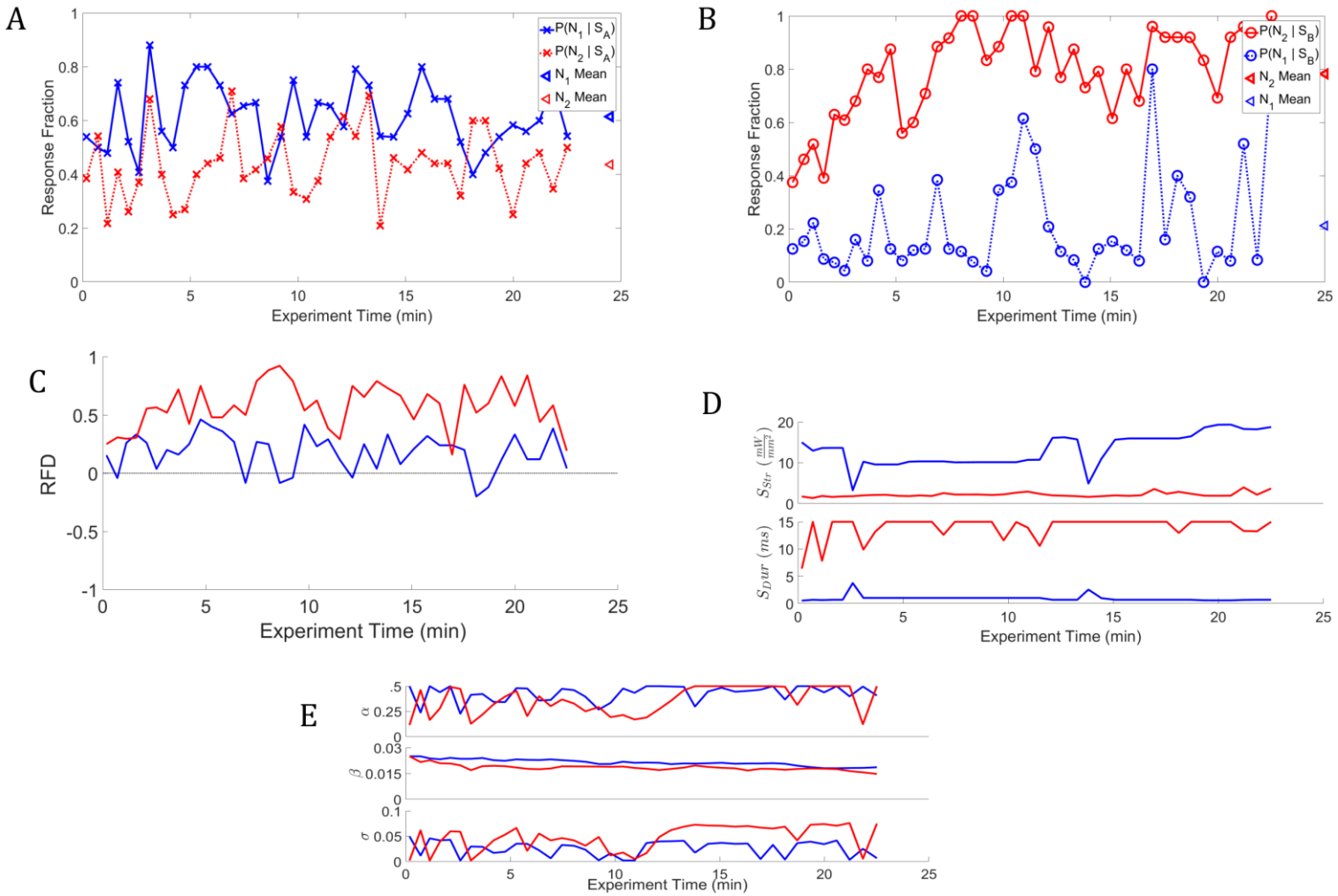

Figure 11: The results of an example controllable pair

The RF's (panels A/B), CQ (panel C), stimulus parameters (panel D), and θ (panel E), plotted over the full course of the experiment for both N_A and N_B. Each point represents a single block of 50 stimuli. S_A and S_B were updated after each block.

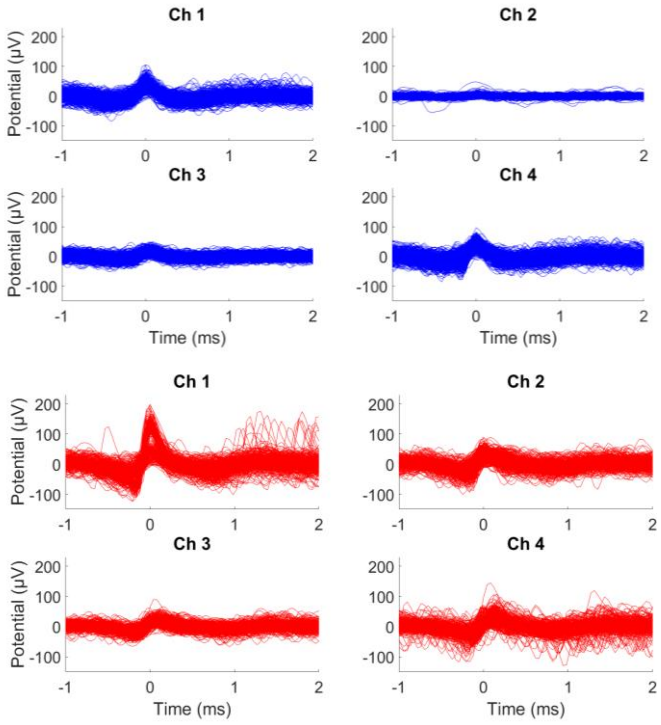

Figure 12: Spike waveforms of an example controllable pair

Spike waveforms are shown for both neurons in a controllable pair (same as Figure 11). All waveforms that were categorized as spikes are shown for all four electrodes of each tetrode associated with the neurons.

The RF's for S_B show that the stimulus was able to cause N_B (red circles) to fire at a higher probability than N_A (blue circles) for each block. S_A was able to bias the N_A (blue crosses) firing probability to be higher than that of N_B (red crosses) for most blocks, though control failed in a few places. Because of the controller's ability to recover control quickly, it is likely that drops in performance were due either to random fluctuations in unit responses, or small changes in fitted parameters leading to momentary poor choices of stimulation parameters.

The RF's for this pair, and therefore its CQ, are relatively stable over time, showing few long-term trends after the first 3 minutes or so. However, despite the long-

term stability, there is a large variance. This variance in RF and CQ likely originates from the variance in both units' θ fits. While β appears to have very little jitter, both α and σ show significant variance, though there do not appear to be any large scale trends over the course of the experiment for either.

The controller appears to tolerate this variance, still managing to produce response fractions that exceed the minimum controllability condition in many blocks, but increased stability will likely be important for using this method in applications. The estimate of θ might be stabilized by increasing N_{Buf}, widening the sliding window to smooth out the results. The size of the buffer N_{Buf} was initially chosen to allow the controller to update its values at a pace near the rate of change of unit parameters, which, based on manual observation, appeared to change on about a 2-minute timescale. A different type of sliding window might also be used, such as a Hann window instead of the current boxcar method, to suppress discontinuities caused by dropping blocks from the buffer, and smooth transitions between blocks.

It was noted in this pair, as in most of the pairs tested, that the estimate for β tends to decrease gradually over time. While this does not necessarily mean that the true β for both units does in fact decrease, it does mean that the controller finds that the optimal fit is that of a continually less sensitive unit as time goes on. There are a number of reasons that the neuron could become less sensitive over time, such as neural plasticity or channelrhodopsin dynamics. It is difficult to tell if this change was due to over-stimulation or intrinsic variation across time, because in the presented work, stimulus count and elapsed time are confounded.

2.6.3 Common Failure Modes During the Experiment

As can be inferred by the small number of candidate pairs found per mouse (~1.2 pairs), there were a number of failure modes that could prevent a pair of neurons from being considered control candidates. Beyond standard surgical and anesthetic issues, the first failure mode concerns the expression level of the rhodopsin. In earlier pilot experiments (not shown), a different promotor, Emx1 (Jax 005628, Jackson Labs, Inc, Bar Harbor ME) (Madisen et al., 2012), was used that expressed ChR2 more densely than the Thy1 promotor. While this increases the number of possible candidate neurons, the large number of light sensitive cells meant that each stimulus caused a very large population response. This population response made separating the neuron of interest from the background activity substantially more difficult. It also amplified the possibility of introducing network effects that could affect the neurons of interest in unknown ways.

Another common outcome was finding one isolated neuron at a given probe position, but not being able to find a second unit with which it could be mutually controlled. In pilot experiments, a pair of carbon fiber glass electrodes (Kation Scientific Carbostar) on independent stereotactic arms were used to search for units. The movement of the electrodes relative to each other often caused unit to change their behavior when subject to stimulation, often times irreversibly. This motivated use of the single 8 tetrode silicon probe, which did not require movement during the process of matching units for control. While the lack of movement stabilized the activity of the initial unit while a matching unit was sought, the limited number of tetrodes meant that the search space was limited, meaning that many isolatable units were rejected when no

match could be found on the other tetrodes.

Earlier theoretical work showed that one might expect about a 25% probability that any two units will be mutual controllable (Ching & Ritt, 2013). Despite this estimate, many units were rejected because they did not satisfy the controllability condition with the initial neuron. This discrepancy may be due to the bias in neurons chosen for control in this study, towards those with high sensitivity and low spontaneous firing rate (2.1.4 Pilot Studies). Additionally, many units were rejected because their behavior was erratic, either due to a large amount of intrinsic or pre-synaptic noise, or a large degree of nonstationarity. These units appeared to change their behavior too rapidly to be consistently controlled.

Another issue that prevented neurons from being controlled was online spike sorting. Due to the nature of the experiment, which requires recording neural responses directly after stimulation, waveform sorting was significantly more difficult than sorting spontaneous activity. Waveforms were embedded in a "hash" response that occurs when a large volume of tissue is activated simultaneously. Online spike sorting was able to separate the unit of interest in many cases, but there were others in which the corruption was too high. In these cases, the unit usually was abandoned. The effects of various types of spike sorting errors will be explored further in Chapter 3.

In addition to failure modes that prevented units from being considered for control candidacy, other failure modes occurred after characterization, whereby a candidate controllable pair lost controllability over the course of the session. There were four general modes of failure that occurred after characterization, and many pairs were

affected by multiple. The first and most common was that the unit pair could be biased towards some target activity with one stimulus, but could not be biased with the other. This occurred in about 14 pairs. In these cases, one stimulus was generally able to produce good performance (a positive RFD), while the other performed poorly (a negative or zero RFD). In general, it was common to find a pair for which one stimulus would bias responses but the other would not, which is expected as this will occur any time two neurons' SD curves do not perfectly overlap. The trivial case was that of a particularly light sensitive unit paired with another with low sensitivity. One unit would always respond given a stimulus, while the other almost always stayed quiescent. Therefore, for any two stimuli, the stimulus targeting the highly sensitive neuron would have a positive RFD while the other would have a negative RFD.

The second mode of failure occurred when one or both stimuli would cause both units to fire with about equal probability (RFD = 0). This case occurred in about 5 pairs, and was common in the early pilot experiments.

A third mode of failure was general instability in the controlled system, which occurred in about 5 pairs. In these cases, the stimulus responses of each unit would tend to vary rapidly in such a way that could not be predicted and therefore the adaptive optimizer could not compensate. This was generally uncommon after a pair was characterized, but this instability was a major factor in disqualifying units from being control candidates.

The final fourth mode of failure for units was for the pair to start as controllable, and then fall out of the controllable region in α/β space over the course of the session,

thereby leading to poor performance. About 4 of the pairs that failed minimum controllability experienced this mode of failure. Some pairs appeared to be controllable during the characterization stage, but started the session with poor performance, then regained control sometime later in the session. Often in these cases, performance would be good for some time during the session, but would then fluctuate or remain poor for the rest of the session. The CQ measure is designed to detect sustained controllability, so it does not identify units controllable only for short periods of time, as CQ is calculated as a mean over the entire session. Temporary controllability is not as valuable as long term controllability, but it is worth noting. Transient control may still be useful in a clinical setting: fluctuating into and out of controllability may not be useful at an individual neuron level, but may have implications for controllability of a large population that can be tracked simultaneously. It may be the case that the neurons to be controlled are selected from some large, clinically relevant population, and while pairwise controllability may change on a short time scale, some fraction of the population might remain controllable at any given time.

2.7 Discussion

2.7.1 Comparison to Other Neurocontrol Studies

The work in this dissertation expands control to an underactuated system of two neurons, as well as adding an adaptive recalibration component. The addition of an adaptive component allows the controller to leverage more information than if only a single epoch of data was used, which both increases the accuracy of the model and allows

it to evolve with the system.

Previous work has shown successful control of single units using extracellular recording electrodes, both in simulation and in vivo. For example, closed loop control has also been shown to drive single unit spike rates towards both constant and sinusoidal targets using PI control (Bolus et al., 2018; Newman et al., 2015). Earlier methods also successfully drove single neurons towards arbitrary spike trains in continuous time (Ahmadian et al., 2011). The work presented by (Iolov et al., 2014) demonstrates a robust in silico method for producing arbitrary spike trains in single neurons. Like the work above, it uses a method to characterize an IAF neuron using only spike-times, then produces a continuous-time stimulus, though the calculation cannot be done online. Other systems have been designed explicitly for use in clinical applications, using optimal system identification to inform a closed-loop controller (Yang et al., 2018). Each of these techniques works in continuous time, producing spike trains or spike rates that are not limited by discretization. This is unlike the work shown in this dissertation, which considered only if more than one spike was produced after each stimulus. However, no previous work has explicitly addressed the underactuation problem in vivo.

Other work presents solutions to similar problems, leading to various potential paths forward for medical neural interfaces. For example, a deep learning, model-free approach has been shown to induce physiologically meaningful states in simulated networks of neurons with underactuated stimulation (Mitchell & Petzold, 2018). While such model-free approaches tend to require more training data and tuning than model-based approaches, they also have the potential to exert more exact control over the neural

population of interest.

A final approach is to avoid the problem of underactuated control by changing the formulation of the problem. Instead of attempting to control a greater number of neurons than there are electrodes in a volume, a single unit may be used to influence overall activity of a globally connected neural population (Nabi & Moehlis, 2011). This reformulation has some problems that must be addressed before implementation, such as the requirement of targeting individual members of populations, but it also relaxes some other stimulation requirements. Alternatively, stimuli can be designed to affect many neurons simultaneously, which can be used for population entrainment (D. Wilson et al., 2015). While the goal is still precise control over a large number of neurons, the problem is simpler because the target state is common across all cells.

3 CONTROL WITH CORRUPTION

3.1 Introduction

3.1.1 The Importance of Observability During Neurocontrol

A spike sorter's effectiveness is typically scored by the degree to which it is able to isolate a neuron from its background activity, much like other classifiers, by calculating its precision and recall (or its false-positives and false-negatives) (Hill et al., 2011). However, considering the fact that the role of spike sorting when used in a control system is to identify meaningful information for the controller, a different metric to evaluate the sorter emerges. The spike sorter's performance can be measured not just by how well it classifies spikes, but by its downstream effects: how well the controller performs using the information that the spike sorter provides. It is possible that poor spike clustering will still allow good control, or that great clustering will lead to poor control. As with all closed-loop control systems, performance is dependent on the accuracy of observations, but the quality of observation does not necessarily directly correlate to quality of control, nor does it necessarily act as a limitation. Many factors contribute to a system's ability to exploit good observations or tolerate poor ones, such as the observable variables' sensitivity to changes in the state (observability), the system's sensitivity to movement in the input space (controllability), and the inherent noise present in both the system and observations.

By choosing to use control accuracy as the metric by which to evaluate clustering, I assert that we do not require high accuracy from the classifier, but simply good synergy between it and the controller. Adjusting the classifier (and surrounding systems) towards

this goal means that effort is not wasted on improving classifier accuracy when that effort would produce greater returns if directed towards better control.

The goal of this study is to explore the relationship between classifier accuracy and control performance when a system is faced with a source of classifier corruption. This will be done by formulating a model neural system to be controlled, along with various models of observational corruption.

3.1.2 Types of Spike Sorting Corruption

In general, spike sorters attempt to maximize isolation of an individual neuron based on clustering of features extracted from the neuron's spike waveform. A perfect sorter will include all spikes from the target neuron into the cluster (perfect recall), and will reject all spikes from non-target neurons (perfect precision). False positives occur when non-target neuron spikes are included in the cluster ("spike addition"), and false negatives occurs when target neuron spikes are not included in the cluster ("spike exclusion" or "spike deletion"). Additions and exclusions/deletions from the cluster can occur due to a number of different reasons. I will focus first on how cluster definitions affect error rates, and later on how spike collisions can lead to false negatives regardless of the sorter's settings.

The three cases I consider are spike exclusion, spike addition, and spike deletion. Consider a target neuron, N_A, whose spikes we are attempting to isolate from background activity using a sorter. The process of identifying clusters inherently involves trade-offs between precision and recall, and while precision is affected by the characteristics of

nearby non-targets (how different they are from N_A, and therefore how easily they are excluded from the cluster), recall is determined only by how restrictive the cluster's boundaries are. The simplest form of corruption, spike exclusion, stems from the case tightness of the cluster's boundaries lead to poor recall. This means that the unit isolated by the spike sorter, denoted by N_{AP}, exhibits a lower firing probability than the true neuron N_A.

For the other two cases, consider a permissive spike sorter, such the recall is high. A side-effect of making the spike sorter more permissive is that precision will generally decrease. Spike addition, occurs when the sorter includes spikes from non-target neurons in the cluster.

In addition to the target neuron N_A, consider an additional neuron N_C, which acts as a corrupting signal. Spike addition can occur if N_C has spike waveforms that are similar enough to N_A's spike waveforms that the sorter includes them in the cluster. Addition is a function of N_C spike waveforms, but not their timing. The result is that the isolated unit N_{AP} exhibits a higher firing probability than the true neuron N_A.

The final case, spike deletion, is largely insensitive both to the settings of the spike sorter and the fine details of spike waveforms. Spike deletion occurs when any other waveform, from either single neuron spikes or multi-neuron "hash", temporally collides with target neuron spikes. Deletion is not a function of the corrupting waveform's shape, but rather of its timing and amplitude. This can cause the target waveform to be significantly and unpredictably altered by the corrupting waveform. Because the target waveform becomes corrupted, its extracted features no longer lie

within the cluster boundaries, and the spike is "deleted", or not recorded. This means that the isolated unit N_{AP} has a lower firing probability than true neuron N_A.

3.2 Methods

3.2.1 Rescaling of the Integrate and Fire Model to Explore Corruption

As in Chapter 2, the model system is a pair of stochastic Integrate and Fire (IAF) neurons whose characteristics satisfy sufficient conditions for underactuated control, as described by Ching & Ritt (2013). Those conditions are that the IAF parameters satisfy the relationships $\alpha_A > \alpha_B$, $\beta_A > \beta_B$, and $\frac{\alpha_A}{\beta_A} > \frac{\alpha_B}{\beta_B}$, for leak α and input strength β. Under these conditions in the deterministic case, a pair of rectangular stimuli S_A and S_B can be chosen to activate the neurons individually in any arbitrary sequence. I assume that the noise and neural parameters are such that this outcome continues to hold approximately in the stochastic case (for example, with small noise and wide margins on the inequalities). As a convention, the neuron with larger leak α will be denoted N_A, and called the "fast" neuron, due to the fact that its internal dynamics occur on a faster timescale. The fast neuron is more likely to fire in response to high amplitude, short duration stimuli. The "slow" neuron, N_B, is more likely to fire in response to low amplitude, long duration stimuli. The stimuli are characterized by their strength G in $\frac{mW}{mm^2}$ and duration T in ms, and chosen such that S_A will bias towards N_A activity, and S_B will bias towards N_B activity.

Because there is no time reference external to the neural system, the IAF equation

can be time rescaled to normalize the rate constant α of one of the neurons. I define one of the neurons as the "standard bearer" neuron, parameterized by $\theta_S = [\alpha_S, \beta_S, \sigma_S]$. I then rescale the IAF equation by defining $\tau = \alpha_S t$, and $dW_\tau = \sqrt{\alpha_S} dW_t$. For convenience, I also define $\gamma = \frac{\beta_S}{\alpha_s}$ and $\epsilon = \frac{\sigma_S}{\sqrt{\alpha_S}}$. The system then becomes

$$dV = \left(-r_\alpha V + r_\beta \gamma S(\tau)\right) \mathrm{d}\tau + r_\sigma \epsilon dW_\tau \tag{12}$$

where either neuron can be parameterized by the ratios $r_\theta = [r_\alpha, r_\beta, r_\sigma] = \left[\frac{\alpha}{\alpha_S}, \frac{\beta}{\beta_S}, \frac{\sigma}{\sigma_S}\right]$ between the neuron's true parameters and the standard bearer's parameter. A neuron parameterized by r_θ is controllable with the standard bearer neuron if either

$$r_\alpha > 1, \qquad r_\beta > 1, \qquad \frac{r_\alpha}{r_\beta} > 1 \tag{13}$$

or

$$r_\alpha < 1, \qquad r_\beta < 1, \qquad \frac{r_\alpha}{r_\beta} < 1 \tag{14}$$

3.2.2 A Probability-Based Framework for Modeling Corruption

Observations of the neurons in the system will be modeled by the probability of observing a target spike given some stimulus. This true firing probability will then be modulated according to one of the three corruption cases.

I first describe a probability-based framework for the three cases, and study how corruption changes the observed response characteristics of a recorded neuron. I then explore how these corruption types may lead to erroneous conclusions about the system being controlled.

I use three separate models spike exclusion, spike addition, and spike deletion.

For all cases, I assume that the spike sorting process is not affected by stimulation, specifically that neither the waveforms from a given neuron nor the probabilities of corruption change with different values of stimulation strength or duration. Further, I define the firing probability as the probability that at least one spike is recorded from the neuron of interest in some small time window starting at stimulus onset. In my experimental application, stimulation power is minimized to avoid activating the non-target neuron, so that typically either one or zero target spikes will occur.

3.2.3 Spike Exclusion

The first case to be considered is spike exclusion, in which the spike sorter is too restrictive, and each true target spike has some chance of being "missed". Define P_P as the observed firing probability of N_A, A as the event an N_A spike occurred, and E_{Ex} as an exclusion corruption event. The exclusion case can be modeled by

$$\begin{aligned} P_P &= P(A \cap \neg E_{Ex}) \\ &= P_A\, P(\neg E_{Ex} | A) \\ &= P_A(1 - P_{E_{Ex}}) \end{aligned} \tag{15}$$

where $P \equiv P(A)$ and $P_{E_{Ex}} \equiv P(E_{Ex})$. The first equation is the statement that the observed firing rate is the probability of an N_A spike occurring and an exclusion event not occurring. The second equation expands the joint probability by conditioning, and the third applies the assumed independence of the probability of corruption from the probability of spiking.

3.2.4 Spike Addition

For the two cases, I make two simplifications. First, I assume that no N_A spikes are lost, meaning that $P_{E_{Ex}}$ is 0. Second, I assume the presence of one or more corruption neurons, which are represented as either a single neuron N_C or as "hash". For spike addition, I assume that N_C has spike waveforms similar enough to N_A's waveforms that the spike sorter has some probability $P_{E_{Inc}}$ of erroneously including N_C's spikes in the cluster. I assume that N_A's and N_C's spike probabilities are conditionally independent given the stimulation strength and duration, which is a biologically plausible assumption in the absence of a fast synaptic coupling, either between N_A and N_C, or with a common pre-synaptic neuron.

Define C as the event a N_C spike occurs, and E_{Inc} as a corruption event in which an N_C spike is erroneously included in the N_A cluster. The model for this spike addition case is

$$\begin{aligned} P_P &= P\big(A \cup (C \cap E_{Inc})\big) \\ &= P_A + P(C \cap E_{Inc}) - P_A P(C \cap E_{Inc}) \\ &= P_A + P_C P_{E_{Inc}}(1 - P_A) \end{aligned} \tag{16}$$

where $P_C \equiv P(C)$ and $P_{E_{Inc}} \equiv P(E_{Inc})$. The first equation is the statement that an observed firing event requires that a true N_A spike occurred, or that a true N_C spike occurred and an inclusion corruption event occurs. The second equation expands the union of events and uses the assumed independence of N_A and N_C. The third equation asserts that inclusion corruption events are independent from N_C spiking. For this case, I assume that no spike collision occurs even if spike timings are similar.

3.2.5 Spike Deletion

Spike deletion occurs when waveforms are distorted by electrical hash; resulting from non-specific background activity of many neurons. This collective population activity is considered differently from individual neuron activity because as the energy of the stimulus pulse increases, higher population recruitment occurs, and there is a higher probability of interference occurring during the post-stimulation window. For this case, I consider hash whose activity is modeled logistically. If this activity is large enough, and it occurs temporally close to the target neuron's activity, then target spikes may be disrupted by the hash's activity, causing the spike sorter to "miss" them.

Define E_{Hash} as an event in which hash occurs in the post-stimulation window, and E_{Del} as a corruption event in which an N_A spike is missed by the spike sorter due to waveform distortion. The probability of observing an N_A spike is given by

$$\begin{aligned} P_P &= P\big(A \cap \neg(E_{Hash} \cap E_{Del})\big) \\ &= P_A P(\neg(E_{Hash} \cap E_{Del})|A) \\ &= P_A\big(1 - P_{E_{Hash}} P_{E_{Del}}\big) \end{aligned} \tag{17}$$

The first equation is a statement that an observed spike occurs when N_A spikes and either no hash occurs, or such hash does not have the shape or timing to significantly distort the N_A waveform. The second line expands the joint probability by conditioning. The third equation asserts that neither the probability of hash occurrence nor the probability of hash being large enough to delete an N_A spike is dependent on whether or not N_A fires. It also asserts that the recruitment of hash and its distorting effects are independent from each other.

3.2.6 Neuron and Hash Models

A Fokker-Planck (FP) implementation of the IAF model was used to calculate spiking probabilities of neurons given their parameterizations $r_\theta = [r_\alpha,\ r_\beta,\ r_\sigma]$ and a stimulus $S = [G, T]$. The FP integration was implemented in the same way as in Chapter 2. In particular, because numerical integration of the FP equation is computationally expensive, a table of pre-calculated firing probability was interpolated for computations below. I chose parameters for a pair N_A and N_B of neurons to control, as well as a neuron N_C that corrupted the simulated spike sorting through spike addition.

For the spike addition case, a set of 3 corrupting neuron instances N_C were tested, with different values for $r_{\theta_C} = [r_{\alpha_C},\ r_{\beta_C},\ r_{\sigma_C}]$ selected to explore addition corruption across various relationships between N_A and N_C. The parameterizations shared the same r_α and r_σ as N_A, but the value of r_{β_C}, representing sensitivity to stimulation, was varied. The sensitivity r_{β_C} for each instance of N_C was found by first defining N_C's ideal firing probability P_{Ideal} in response to certain stimuli along N_A's SD curve, and then calculating a value for r_β that would come closest to matching that firing probability. To find each r_{β_C}, I first fixed a set of $n = 100$ equally spaced stimulation durations, T_{N_C}, in the interval $B_{N_C} = [0, 15]$ ms. The strength-duration (SD) curve was then calculated for both N_A and N_B, denoted by G_{Curve_A} and G_{Curve_B} respectively. The SD curve is defined as the set of strengths G_{Curve} that cause a neuron to fire at some probability P_{SD}. I will be considering the curves generated by $P_{SD} = 0.5$ throughout. The SD curves of both neurons were found by calculating

(18)

$$G_{Curve_X} = \underset{S}{\operatorname{argmin}} \sum_{D_{NC}} (P(N_X|G_i, T_i, r_\theta) - P_{SD})^2$$

where G_{Curve_X} is the SD curve of neuron N_X, $P(N_X|G_i, T_i, r_\theta)$ is the firing probability of neuron N_X given some stimulus duration D_i, stimulus strength S_i, and neuron parameterization r_θ.

For each instance of N_C, a desired firing probability was determined, $P_{Ideal} = [.5, .85, .15]$. For each of these desired firing probabilities, $r_{\beta_{C_K}}$ was found by minimizing the sum squared error between the firing probability of the FP model given r_{θ_C} with variable r_{β_C} and each P_{Ideal} value, on the domain $[G_{Curve_A}, T_{N_C}]$.

$$r_{\beta_{C_K}} = \underset{r_\beta}{\operatorname{argmin}} \sum \left(P\left(N|\, G_{Curve_A}\, T_{N_C}, \left[r_{\alpha_C}\ \mathrm{r}_\beta\ \mathrm{r}_{\sigma_C}\right]\right) - P_{Ideal_K}\right)^2 \tag{19}$$

The "firing rate" of hash, that is, the probability of a significant amplitude of hash occurring following a stimulus with parameters $S = [G, T]$, was modeled logistically as a function of the energy GT of the stimulus, via

$$P_{Hash} = \frac{1}{1 + e^{-k(GT - E_{Half})}} \tag{20}$$

The probability is characterized by the two parameters: E_{Half}, the stimulation energy in $\frac{mW\ ms}{mm^2}$ required to produce a 50% occurrence probability for the hash, and k, the sensitivity of the probability to stimulus energy. As in the spike addition case, three parameterizations of the hash were considered. They were chosen to behave similarly to the N_C instances described above. A maximum of 5 ms was used to weight the fit towards low stimulation durations, so that the hash would have similar behavior to N_C in the duration regime in which N_A's control stimulus S_A would most likely lie.

I defined a subset T_{Hash} of T_{N_C}, containing equally spaced stimulation durations in the interval $B_{T_{Hash}} = [0, 5]$ ms, $n_{Hash} = 34$. I then defined a subset $G_{Curve_{C_{Hash}}}$ as the stimulation strengths in G_{Curve_C} that correspond to the durations in T_{Hash}. The parameter E_{Half} was calculated for each hash instance by finding the mean stimulation power

$$E_{Half} = \frac{T_{Hash}\ G_{Curve_{C_{Hash}}}}{n_{Hash}} \tag{21}$$

where $n_{Hash} = 34$ is the number of durations in T_{Hash}. A set of n_{Hash} equally spaced stimulation strengths S_{Hash} was then generated with the bounds $B_{S_{Hash}} = [0\ 2\ \max(G_{Curve_A})]$, where G_{Curve_A} is the set of strengths that define the strength-duration curve for N_A on the domain T_{Hash}. The parameter k was calculated for each hash instance by minimizing the sum squared error between the logistic model parameters and the spike probabilities of the corresponding instance of N_C on the grid of all $[G_{Hash}, T_{Hash}]$.

3.3 Results

3.3.1 Exploration of Corruption Paradigms

To build an intuition for the effects of the various corruption cases, I examined the differences in observed strength-duration curves from their true values. The strength-duration curve is a simple metric to characterize a neuron's response characteristics. For example, if the spike sorter adds spikes, the estimated curve will move down and/or left in strength-duration space, because weaker stimuli will appear to produce a spike probability of $P_{SD} = 0.5$.

All simulations used a fixed parameterization $r_{\theta_A} = [1,\ 1,\ 1]$ and $r_{\theta_B} = [0.071,\ 0.571,\ 1]$ for N_A and N_B, which are mutually controllable according to necessary and sufficient conditions in Ching & Ritt (2013). In all cases, observations of N_B are uncorrupted; corruption occurred in the sorting of N_A. The parameter values for N_C during the spike addition case varied according to Table 3. In the first simulation, N_C has identical parameters to N_A. In the other simulations, N_C is either more or far less sensitive to stimulation than N_A, represented by changes to β_C. The parameterization of the multi-unit activity in the spike deletion case is given by Table 4. Each instance of hash is modeled to behave similarly to the corresponding instance of N_C.

	α	β	σ
Simulation 1	1	1	1
Simulation 2	1	1.38	1
Simulation 3	1	0.66	1

Table 3: IAF parameters for the corrupting neuron NC

	K	E_{Half}
Simulation 1	.628	7.435
Simulation 2	.783	5.388
Simulation 3	.424	11.284

Table 4: Logistic parameters for the corrupting hash

3.3.2 Exploration of Spike Exclusion

In the excluded spikes case, as the corruption level increases, the apparent sensitivity of the neuron decreases. This means that the observed strength-duration curve rises, as can be seen in Figure 13. As expressed in the corruption equation Eq 15, the rise in the SD curve is proportional across all stimulation durations, and increases monotonically with the corruption level. Clearly, if $P_{E_{Ex}} = 0$, then $P_P = P_A$, since in the absence of corruption the observed firing probability is equal to the true firing probability. Also, if $P_{E_{EX}} = 1$, indicating that every spike is lost, $P_P = 0$. In particular, above a certain value of $P_{E_{Ex}}$, the level of corruption will be so great that N_{A_P} will not appear to spike for half of the stimuli presented, regardless of their strengths. SD curves in this paper are defined at the 50% firing probability level. Therefore, for high values of corruption, the SD is undefined. In the limit as the corruption level goes to some critical value, the SD curve will approach infinity. For this reason, only a subset of SD curves (depending on the specific properties of the neuron(s)) will be plotted for the deletion corruption cases.

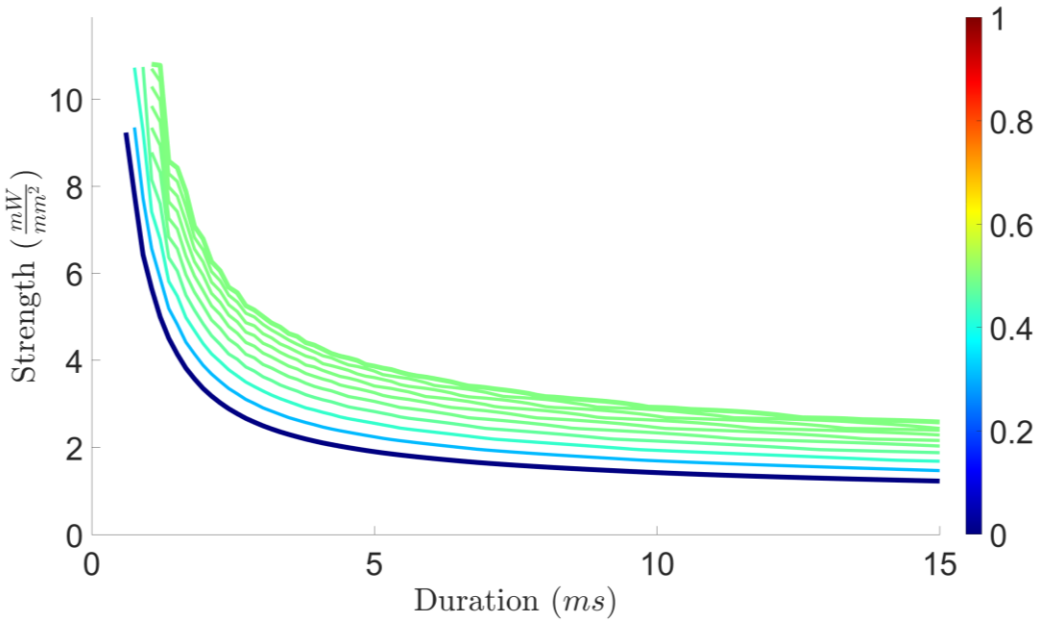

Figure 13: The SD curves of a neuron under exclusion corruption, with lines colored by their value of $P_{E_{Ex}}$

Values of $P_{E_{Ex}}$ logarithmically approach .5. No SD curve is defined for $P_{E_{Ex}} \geq .5$.

3.3.3 Exploration of Spike Addition

In the spike addition case, as expected, small amounts of corruption yield observed SD curves that are very similar to N_A. In the first simulation neuron N_A and N_C have identical parameterization ($\theta_A = \theta_C$). In this case, corruption level has a relatively small impact on the location of the SD curve, as seen in Figure 14. For high values of corruption ($P_{E_{Inc}} = 1$), the observed firing probability (Eq. 16) becomes

$$\begin{aligned} P_P &= P(A \cup C) \\ &= P_A + P_C - P_A P_C \\ &= 2P_A - P_A^2 \end{aligned} \tag{22}$$

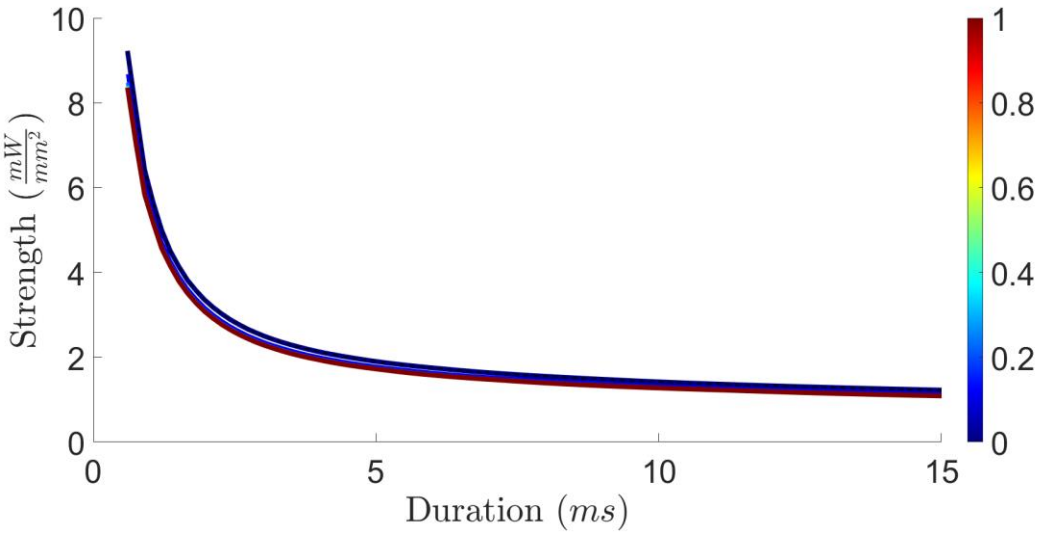

Figure 14: The SD curves of a neuron under addition corruption from another neuron with identical parameters, with lines colored by their value of $P_{E_{Inc}}$.

A dashed curve representing the SD curve of N_C is shown, but obscured by the blue SD curve.

The high corruption case is represented by the red curve in Figure 14. There is very little change in the SD curve in this instance because the firing probability is the probability of either N_A or N_C firing. For the inclusion case, the firing probability is dominated by the more sensitive neuron, as I will show in the next two cases later. This is due to the fact that there is no increase in firing probability when both neurons fire, as compared to only one neuron firing. When both neurons have the same firing probability, it is more likely that an observed N_A spike will be recorded, but only by a small amount.

Despite the small change, this may lead to worse control, since corruption increases the apparent sensitivity of N_A, leading to a higher rate of false alarms under stimulus S_B. As seen in Figure 14, the estimated SD curve lowers as corruption

increases, until possibly violating the necessary condition for control with neuron B. It is worth noting that SD curves may change by small amounts in SD space, but lead to large changes in the calculated optimal stimuli, which could lead to significant decreases in controller performance.

The second simulation demonstrates a case in which NC has a significantly higher sensitivity than N_A ($r_{\beta_C} > r_{\beta_A}$). A loss of control also occurs as corruption level increases, with a faster increase in apparent sensitivity relative to the level of corruption than in the $\theta_A = \theta_C$ case (Figure 15). This is due to the fact that spike-addition corruption is a function of both the probability that N_C fires within the post-stimulation window, and the probability that such a spike would be misclassified as an N_A spike. Because of the increased firing probability of N_C, more spikes are "available" to be added. A similar effect would occur if C had a high spontaneous firing rate (large σ and relatively small α), regardless of sensitivity to the stimulus. As P_C increases to 1, the corruption equation (Eq. 16) reduces to

$$\begin{aligned} P_P &= P(A \cup E_{Inc}) \\ &= P_A + P_{E_{Inc}}(1 - P_A) \end{aligned} \tag{23}$$

Therefore, in the case of corruption by a neuron that is highly sensitive or that has a high spontaneous firing rate, the observed firing rate will increase linearly as a function of the error rate of the spike sorter. In this case, the specificity of the spike sorter may have direct implications on the quality of control, as the probability of sorting errors will directly increase the estimated firing probability of N_A.

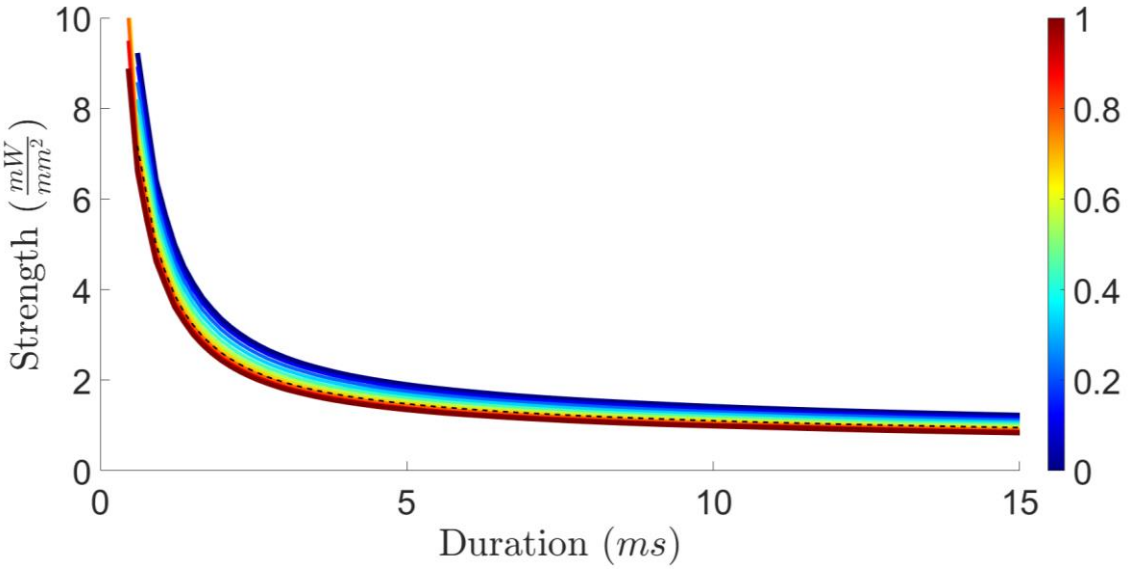

Figure 15: The SD curves of a neuron under addition corruption from another neuron that has higher sensitivity, with lines colored by their value of $P_{E_{Inc}}$.

The dashed curve shows the SD curve of N_C.

The third addition-corruption simulation assumes that N_C has a significantly lower sensitivity to the stimulus than N_A ($r_{\beta_C} < r_{\beta_A}$). In this case, N_C has a significantly lower firing probability than N_A. Therefore, the probability of N_C spikes being added to the cluster is low, regardless of $P_{E_{Inc}}$, and, as seen in Figure 16, the estimated SD curve changes very little. This case has the smallest likely impact on control, as it is unlikely that N_C will spike at all. As P_C decreases to 0, the framework's corruption equation reduces to $P_P = P_A$, so that the estimated firing rate becomes identical to the neuron's true firing rate, regardless of the spike sorter's performance.

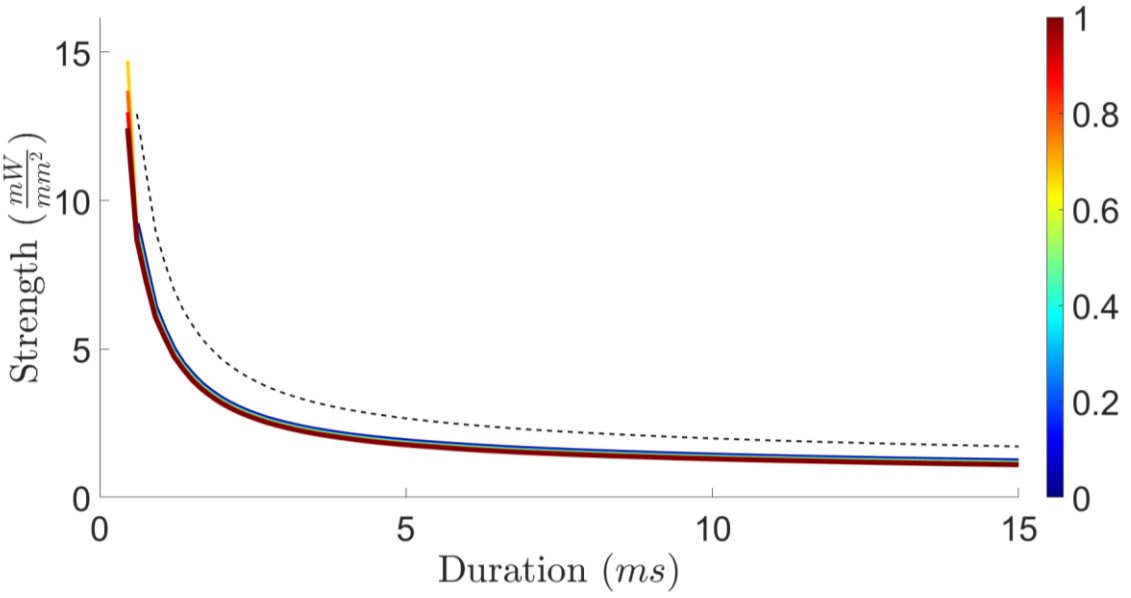

Figure 16: The SD curves of a neuron under addition corruption from another neuron that has lower sensitivity, with lines colored by their value of $P_{E_{Inc}}$.

The dashed curve shows the SD curve of N_C.

3.3.4 Exploration of Spike Deletion

In the spike deletion case, the estimated firing probability is

$$P_P = P_A\left(1 - \frac{P_{E_{Del}}}{1 + e^{-k(GT - E_{Half})}}\right) \tag{24}$$

Small amounts of corruption $P_{E_{Del}}$ yield SD curves that are similar to the uncorrupted SD curve. As the level of corruption $P_{E_{Del}}$ increases, the likelihood of hash being high enough amplitude to delete an existing N_A spike increases. This increase in corruption leads to a decrease in the apparent sensitivity of N_A, manifesting as a rise in the SD curve (Figure 17). This hash model has some non-zero probability of hash occurrence even with $G = 0$ (a "sham" stimulus), although k and E_{Half} are chosen such that this probability is small ($P_{Hash_{Sham}} < .015$ for all tested parameterizations). At a

given stimulus energy level, it may happen that a high enough corruption probability $P_{E_{Del}}$ pushes the apparent spike probability below ½, even if $P_A = 1$. This phenomenon is similar to the exclusion case, and means that some corruption levels will have a partially defined SD curve, and others may have no SD curve.

For the first spike deletion case, the corrupting hash has an occurrence probability that is generally similar to N_A's firing probability at various stimulation energies. As can be seen in Figure 17, non-zero levels of corruption will lead to an apparent decrease in sensitivity of N_A, and therefore a rise in the SD curve. Unlike the exclusion case, the impact is now non-uniform across SD space.

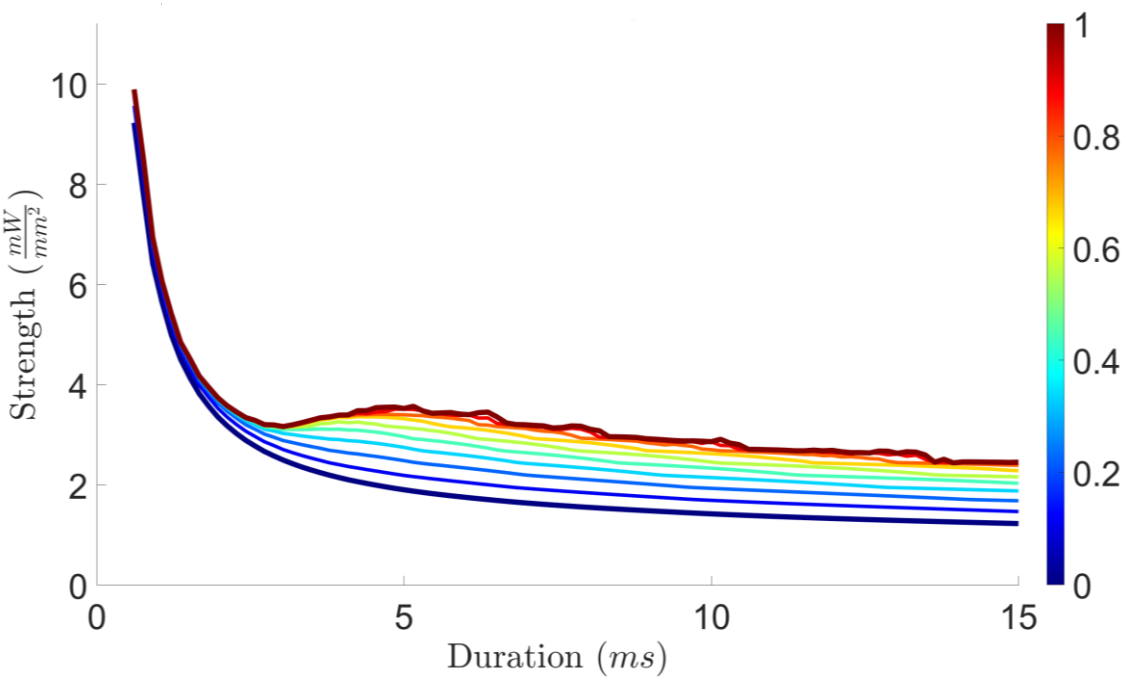

Figure 17: The SD curve of a neuron under deletion corruption by hash with comparable sensitivity to the neuron, with lines colored by their value of $P_{E_{Del}}$

In the second spike deletion case, the corrupting hash is significantly more sensitive than N_A. This manifests itself as undefined SD curves above a corruption level of $P_{E_{Del}} = .5$, due to the fact that the hash given by the logistic model will fire nearly 100% of the time that N_A fires. This case produces an apparent firing probability that reduces to

$$\begin{aligned} P_P &= P_A\left(1 - \frac{P_{E_{Del}}}{1 + e^{-k(GT - E_{Half})}}\right) \\ &\approx P_A(1 - P_{E_{Del}}) \end{aligned} \tag{25}$$

This is very similar to the spike exclusion equation, which is seen in the similar way in which SD curves change across corruption levels (Figure 18). As the corruption level increases, eventually the apparent response is unable to produce a 50% firing probability for stimuli above a certain energy. This is due to the fact that the interfering hash has become active enough that most N_A spikes are deleted before they are recorded by the spike sorter.

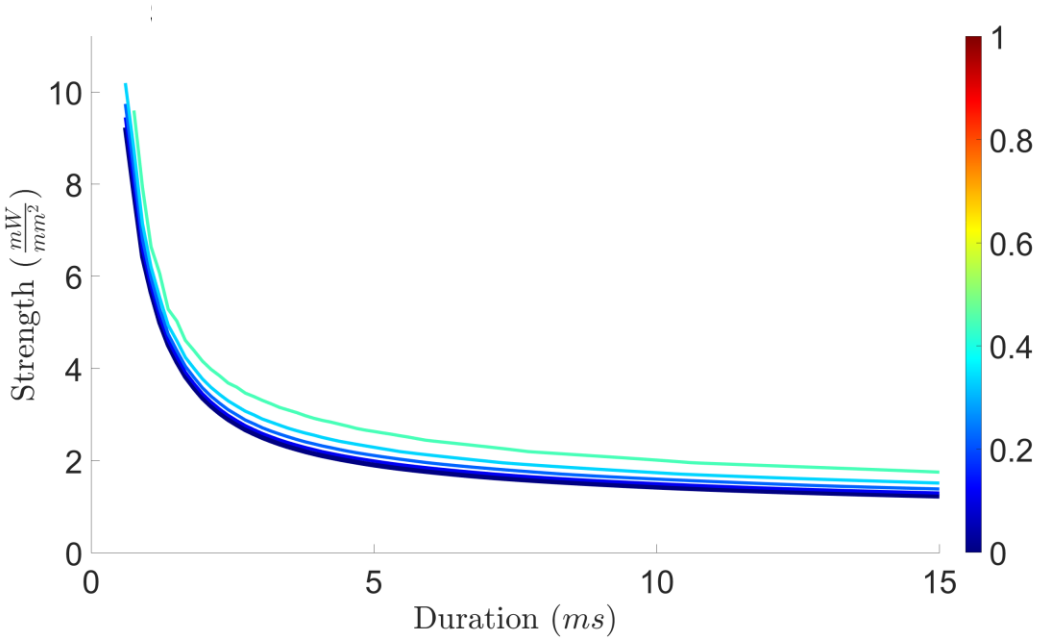

Figure 18: The SD curve of a neuron under deletion corruption by hash with high sensitivity, with lines are colored by their value of $P_{E_{Del}}$. No SD curve is defined for high levels of corruption

In the third spike deletion case, the corrupting hash is less sensitive than N_A for most stimuli. Thus, an SD curve can be defined for higher levels of corruption than with more sensitive hash (Figure 19). However, high duration stimuli of any strength will still produce enough corruptive hash events that above a certain duration, the SD curve could be undefined.

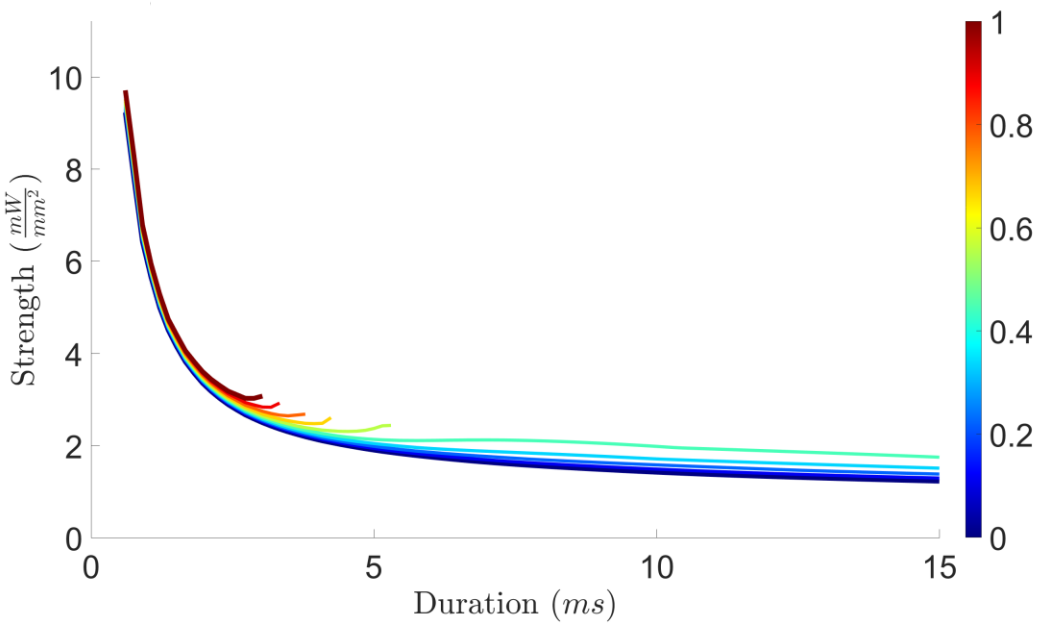

Figure 19: The SD curve of a neuron under deletion corruption by hash with low sensitivity, with lines colored by their value of $P_{E_{Del}}$. No SD curves (or only partial SD curves) are defined for high levels of corruption.

The partial SD curves do not simply terminate, however. Because these SD curves represent isolines of firing probability curves over SD space, which is continuous, the SD curves themselves must also be continuous. The apparent firing probability in the deletion case takes the form

$$P_P(S) = f(S)\big(1 - g(S)\big) \tag{26}$$

where $S = [G, T]$. P_P is thus a surface of two dimensional strength-duration space, and the SD curves are the level curves for $P_P = \frac{1}{2}$. For a given duration, as strength increases from zero, so does firing probability due to increased N_A spiking (represented by f). However, eventually a maximum apparent firing probability could be encountered, due to the effect of the hash through $g(S)$, after which the firing probability begins decreasing.

Therefore, the full SD plot for the deleted spikes case can include a "doubled" SD curve, as seen in Figure 20. This case can lead to very poor control if the controller is not properly designed, because cases in which stimuli are powerful enough to push P_P beyond its maxima will cause significant overdriving of N_A on the "high" branch of the apparent SD curve.

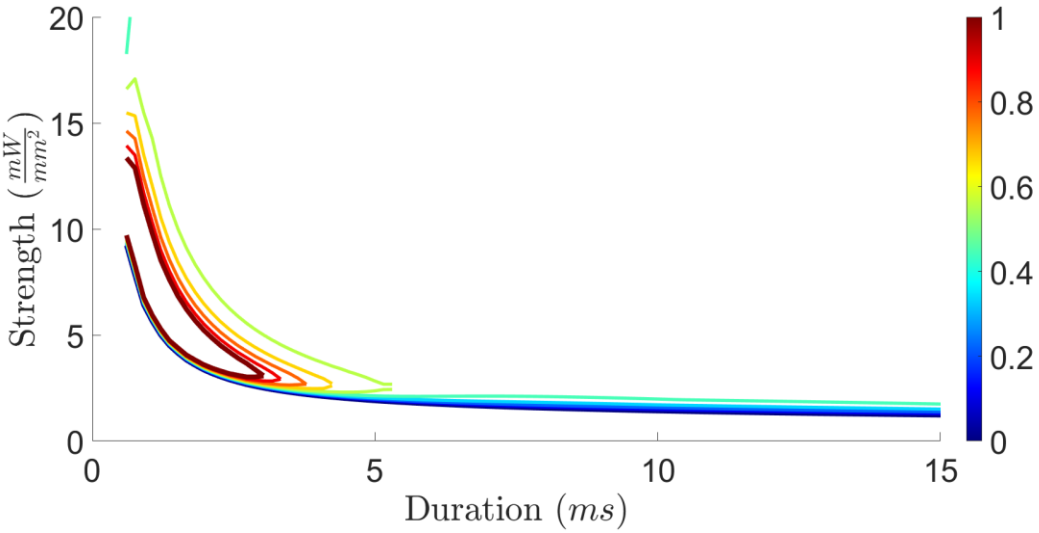

Figure 20: The SD curve of a neuron under deletion corruption by hash with high sensitivity, including the "doubled" SD curves.

The firing rate starts decreasing as strength increases due to an increased number of hash collisions. Lines are colored by their value of $P_{E_{Del}}$.

However, while this case is pathological, it is easily avoided in practice. The "upper branch" of the SD curve will be ignored for the remainder of this dissertation, as this double-SD curve artifact can be rejected by using a governor on the spike sorter that tracks total threshold crossings or estimated SD-curves. If a large number of unsorted threshold crossings occur in a given post-stimulation window, then there may be a large amount of hash corrupting a signal, despite seeing one or zero N_A spikes as classified by

the spike sorter. Furthermore, a well-designed system may also trigger an error if the firing probability of a neuron appears to decrease as stimulation strength increases. Such behavior may be physiologically possible, such as if stimulating the recorded neuron indirectly via an inhibitory presynaptic cell, but, due to the expected rarity of such a physiological case and the ease with which "doubled" SD curves can be rejected in practice, I restrict study below to the lower branch of SD curves, for which spike probability P_P monotonically increases with stimulus strength.

3.3.5 Boundaries in neural parameter space

Using the intuition from the previous section about how each neuron's SD curve tends to change, one can study trade-offs between the parameters used in the corruption models. The framework that I use, underactuated control of a pair of neurons, implicitly defines both lower and upper limits on the firing probability of the two neurons when subject to the chosen stimuli. For example, when N_A (the neuron in the pair whose α and β are larger, also called the fast neuron) is subject to the short duration stimulus, it is the target neuron, so it must have a higher firing probability than N_B, as seen in Figure 21.A. Said another way, at some chosen short duration, N_A's SD curve must be lower than N_B's curve, implying that it is more likely to fire at lower strengths. This means that, at the strength on N_B's curve at that low duration, N_A must have a firing probability $P_A > 0.5$. Conversely, at the strength on N_B's curve at a high duration, as seen in Figure 21.C, N_A must have a firing probability $P_A < 0.5$. If these two conditions are met, then the neuron pair is controllable.

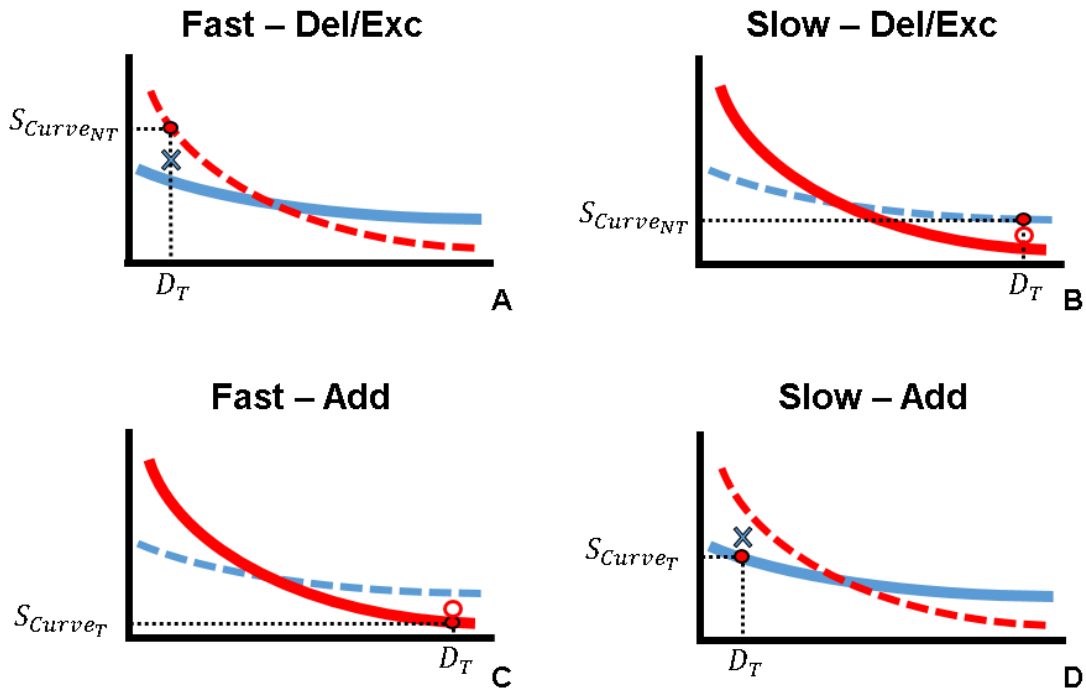

Figure 21: The thresholds of losing controllability due to corruption affecting either the fast (blue) or slow (red) neuron.

Each panel's title indicates the neuron type affected and the type of corruption. When the SD curve of the affected neuron appears to cross the red dot due to corruption, then the pair will appear to lose controllability.

There are two categorical ways in which corruption may "fool" a control system: it may cause the system to believe that the pair is not controllable when it is, or that the pair is controllable when it is not. Each corruption type will either raise or lower the SD curve, based on whether spikes are erroneously being removed from or added to the cluster. For the case in which the neural system is in fact controllable, corruption may cause observed uncontrollability in two ways. Corruption may cause the measured SD curves to uncross on one side, meaning that no choice of stimuli will allow both neurons to be selectively activated. Alternatively, even if the corrupted curves appear to meet controllability conditions, shifts in those SD curves may cause the estimated optimal stimuli to leave the

controllability zone of the true SD curves. This effect is dependent on the choice of cost function. A stimulus leaving the control region will cause either the target neuron to fire at below 50% probability, or the non-target neuron to fire above 50% probability. Figure 22 demonstrates the case in which corruption causes the estimated SD curves to uncross each other.

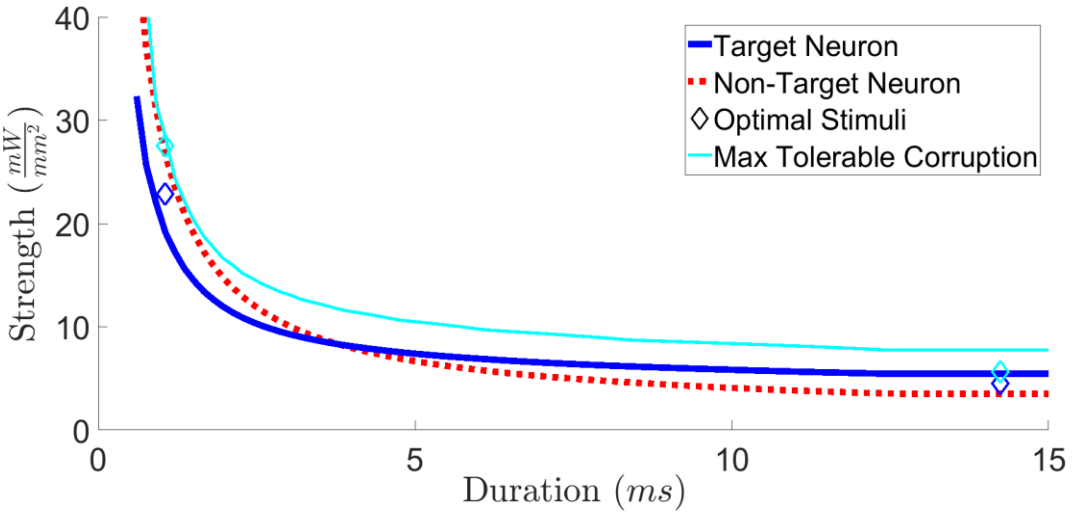

Figure 22: A demonstration of the two control failure modes during corruption

The dark blue curve represents an uncorrupted neuron that is controllable with the red neuron. The cyan curve shows the observed SD curve of the blue neuron under a given amount of exclusion corruption, and the diamonds show the calculated optimal GT. On the left side, the corrupted neuron appears to "uncross" with the red neuron, apparently losing its controllability condition. On the right side, the neurons stay crossed, but the optimal GT value moves above the blue neuron's SD curve.

For the case in which the neural system is not controllable, corruption may cause apparent controllability. This may happen if the corruption causes the estimated SD curves of the neurons to cross, thereby satisfying the controllability condition. This can

happen only if a highly sensitive neuron is subject to subtractive corruption, or an insensitive neuron is subject to additive corruption.

To assess the interplay between the parameters in a given type of corruption, the maximum tolerable corruption probability $P_{Corr_{Max}}$ was found for a variety of N_A, N_B pairs. $P_{Corr_{Max}}$ is defined as the largest probability of corruption that can be tolerated by the system without causing the control system to incorrectly classify the pair's controllability. It can be used to test for both observed gain and loss of controllability, because it is a local test; for some neuron and stimulus parameterization, it will find the corruption level at which the curves will appear to switch their order (which neuron will fire first as stimulation strength increases, or which curve is above the other).

Because the $P_{Corr_{Max}}$ calculation tests for both gain and loss of controllability, I simplify the study and consider only a neuron pair that is controllable, with corruption that makes them appear uncontrollable. Further, full analysis of the case of true controllability but misestimated stimuli requires choosing a cost function to define optimal stimuli, as well as analytically calculating the movement of the SD curve, which is challenging for the IAF neuron as there is no known closed form expression for the firing probability. Because of this, I will only be examining the case of appearing to lose controllability due to corruption through SD curve uncrossing.

The effect of addition corruption will be considered only for non-target neurons, as corruption increases the estimated firing probability (Figure 21.C and Figure 21.D). Thus, addition may make an inactive neuron appear active, but not lead to other errors. When analyzing the strength S_{Curve_T} on the target neuron's SD curve at the target

neuron's preferred duration T_T (the red dots in Figure 21), the non-target's firing probability must be below the target's firing probability. In other words, if the neuron pair appears to be controllable, it must be true that at the point $[G_{Curve_T}, T_T]$, $P_T > P_{NT_{Perceived}}$, so $0.5 > P_{NT_{Perceived}}(G_{Curve_T}, T_T)$. Therefore, the maximum tolerable probability $P_{Corr_{Max}}$ of corruption is the level of addition corruption that will cause the non-target neuron to appear to fire above 50% probability. Therefore, $P_T(G_{Curve_T}, T_T) = 0.5$ and $P_{NT_{Perceived}}(G_{Curve_T}, T_T)$ must be within the bounds [0 .5).

By similar logic, it must be true for a controllable pair that at the point $[G_{Curve_{NT}}, T_T]$, $P_{T_{Perceived}} > P_{NT}$, so $P_{T_{Perceived}}(G_{Curve_{NT}}, T_T) > 0.5$. $P_{Corr_{Max}}$ for deletion or exclusion corruption (corruption which will affect the neuron when it is the target of stimulation) is the level of corruption that will cause the target neuron to fire below 50% probability. In this case, the strength is on the non-target neuron's SD curve, at a duration the target prefers. This is demonstrated in Figure 21, panels A and B. Therefore, $P_{NT}(G_{Curve_{NT}}, T_T) = 0.5$, $P_{T_{Perceived}}(G_{Curve_{NT}}, T_T)$ must be within the bounds (.5 1].

To explore the controllability of a neuron pair where one neuron is corrupted, I will examine the change in the corrupted neuron's apparent SD curve behavior. This examination will be done at a fixed stimulus. The stimulus will be chosen such that it lays along the SD curve of the non-corrupted neuron in the pair, at both a short and long duration.

3.3.6 Parameter Boundaries of Spike Exclusion

For the exclusion case, a number of corruption target neurons were tested. For the pair to appear to be controllable, it must follow that

$$P_{Perceived} = P_T\,(1 - P_{Exc}\,) > .5$$
$$P_{Exc} < 1 - \frac{.5}{P_T} \tag{27}$$

If this is true for any parameterization of the target neuron under some corruption probability P_{Exc}, then the pair will still appear to be controllable. The maximum tolerable exclusion $P_{Exc_{Max}}$ can then therefore be defined as

$$\mathrm{P_{Exc_{Max}}} = 1 - \frac{.5}{\mathrm{P_T}} \tag{28}$$

Note that as the true firing probability increases, so too does $P_{Exc_{Max}}$.

An ensemble of target neurons was defined on a 100x100 grid of $[r_{\alpha_T}, r_{\beta_T}]$, in the range of $[.5, 2]$, with $r_\theta = 1$. The maximum tolerable corruption probability $P_{Exc_{Max}}$ was then found between each corrupted target neuron and the standard bearer neuron. The value of $P_{Exc_{Max}}$ was calculated at two stimuli on the standard bearer neuron's SD curve, one long duration and one short. Note that some tested target neurons do not meet the necessary and sufficient controllability conditions with the standard bearer neuron, but are included in the dataset regardless.

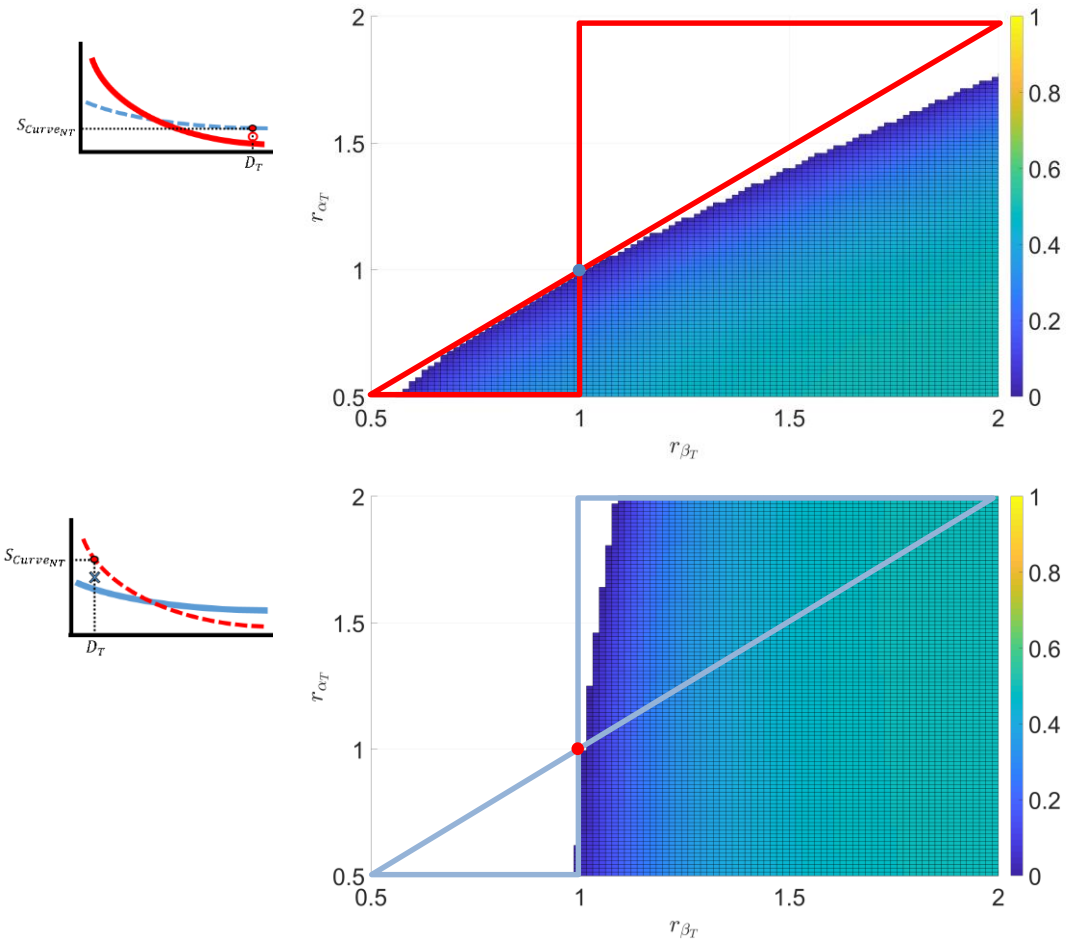

Figure 23: The maximum P_{Exc}, indicated by color, that can be tolerated by a neuron parameterized by $r_{\theta_T} = [\alpha_T, \beta_T, 1]$ when being controlled with a standard bearer neuron.

The top panel is for a corrupted red neuron under a long stimulus, $GT = [3.5, 14.2]$ (on the SD curve of a "fast" neuron). The bottom panel is shown for a corrupted blue neuron under a short stimulus, $GT = [26.6, 1.1]$ (on the SD curve of a "slow" neuron). The boundaries shown on each panel indicate the area representing neurons that are fully controllable with the standard bearer neuron, indicated by the dot at [1, 1]. The white area on the left indicates neurons that fire at less than 50% firing probability at the tested GT, and therefore cannot tolerate exclusion corruption because they are not controllable with the standard bearer to begin with. The colored area indicates neurons that fire above 50% firing probability at the tested GT, and therefore can tolerate some amount of exclusion corruption.

As seen in Figure 23, $P_{Exc_{Max}}$ is proportional to the firing probability of the target neuron P_T, which increases with β and decreases with α, for a given stimulus. It is worth

noting that the α leakiness parameter plays a more significant role in determining firing probability for longer duration stimuli, whereas the β sensitivity influences the firing probability at any stimulation duration.

For any given parameterization, a higher value of β or a lower value of α leads to a higher firing probability, and therefore a higher tolerance for exclusion corruption. This is true for both fast and slow neurons. This therefore means that, to make any given neuron more tolerant to exclusion corruption, β should be increased and/or α should be decreased.

3.3.7 Parameter Boundaries of Spike Addition

For the addition case, the corrupted neuron is the non-target neuron. For the pair to appear to be controllable, it must follow that

$$\begin{gathered} P_P = P_{NT} + P_C P_{Inc} - P_{NT} P_C P_{Inc} < .5 \\ P_C\, P_{Inc} < \frac{.5 - P_{NT}}{1 - P_{NT}} \end{gathered} \tag{29}$$

If this inequality holds, then the neuron pair will appear to be controllable. The maximum tolerable corruption can then be defined as

$$P_{Corr_{Max}} = P_C P_{Inc_{Max}} = \frac{.5 - P_{NT}}{1 - P_{NT}} \tag{30}$$

Two non-target neuron instances were tested, one that is faster than the standard bearer and one that is slower. For both of these instances, the stimulus in which the neuron is the non-target (the long stimulus S_B for the fast neuron, and the short stimulus S_A for the slow neuron) was used. For both neurons, $P_{Corr_{Max}}$ was calculated as a

function of the true non-target neuron firing probability. A 100x100 grid of the corruption neuron parameters $[r_{\alpha_C}, r_{\beta_C}]$ were tested in the range of $[.5, 2]$. For each corruption neuron instance, the firing probability P_C was calculated at the given stimuli. The maximum tolerable inclusion corruption $P_{Inc_{Max}}$ was calculated as $P_{Inc_{Max}} = \frac{P_{Corr_{Max}}}{P_C}$.

Figure 24 shows that the maximum tolerable inclusion corruption is a function primarily of how likely it is for the corruption neuron to fire, which increases as β increases and α decreases. At low firing probabilities, such as those given by low $r_{\beta C}$ and/or high $r_{\alpha C}$, significantly more corruption is tolerable before the corrupted neuron is observed to be uncontrollable. While these plots are quantitatively different, the difference in shape between the two is due primarily to the difference in stimulus duration, because r_{α_C} affects firing probability more during long duration stimuli.

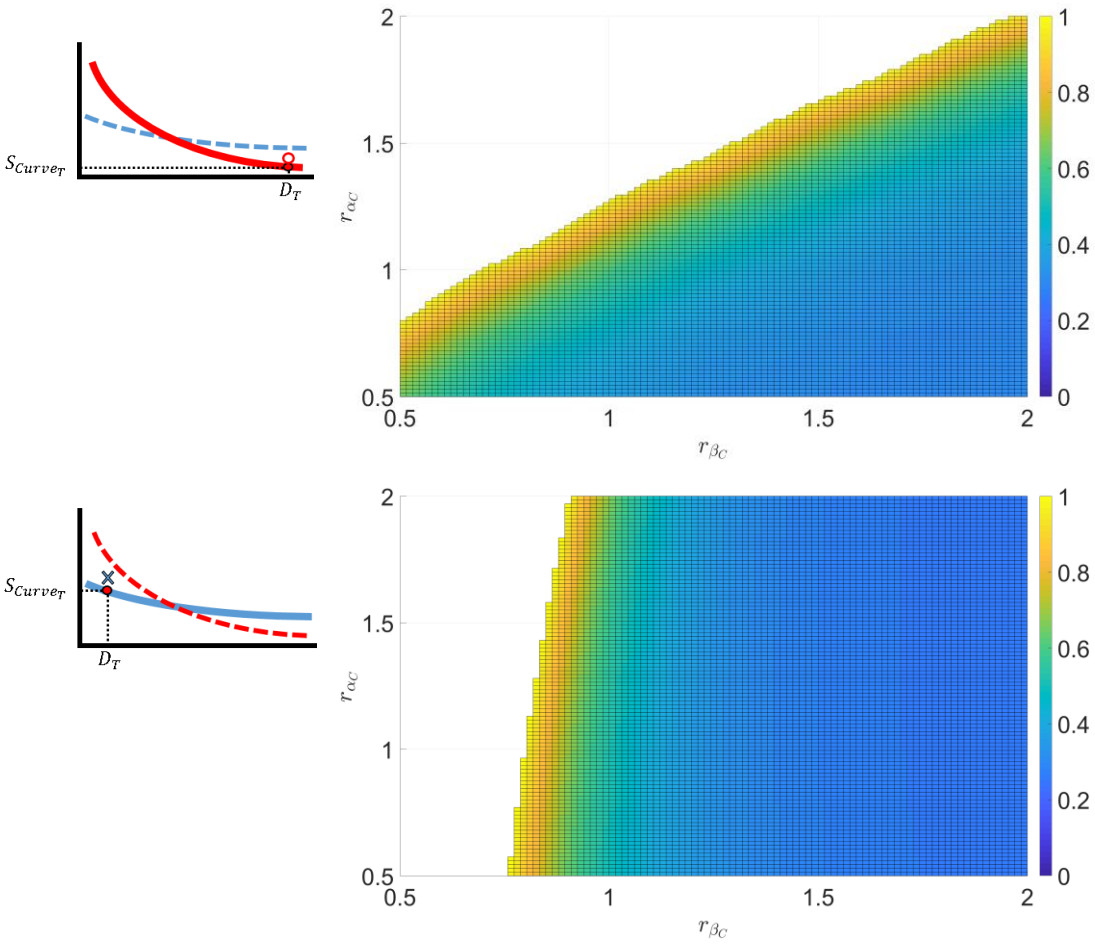

Figure 24: The maximum P_{Inc} that can be tolerated by a non-target neuron, across parameterizations of the corruptor neuron, when being controlled with a standard bearer neuron.

The top panel is shown for a corrupted blue non-target neuron with $r_{\theta_{NT}} = [1.5, 1.25, 1]$ and $GT = [3.5, 14.2]$ (on the SD curve of a "slower" standard bearer neuron). The bottom panel is shown for a corrupted red non-target neuron with $r_{\theta_{NT}} = [.67, .83, 1]$ and $GT = [26.6, 1.1]$ (on the SD curve of a "faster" neuron). The white area on the left indicates corruption neurons that never cause the non-target corrupted neuron to appear to fire at greater than 50% firing probability at the tested GT, and therefore can tolerate any value of P_{Inc} because the corrupted neurons always appears controllable with the standard bearer. The colored area indicates corruption neurons that may cause the non-target corrupted neuron to appear to fire at greater than 50% probability, and can therefore tolerate only the indicated amount of P_{Inc} without appearing to lose controllability.

3.3.8 Parameter Boundaries of Spike Deletion

For the hash deletion case, the corrupted neuron is the target neuron, as in the excluded spikes case. For the pair to appear to be controllable, it must follow that

$$P_P = P_T(1 - P_{Del}P_{Occur}) > .5$$

$$P_{Del}P_{Occur} < 1 - \frac{.5}{P_T} \tag{31}$$

If this inequality holds, then the neuron pair will appear to be controllable. The maximum tolerable corruption may therefore be defined as

$$P_{Corr_{Max}} = P_{Occur}P_{Del_{Max}} = 1 - \frac{.5}{P_T} \tag{32}$$

Like the addition case, two neuron instances were tested, but the stimulus in which the neuron was the target (the short stimulus S_A for the fast neuron, and the long stimulus S_B for the slow neuron) was used. $P_{Corr_{Max}}$ was calculated as a function of the true target neuron firing probability, and a 100x100 grid of the hash occurrence function parameters $[k, E_{Half}]$ were tested, in the range $k = [0, 5]$ and $E_{Half} = [0, \frac{E_{Max}}{2}]$ where E_{Max} is the maximum energy stimulus that I will reasonably expect to administer, $E_{Max} = 2\max(G_{Curve_S}) * T_{Max} = 55\frac{mW\ ms}{mm^2}$, where G_{Curve_S} is the set of strengths that represent the SD curve of the standard bearer neuron and T_{Max} is the duration of the maximum strength stimulus on the SD curve S_{Curve_S}. For each hash instance, the maximum tolerable hash deletion corruption $P_{Del_{Max}}$ was found, by $P_{Del_{Max}} = \frac{P_{Corr_{Max}}}{P_{Occur}}$.

Unlike the other corruption types, deletion corruption is defined analytically since the hash function is defined as a logistic function, which allows easier analysis of

parameter relationships. According to the definition of P_{Occur}, the parameter E_{Half} acts as an offset for the logistic curve, changing the overall sensitivity of the hash to stimulation. As E_{Half} increases, the stimulus power required to elicit a response increases, meaning that the hash is less likely to fire for some given stimulus S. The parameter k is a gain for the logistic curve. As k goes up, the occurrence probability curve becomes sharper with respect to E_{Half}. As can be seen in Figure 25, low hash occurrence regions lead to higher maximum tolerable P_{Hash}. The energy of the stimulus being applied shifts the occurrence probability, and therefore P_{Hash}, while maintaining the same general shape. Higher energy stimuli yield higher occurrence probabilities for all hash parameterizations (except where $k = 0$), in effect shifting the surface of P_{Hash} to the right, as can be seen in Figure 25.A which represents the longer duration and higher energy stimulus.

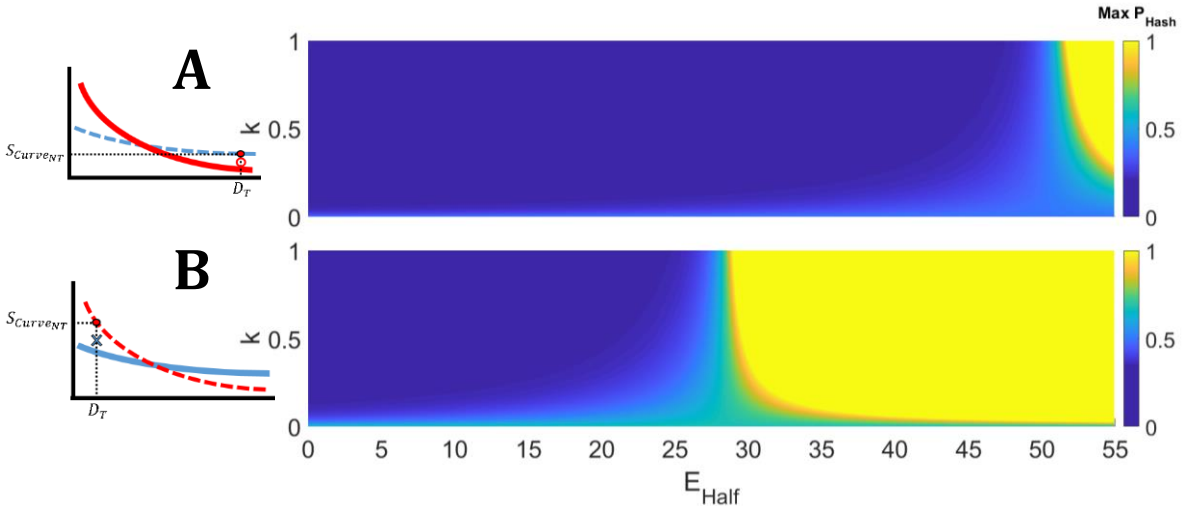

Figure 25: The maximum P_{Hash}, indicated by color, that can be tolerated by a target neuron, controlled with the standard bearer neuron, at a given GT, with different parameterizations of the hash

Panel A is shown for corrupted red target neuron $r_{\theta_T} = [.67, .83, 1]$ and $GT = [3.5, 14.2]$ (on the SD curve of a "faster" standard bearer neuron). Panel B is shown for a corrupted blue target neuron $r_{\theta_T} = [1.5, 1.25, 1]$ and $GT = [26.6, 1.1]$ (on the SD curve of a "slower" standard bearer neuron). The white areas indicate hash parameterizations that never cause the target corrupted neuron to appear to fire at lower than 50% firing probability at the tested GT, and therefore can tolerate any value of P_{Hash} because the corrupted neuron always appears controllable with the standard bearer. The colored area indicates hash parameterizations that may cause the target corrupted neuron to appear to fire at lower than 50% probability, and can therefore tolerate only the indicated amount of P_{Hash} without appearing to lose controllability.

3.4 Discussion

3.4.1 The Use of Spike Sorting in Neurocontrol

Clinical neurocontrol systems are currently used for administering artificial percepts, such as in cochlear implants (B. S. Wilson & Dorman, 2008; Zeng, 2017). If a patient needs a neuro-prosthetic of some kind, the range of activity that prosthetic is able to induce informs the design and parameters of the sensory feedback system. For the cochlear implant, the patient's cochlea is stimulated at a variety of spatial locations and

patterns, and the patient's sensory response to these stimuli is recorded and used to program the speech processor. In the future, populations of neurons may be stimulated and characterized through direct neural stimulation and recording, and a mapping can be developed between stimuli and their intended results. In both of these cases, the exact identities of the neurons being stimulated have clinical importance. However, in the direct stimulation and recording case, corruption in the feedback signal to the controller may have direct implications about its performance. Because of this, the ability to diagnose and reduce the effects of corruption may have application to future clinical devices, and could be valuable for ensuring high quality control.

One possible solution to the problem of corruption during spike sorting is to simply not use spike sorting. Previous work has shown that motor brain machine interfaces that include no spike sorting can provide decoding with comparable performance to systems that use spike sorting (Christie et al., 2015; Fraser et al., 2009). It is worth noting, however, that use of a system without spike sorting has different implications when used as part of a stimulating control system. As mentioned previously, the neurons of interest in a sensory control system will likely have been identified as important for the induction of artificial percepts during the system characterization step. This is different from the motor control case due to the fact that, while individual neurons carry significant information about motor intention, so too does the population activity. However, it may not be the case that general population activity, as would be achieved with a sensory control system without spike sorting, would be valuable for inducing percepts. Individual neuron control may or may not be required for

an effective neurocontrol interface. If it is required, however, then cluster cutting likely will be required for the implementation of that interface, and spike sorting corruption may be a central challenge.

3.4.2 Detecting and Compensating for Corruption

The models above show an idealistic view on the effects of corruption, but they offer insights into how these general types of corruption might be identified in electrophysiology recordings, as well as how technicians might compensate for them. Corruption may be difficult to identify, particularly in systems where spike sorting is done automatically (Hill et al., 2011), such as those with large electrode counts. Without human intervention, it is often difficult to differentiate between two similar spike waveforms that belong to different sources. Even with humans in the loop, such a problem is still difficult to solve.

The first insight that these models provide is identifying systems that may be vulnerable to corruption, and how that corruption may present itself. For example, to make any neuron in a pair more tolerant to apparent loss of control via exclusion corruption, the neuron's firing probability should be increased, which means β should be increased and/or α should be decreased. However, this is not viable as a strategy for selecting neuron parameterizations for robust control. The tolerance of the pair to exclusion corruption, as demonstrated by the maximum tolerable corruption (Figure 23), is expressed as a ratio between the α's and β's of each neuron. Therefore, one cannot adjust both neurons' in the same way and achieve higher tolerance against corruption.

This implies a tradeoff between the [α, β] parameterization of each neuron; both neurons cannot robustly tolerate exclusion corruption simultaneously.

In a clinical application, the technician would not be able to select the parameterizations of each neuron. Therefore, the value in this framework is that it provides intuition about how tolerant a system is to corruption given its parameterization. For example, if the β values of both neurons are similar, then this framework implies that the fast neuron (the neuron whose β is larger) is vulnerable to exclusion corruption, due to the fact that a small decrease in its observed firing probability could cause a qualitative shift in apparent controllability.

Tracking the responses of a unit of interest to various stimuli across SD space may also allow the controller to build a model for the neuron. For example, a unit may be fit to an IAF model, but as the experiment goes on, future stimuli may eventually have a low model likelihood. This can happen for a number of reasons. The first may be that the neuron's stimulation response has changed, meaning that the IAF parameters must be updated. However, other behaviors may show that no IAF implementation would explain the observations, which means that corruption may be playing a role.

Corrupted clusters may behave significantly differently from uncorrupted units, and this behavior may present itself in a number of ways. For example, if no stimulus, regardless of power, is able to cause the neuron to fire with probability above 50%, then it is possible that the cluster is being subject to exclusion corruption. When some fraction of a target neuron's spikes is excluded, the controller will estimate that the neuron is significantly less sensitive than it actually is. To counteract, the controller will likely

increase the energy of stimulus, in an attempt to increase the firing rate of the target neuron. The controller might over drive the target neuron, particularly if it is attempting to induce a high firing probability. Countering this issue could require tracking the total number of threshold crossings (Christie et al., 2015; Fraser et al., 2009), and using this information to detect if the firing probability as a function of stimulation strength is approaching an asymptote below 1. If a relationship that asymptotes below 1 is detected in the cluster's firing probability, but the model firing probability as a function of threshold crossings reaches to 1, it may reveal the existence of excluded spikes, indicating that the clusters should be redefined.

An underutilized opportunity is that the spike sorter has access to more information when used in a control system than when used as a passive sorter, such as the properties of stimuli that trigger neural activity, or the ability to use optogenetics to tie waveforms to genetic identification (Lima & Miesenböck, 2005). Additionally, information about the quality of clustering can be fed to the controller, for example, in the form of an L-ratio. This information can be used to gauge the likeliness of corruption for each cluster.

The models may also offer insight into how corruption can be minimized once detected. One response is to try to improve the spike sorter. Different metrics could be employed, or different automated systems (Lewicki, 1998) such as automatic sorting using k-means clustering (Salganicoff et al., 1988) may increase the capability and autonomy of spike sorters. The primary challenge is that stimulation tends to simultaneously activate a region of tissue around the electrode, leading to hash. This

unpredictable, high amplitude background noise can make it difficult to isolate units of interest from the collision of waveforms. Information from the spike sorter can be used to modify the clusters this, or other forms of corruption. For example, the controller can use information on how restrictive the cluster is to gauge which type of corruption is more likely. This is because exclusion corruption and addition corruption are functionally opposites due to the fact that exclusion corruption tends to occur when generating overly restrictive clusters in an attempt to reduce the effects of addition corruption. Using this information, the spike sorter can decide to expand or contract a cluster to reduce corruption while conserving the signal from the neuron of interest.

Care must be taken while interpreting the implications of uncommon responses. The plateauing effect, for example, where stimulation is not able to increase a unit's observed firing probability above a certain level, and may lead to decreases of firing probability with stimulus power increases, may be the result of corruption. However, it may also be the result of normal physiology, where the unit of interest is part of an inhibitory network that is sensitive to the stimulus (due to high opsin expression, or proximity to an electrode). Only information about the region of interest, context, and goal of the control system can be used to make a final decision about whether an observation is physiological or corruptive in nature.

4 DISCUSSION

4.1 Limitations of the Current Study

In this dissertation, I present a demonstration of 1:2 underactuated control, in which one input (laser stimulation) controls two outputs (neurons) simultaneously. Additionally, I explore various obstacles to accurate identification of the system, as well as their effects on estimates of the system's state and ability to be controlled.

The stimulation and modeling had a number of limitations, a primary example being the simplicity of the neural model, the IAF neuron. While the model is convenient because it has few parameters and is analytically tractable, more complex neural models could be used to capture more sophisticated dynamics and in turn suggest more strategies for control. The stimuli used throughout this study were rectangular pulses, parameterized only by their strengths and durations $[G, T]$, though this limitation is a direct result of the time optimality of bang control on linear systems such as the IAF model (Nabi & Moehlis, 2012). Removing this limitation may significantly increase the controllability of neuron populations. However, when leaving the IAF model, requirements for control become less clear, and model fitting more challenging due to the increased number of parameters. While it is possible to produce control inputs for other models, such as Hodgkin-Huxley (HH) neurons (Ullah & Schiff, 2009), it is likely that novel approaches like machine learning will be required to develop robust and general control strategies (Liu et al., 2018; Narayanan et al., 2019).

The 1:2 control presented in this work is likely near the upper limit of control ratios for which this control schema is feasible. At higher ratios it could be difficult to

find neurons that jointly satisfy the prerequisite control conditions. For any N deterministic neurons to be pairwise controllable, their parameters must satisfy the IAF relation (Ching & Ritt, 2013)

$$\alpha_1 > \alpha_2 > \cdots > \alpha_N$$

$$\beta_1 > \beta_2 > \cdots > \beta_N$$

$$\frac{\alpha_1}{\beta_2} > \frac{\alpha_2}{\beta_2} > \cdots > \frac{\alpha_N}{\beta_N}$$

$$\frac{\alpha_1 - \alpha_0}{\beta_1 - \beta_0} > \frac{\alpha_2 - \alpha_1}{\beta_2 - \beta_1} > \cdots > \frac{\alpha_N - \alpha_{N-1}}{\beta_N - \beta_{N-1}}$$

An example of the SD curves of a simulated triplet of neurons that satisfy these relations can be seen in Figure 26. When considering the logistical difficulties of performing this kind of control in vivo, including biasing towards neurons that have high sensitivity and a low spontaneous firing rate (large β and a large $\frac{\alpha}{\sigma}$ ratio), and that an average of about 1.2 candidate pairs of neurons were found per mouse (with only about a third of these candidates satisfying control), the probability of finding neurons that satisfy the requirements for $N > 2$ appears small. It is worth noting, however, that lifting some of the limitations of this study, such as by using nonrectangular stimuli, may make it easier to perform control at higher ratios of neurons to inputs.

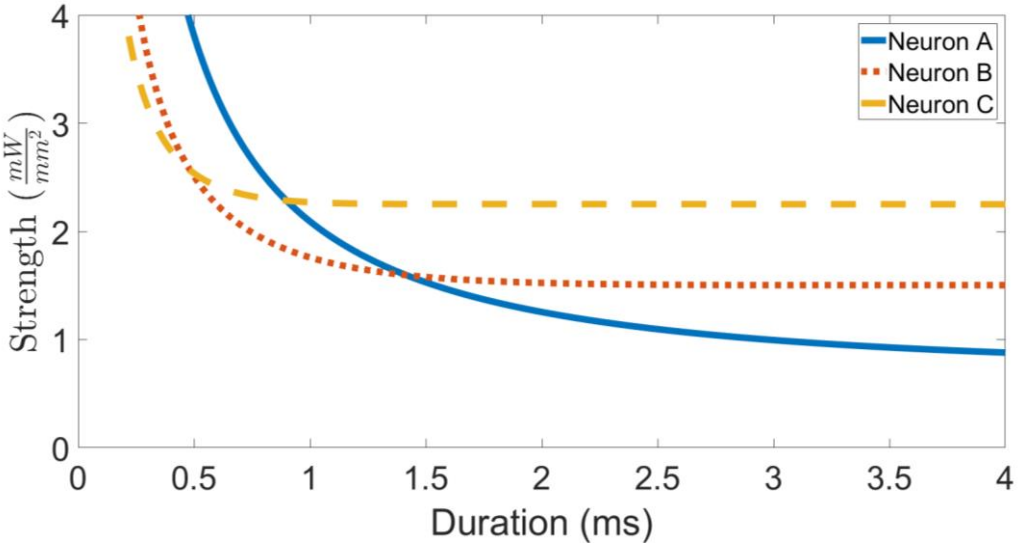

Figure 26: A set of three simulated neurons that are mutual controllable

While such control is theoretically possible, limitations on instrumentation and physiology make it unlikely that such configurations will be feasible for control in clinical settings.

Lower control ratios may also be possible, by increasing the number of inputs. For example, a 2:3 system may be easier to control than a 1:2 system due to a smaller demand placed on each input. I expect that controllability will become significantly better with each additional input added to the system, due to exploitable interactions between the inputs. These systems may take a number of different forms. Spatial distributions may be used, such that the two inputs each activate unique neural populations on their own, and are able to activate a third when the inputs are in unison (Figure 27.A). Bidirectional inputs may be used by expressing both an excitatory and an inhibitory rhodopsin in the neural population (Figure 27.B). The cells may be stimulated with two wavelengths of light, one each for excitation and inhibition, which can be used in unison to address individual cells in the population. The two approaches might be used simultaneously to combinatorially increase the dimensionality of the input.

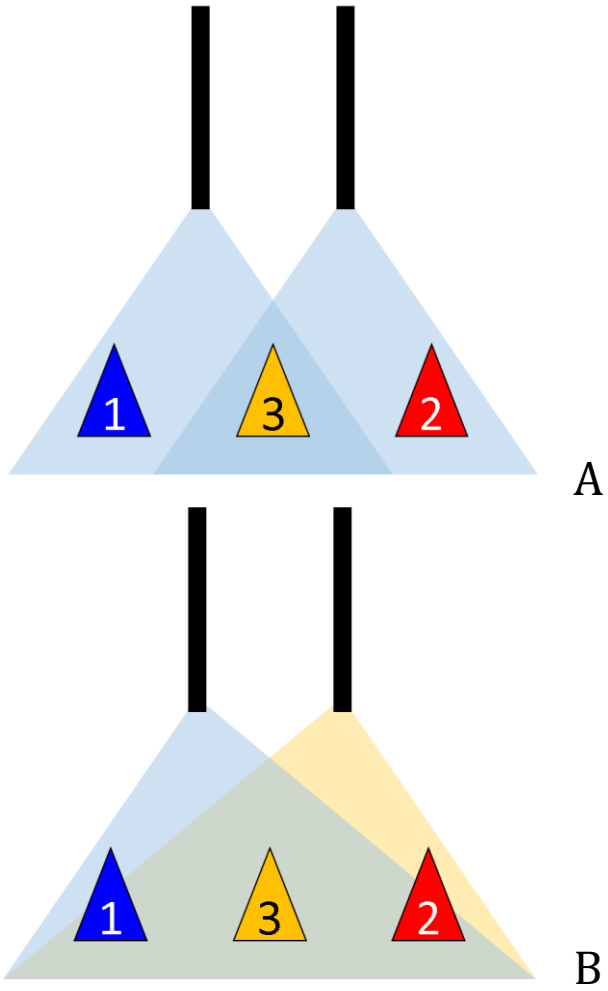

Figure 27: A system which uses spatial encoding to address multiple cells in an underactuated system

In the setup shown in panel A, each cell expresses ChR2 (or some other excitatory rhodopsin), and spatial differences allow each input to activate different neural populations individually, or a third when both are used in unison. In the setup shown in panel B, each cell expresses both an excitatory rhodopsin (such as ChR2, activated by blue light) and an inhibitory rhodopsin (such as NpHR, activated by yellow light). The two inputs activate one rhodopsin each.

Success with any level of underactuated control, whether it is the 1:2 control demonstrated here or a future lower ratio control schema, has relevance for high density electrode arrays. Even small increases in the dimensionality of control on a large array may significantly increase the number of independently controllable neurons.

4.2 Implications for Clinical Neuro-control

It is unlikely that wearable neuro-stimulation hardware will allow 1:1 or few-to-one ratios between stimulation electrodes and target neurons for the foreseeable future, so control techniques that leverage the full available control space may have clinical relevance for some time to come. Other methods to increase stimulator dimensionality, such as using current steering techniques to aid in deep brain stimulation (Barbe et al., 2014; Butson & McIntyre, 2008) or cochlear implants (Firszt et al., 2007), have been studied for some time. Underactuated control methods are yet another technique to take advantage of the full control space, and may lead to further developments for artificial percepts and other neuro-stimulation applications.

APPENDIX

Code to reproduce the data analysis and figures found in this dissertation can be found in the following GitHub repository.

https://github.com/samuelgbrown/Acute_Control

BIBLIOGRAPHY

Ahmadian, Y., Packer, A. M., Yuste, R., & Paninski, L. (2011). Designing optimal stimuli to control neuronal spike timing. *Journal of Neurophysiology*, *106*(2), 1038–1053. https://doi.org/10.1152/jn.00427.2010

Barbe, M. T., Maarouf, M., Alesch, F., & Timmermann, L. (2014). Multiple source current steering - A novel deep brain stimulation concept for customized programming in a Parkinson's disease patient. *Parkinsonism & Related Disorders*, *20*(4), P471–473. https://doi.org/10.1016/j.parkreldis.2013.07.021

Bolus, M. F., Willats, A. A., Whitmire, C. J., Rozell, C. J., & Stanley, G. B. (2018). Design strategies for dynamic closed-loop optogenetic neurocontrol in vivo. *Journal of Neural Engineering*, *15*(2), 026011. https://doi.org/10.1088/1741-2552/aaa506

Boyden, E. S., Zhang, F., Bamberg, E., Nagel, G., & Deisseroth, K. (2005). Millisecond-timescale, genetically targeted optical control of neural activity. *Nature Neuroscience*, *8*, 1263–1268. https://doi.org/10.1038/nn1525

Burridge, J. H., & Ladouceur, M. (2001). Clinical and Therapeutic Applications of Neuromuscular Stimulation: A Review of Current Use and Speculation into Future Developments. *Neuromodulation: Technology at the Neural Interface*, *4*(4), 147–154. https://doi.org/10.1046/j.1525-1403.2001.00147.x

Butson, C. R., & McIntyre, C. C. (2008). Current steering to control the volume of tissue activated during deep brain stimulation. *Brain Stimulation*, *1*(1), 7–15. https://doi.org/10.1016/j.brs.2007.08.004

Byrd, R. H., Gilbert, J. C., & Nocedal, J. (2000). A trust region method based on interior point techniques for nonlinear programming. *Mathematical Programming*, *Series B*, *89*(1), 149–185. https://doi.org/10.1007/PL00011391

Byrd, R. H., Hribar, M. E., & Nocedal, J. (1997). An Interior Point Algorithm for Large Scale Nonlinear Programming. *SIAM Journal on Optimization*, *9*(4), 877–900. https://doi.org/10.1137/S1052623497325107

Ching, S. N., & Ritt, J. T. (2013). Control strategies for underactuated neural ensembles driven by optogenetic stimulation. *Frontiers in Neural Circuits*, *7*(Mar), 54. https://doi.org/10.3389/fncir.2013.00054

Christie, B. P., Tat, D. M., Irwin, Z. T., Gilja, V., Nuyujukian, P., Foster, J. D., Ryu, S. I., Shenoy, K. V, Thompson, D. E., & Chestek, C. A. (2015). Comparison of spike sorting and thresholding of voltage waveforms for intracortical brain–machine interface performance. *Journal of Neural Engineering*, *12*(1), 016009. https://doi.org/10.1088/1741-2560/12/1/016009

Dayan, P., & Abbott, L. (2001). *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. The MIT Press: Cambridge, Mass. and London, England.

Dorato, P., Hsieh, C. M., & Robinson, P. N. (1967). Optimal Bang-Bang Control of Linear Stochastic Systems with a Small Noise Parameter. *IEEE Transactions on Automatic Control*, *AC-12*(6), 682–690. https://doi.org/10.1109/TAC.1967.1098731

Eversmann, B., Lambacher, A., Gerling, T., Kunze, A., Fromherz, P., & Thewes, R. (2011). A neural tissue interfacing chip for in-vitro applications with 32k recording / stimulation channels on an active area of 2.6 mm^2. *European Solid-State Circuits Conference*, 211–214. https://doi.org/10.1109/ESSCIRC.2011.6044902

Firszt, J. B., Koch, D. B., Downing, M., & Litvak, L. (2007). Current Steering Creates Additional Pitch Percepts in Adult Cochlear Implant Recipients. *Otology & Neurotology*, *28*(5), 629–636. https://doi.org/10.1097/01.mao.0000281803.36574.bc

Fraser, G. W., Chase, S. M., Whitford, A., & Schwartz, A. B. (2009). Control of a brain-computer interface without spike sorting. *Journal of Neural Engineering*, *6*(5), 055004. https://doi.org/10.1088/1741-2560/6/5/055004

Frey, U., Sedivy, J., Heer, F., Pedron, R., Ballini, M., Mueller, J., Bakkum, D., Hafizovic, S., Faraci, F. D., Greve, F., Kirstein, K. U., & Hierlemann, A. (2010). Switch-matrix-based high-density microelectrode array in CMOS technology. *IEEE Journal of Solid-State Circuits*, *45*(2), 467–482. https://doi.org/10.1109/JSSC.2009.2035196

Grün, S., Riehle, A., & Diesmann, M. (2003). Effect of cross-trial nonstationarity on joint-spike events. *Biological Cybernetics*, *88*(5), 335–351. https://doi.org/10.1007/s00422-002-0386-2

Hatsopoulos, N. G., & Donoghue, J. P. (2009). The Science of Neural Interface Systems. *Annual Review of Neuroscience*, *32*(1), 249–266. https://doi.org/10.1146/annurev.neuro.051508.135241

Hill, D. N., Mehta, S. B., & Kleinfeld, D. (2011). Quality metrics to accompany spike sorting of extracellular signals. *Journal of Neuroscience*, *31*(24), 8699–8705. https://doi.org/10.1523/JNEUROSCI.0971-11.2011

Huang, S. (2019). *Controllability Analysis and Design for Underactuated Stochastic Neurocontrol* [Doctoral dissertation – Boston University]. https://open.bu.edu/handle/2144/34932

Hyndman, R. J. (1996). Computing and Graphing Highest Density Regions. *American Statistician*, *50*(2), 120–126. https://doi.org/10.1080/00031305.1996.10474359

Iolov, A., Ditlevsen, S., & Longtin, A. (2014). Stochastic optimal control of single neuron spike trains. *Journal of Neural Engineering*, *11*(4), 046004. https://doi.org/10.1088/1741-2560/11/4/046004

Iolov, A., Ditlevsen, S., & Longtin, A. (2017). Optimal Design for Estimation in Diffusion Processes from First Hitting Times. *SIAM/ASA Journal on Uncertainty Quantification*, *5*(1), 88–110. https://doi.org/10.1137/16M1060376

Jun, J. J., Steinmetz, N. A., Siegle, J. H., Denman, D. J., Bauza, M., Barbarits, B., Lee, A. K., Anastassiou, C. A., Andrei, A., Aydin, Ç., Barbic, M., Blanche, T. J., Bonin, V., Couto, J., Dutta, B., Gratiy, S. L., Gutnisky, D. A., Häusser, M., Karsh, B., … Harris, T. D. (2017). Fully integrated silicon probes for high-density recording of neural activity. *Nature*, *551*(7679), 232–236. https://doi.org/10.1038/nature24636

Kalman, R. E. (1959). On the general theory of control systems. *IRE Transactions on Automatic Control*, *4*(3), 110. https://doi.org/10.1109/TAC.1959.1104873

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*(4). https://doi.org/10.1088/0954-898X_9_4_001

Li, J.-S., Dasanayake, I., & Ruths, J. (2012). Control and Synchronization of Neuron Ensembles. *IEEE Transactions on Automatic Control 58*(8), 1919–1930.

Lima, S. Q., & Miesenböck, G. (2005). Remote control of behavior through genetically targeted photostimulation of neurons. *Cell*, *121*(1), 141–152. https://doi.org/10.1016/j.cell.2005.02.004

Liu, S., Sock, N. M., & Ching, S. (2018). Learning-based Approaches for Controlling Neural Spiking. *Proceedings of the American Control Conference*, *2018-June*, 2827–2832. https://doi.org/10.23919/ACC.2018.8431158

Madisen, L., Mao, T., Koch, H., Zhuo, J. M., Berenyi, A., Fujisawa, S., Hsu, Y. W. A., Garcia, A. J., Gu, X., Zanella, S., Kidney, J., Gu, H., Mao, Y., Hooks, B. M., Boyden, E. S., Buzsáki, G., Ramirez, J. M., Jones, A. R., Svoboda, K., … Zeng, H. (2012). A toolbox of Cre-dependent optogenetic transgenic mice for light-induced activation and silencing. *Nature Neuroscience*, *15*(5), 793–802. https://doi.org/10.1038/nn.3078

Meng, L., Kramer, M. A., & Eden, U. T. (2011). A sequential Monte Carlo approach to estimate biophysical neural models from spikes. *Journal of Neural Engineering*, *8*(6), 065006. https://doi.org/10.1088/1741-2560/8/6/065006

Mitchell, B. A., & Petzold, L. R. (2018). Control of neural systems at multiple scales using model-free, deep reinforcement learning. *Scientific Reports*, *8*(1), 1–12.

https://doi.org/10.1038/s41598-018-29134-x

Nabi, A., & Moehlis, J. (2011). Single input optimal control for globally coupled neuron networks. *Journal of Neural Engineering*, *8*(6), 065008. https://doi.org/10.1088/1741-2560/8/6/065008

Nabi, A., & Moehlis, J. (2012). Time optimal control of spiking neurons. *Journal of Mathematical Biology*, *64*(6), 981–1004. https://doi.org/10.1007/s00285-011-0441-5

Nadarajah, S., & Kotz, S. (2008). Exact Distribution of the Max/Min of Two Gaussian Random Variables. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *16*(2). https://doi.org/10.1109/TVLSI.2007.912191

Nandi, A., Schättler, H., Ritt, J. T., & Ching, S. N. (2017). Fundamental Limits of Forced Asynchronous Spiking with Integrate and Fire Dynamics. *Journal of Mathematical Neuroscience*, *7*(1), 11. https://doi.org/10.1186/s13408-017-0053-5

Narayanan, V., Ritt, J. T., Li, J. S., & Ching, S. (2019). A learning framework for controlling spiking neural networks. *Proceedings of the American Control Conference*, *2019-July*, 211–216. https://doi.org/10.23919/acc.2019.8815197

Newman, J. P., Fong, M. F., Millard, D. C., Whitmire, C. J., Stanley, G. B., & Potter, S. M. (2015). Optogenetic feedback control of neural activity. *ELife*, *4*(July 2015), 1–24. https://doi.org/10.7554/eLife.07192

Ostrowsky, K., Magnin, M., Ryvlin, P., Isnard, J., Guenot, M., & Mauguière, F. (2002). Representation of Pain and Somatic Sensation in the Human Insula: a Study of Responses to Direct Electrical Cortical Stimulation. *Cerebral Cortex*, *12*(4), 376–385. https://doi.org/10.1093/cercor/12.4.376

Palanker, D., Vankov, A., Huie, P., & Baccus, S. (2005). Design of a high-resolution optoelectronic retinal prosthesis. *Journal of Neural Engineering*, *2*(1), S105–S120. https://doi.org/10.1088/1741-2560/2/1/012

Rickgauer, J. P., Deisseroth, K., & Tank, D. W. (2014). Simultaneous cellular-resolution optical perturbation and imaging of place cell firing fields. *Nature Neuroscience*, *17*(12), 1816–1824. https://doi.org/10.1038/nn.3866

Rosin, B., Slovik, M., Mitelman, R., Rivlin-Etzion, M., Haber, S. N., Israel, Z., Vaadia, E., & Bergman, H. (2011). Closed-loop deep brain stimulation is superior in ameliorating parkinsonism. *Neuron*, *72*(2), 370–384. https://doi.org/10.1016/j.neuron.2011.08.023

Sakmann, B., & Neher, E. (1984). Patch clamp techniques for studying ionic channels in excitable membranes. *Annual Review of Physiology*, *46*, 455–472.

https://doi.org/10.1146/annurev.ph.46.030184.002323

Salganicoff, M., Sarna, M., Sax, L., & Gerstein, G. L. (1988). Unsupervised waveform classification for multi-neuron recordings: a real-time, software-based system. I. Algorithms and implementation. *Journal of Neuroscience Methods*, *25*(3), 181–187. https://doi.org/10.1016/0165-0270(88)90132-X

Santaniello, S., Fiengo, G., Glielmo, L., & Grill, W. M. (2011). Closed-loop control of deep brain stimulation: A simulation study. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *19*(1), 15–24. https://doi.org/10.1109/TNSRE.2010.2081377

Schmidt, E. M., Bak, M. J., Hambrecht, F. T., Kufta, C. V, O 'rourke, D. K., & Vallabhanath, P. (1996). Feasibility of a visual prosthesis for the blind based on intracorticai microstimulation of the visual cortex. *Brain*, *119*, 507–522. https://academic.oup.com/brain/article-abstract/119/2/507/382434

Ullah, G., & Schiff, S. J. (2009). Tracking and control of neuronal Hodgkin-Huxley dynamics. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *79*(4), 040901. https://doi.org/10.1103/PhysRevE.79.040901

Vidal, G. W. V., Rynes, M. L., Kelliher, Z., & Goodwin, S. J. (2016). Review of Brain-Machine Interfaces Used in Neural Prosthetics with New Perspective on Somatosensory Feedback through Method of Signal Breakdown. *Scientifica, 2016*, 8956432. https://doi.org/10.1155/2016/8956432

Wang, J., He, T., & Lee, C. (2019). Development of neural interfaces and energy harvesters towards self-powered implantable systems for healthcare monitoring and rehabilitation purposes. *Nano Energy 65*, 104039. https://doi.org/10.1016/j.nanoen.2019.104039

Wilson, B. S., & Dorman, M. F. (2008). Cochlear implants: A remarkable past and a brilliant future. *Hearing Research*, *242*(1–2), 3–21. https://doi.org/10.1016/J.HEARES.2008.06.005

Wilson, D., Holt, A. B., Netoff, T. I., & Moehlis, J. (2015). Optimal entrainment of heterogeneous noisy neurons. *Frontiers in Neuroscience*, *9*(May), 192. https://doi.org/10.3389/fnins.2015.00192

Wolff, S. B., & Ölveczky, B. P. (2018). The promise and perils of causal circuit manipulations. *Current Opinion in Neurobiology*, *49*, 84–94. https://doi.org/10.1016/j.conb.2018.01.004

Yang, Y., Connolly, A. T., & Shanechi, M. M. (2018). A control-theoretic system identification framework and a real-time closed-loop clinical simulation testbed for

electrical brain stimulation. *Journal of Neural Engineering, 15,* 066007. https://doi.org/10.1088/1741-2552/aad1a8

Zeng, F. G. (2017). Challenges in improving cochlear implant performance and accessibility. *IEEE Transactions on Biomedical Engineering*, *64*(8), 1662–1664. https://doi.org/10.1109/TBME.2017.2718939

CURRICULUM VITAE