2013

# Stochastic modeling and control of neural and small length scale dynamical systems

Gautam Kumar
*Lehigh University*

# STOCHASTIC MODELING AND CONTROL OF NEURAL AND SMALL LENGTH SCALE DYNAMICAL SYSTEMS

by

Gautam Kumar

A Dissertation Presented to the Graduate and Research Committee

of Lehigh University

in Candidacy for the Degree of

Doctor of Philosophy

in

Chemical Engineering

Lehigh University

Bethlehem, PA

September 2013

Approved and recommended for acceptance as a dissertation in partial fulfillment of requirements for the degree of Doctor of Philosophy.

_____
Date

_____
Dr. Mayuresh V. Kothare
(Dissertation Advisor)

_____
Accepted Date

Committee members:

_____
Dr. Joseph E. Yukich

_____
Dr. Manoj K. Chaudhury

_____
Dr. Piotr J. Franaszczuk

_____
Dr. William E. Schiesser

**To my teachers,**

**my friends,**

**and my family**

# Acknowlegements

I would like to express my sincere gratitude to my advisor Prof. Mayuresh V. Kothare for his continuous guidance, motivation, patience, and support throughout the course of my Ph.D. study and research. I must thank him for the freedom he allowed me during my research which helped me cultivate the qualities of an independent researcher. In my mind, I have no doubt that his thoughts have greatly influenced my long term life. It is an honor for me to obtain my doctoral training under him.

I am greatly thankful to the members of my committee Prof. Joseph E. Yukich, and Prof. William E. Schiesser for their help and feedback on various problems I studied as the part of this dissertation. I would like to express my thanks to the other members of my committee Prof. Manoj K. Chaudhury, and Prof. Piotr J. Franaszczuk for their invaluable advise at various levels during the course of my study.

I would like to acknowledge my great piano teacher Prof. Helen Beedle. Her motivation and piano lessons helped me in endeavoring through difficult circumstances and in rethinking my research problems with a new mind. I would like to acknowledge my amazing colleagues and friends Jonathan Longley, Reza Arastoo, and Abhishek Mishra for their valuable time which they spent with me on discussing various problems studied in this dissertation.

I would like to thank Michael Chupa for providing assistance with high performance computing. I greatly acknowledge financial support from the Department of Chemical Engineering, General Dynamics Land Systems, and National Science Foundation.

# Contents

# List of Tables

# List of Figures

# Abstract

Recent advancements in experimental and computational techniques have created tremendous opportunities in the study of fundamental questions of science and engineering by taking the approach of stochastic modeling and control of dynamical systems. Examples include but are not limited to neural coding and emergence of behaviors in biological networks. Integrating optimal control strategies with stochastic dynamical models has ignited the development of new technologies in many emerging applications. In this direction, particular examples are brain-machine interfaces (BMIs), and systems to manipulate submicroscopic objects. The focus of this dissertation is to advance these technologies by developing optimal control strategies under various feedback scenarios and system uncertainties.

Brain-machine interfaces (BMIs) establish direct communications between living brain tissue and external devices such as an artificial arm. By sensing and interpreting neuronal activity to actuate an external device, BMI-based neuroprostheses hold great promise in rehabilitating motor disabled subjects such as amputees. However, lack of the incorporation of sensory feedback, such as proprioception and tactile information, from the artificial arm back to the brain has greatly limited the widespread clinical deployment of these neuroprosthetic systems in rehabilitation. In the first part of the dissertation, we develop a systematic control-theoretic approach for a system-level

rigorous analysis of BMIs under various feedback scenarios. The approach involves quantitative and qualitative analysis of single neuron and network models to the design of missing sensory feedback pathways in BMIs using optimal feedback control theory. As a part of our results, we show that the recovery of the natural performance of motor tasks in BMIs can be achieved by designing artificial sensory feedbacks in the proposed optimal control framework.

The second part of the dissertation deals with developing stochastic optimal control strategies using limited feedback information for applications in neural and small length scale dynamical systems. The stochastic nature of these systems coupled with the limited feedback information has greatly restricted the direct applicability of existing control strategies in stabilizing these systems. Moreover, it has recently been recognized that the development of advanced control algorithms is essential to facilitate applications in these systems. We propose a novel broadcast stochastic optimal control strategy in a receding horizon framework to overcome existing limitations of traditional control designs. We apply this strategy to stabilize multi-agent systems and Brownian ensembles. As a part of our results, we show the optimal trapping of an ensemble of particles driven by Brownian motion in a minimum trapping region using the proposed framework.

# Chapter 1

# Introduction

The approach of stochastic modeling in dynamical systems allows one to distinguish internal fluctuations of the system from the external ones and to study their effects on the system behavior. This approach has become essential to understand dynamical behaviors of many biological and physical systems in their natural noisy environment. Examples include but are not limited to neurons and their networks [161, 10], biological processes [7], and micro and nano scale dynamical systems [9]. Integration of control theory with stochastic dynamical models has recently offered many emerging applications in biological and physical systems. In this direction, particular examples are an artificial arm controlled by the brain activity of an amputee [117], design of smart robots [25], biological processes on a chip [58], and drug delivery systems to cure diseases efficiently [130]. The key challenges in facilitating these applications are the stochastic nature of the system coupled with the limited feedback information in designing appropriate control actions. The focus of this dissertation is to develop advanced optimal control strategies to facilitate emerging applications in biological and physical systems. Particular systems studied are brain-machine interfaces (BMIs)

and small length scale dynamical systems.

## 1.1 Brain-Machine Interfaces

Brain-machine interfaces (BMIs) [100, 117, 118] are broadly defined as systems that establish direct communications between living brain tissue and external devices such as artificial arm. The major components of these systems include measurements of cortical neuronal activity, extraction of task-relevant motor intention (decoder), and an encoder that feeds back the motor relevant sensory information back to the brain. Thus the brain, the BMI and the prosthetic device together act as a closed-loop BMI. Fig. 1.1 shows a closed-loop BMI (also known as brain-machine-brain interface (BMBI) [38]) structure.

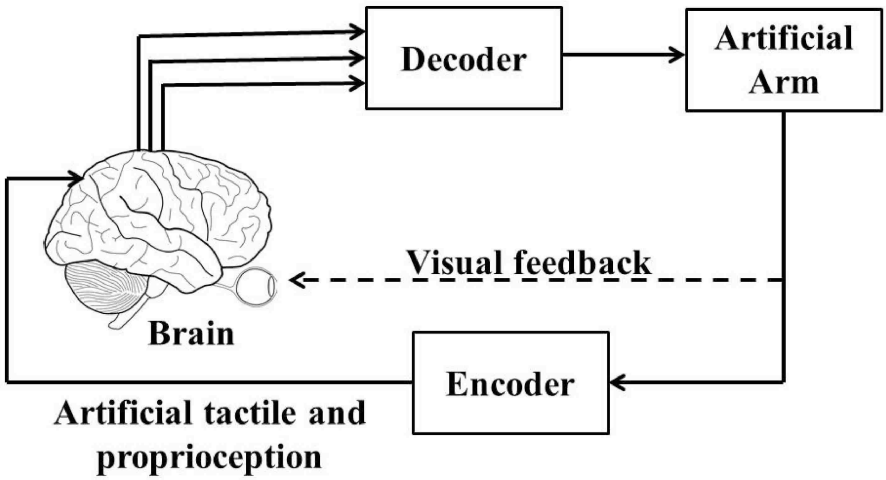



Figure 1.1: A closed-loop brain-machine interface (BMI). Brain shot courtesy of wikipedia.

By sensing and interpreting neuronal activities to actuate an external device, BMI-based neuroprostheses hold great promise in rehabilitating motor disabled subjects

such as amputees. Design and implementation of BMIs are beneficial to human subjects suffering from peripheral neuropathies, Parkinsons, and amputations etc. The impact of such remedies is substantial in providing relief and enabling motor impaired individuals who are otherwise able citizens to lead productive lives which provide motivation for developing such systems.

BMIs are classified into two categories, namely non-invasive [170, 23] and invasive, based on methods of electrophysiological measurements of neuronal firing activity of the brain [98]. Non-invasive BMIs are based on electroencephalograms (EEGs) methods of capturing neuronal firing activity and captures an average behavior with limited capacity. On the other hand, invasive BMIs uses ensemble recordings of multiple neurons associated with a particular activity by implantation of multiple physical micro-electrodes. In this dissertation, we only consider invasive BMIs. Therefore from here onwards, we omit the term invasive and we always mean that BMIs are invasive whenever they are mentioned.

### 1.1.1 State of the art

The scientific origins of controlling external devices using the brain activity go back at least to the initial electrophysiological experiments in late 1960s from primary motor cortex (M1) of awake, behaving non-human primates. In 1968, Evarts showed in a classic experiment that the firing of M1 neurons in monkeys were strongly correlated with the amount of force generated by the joints during the arm movements [44]. An experiment performed by Fetz in 1969 showed that monkeys could modulate and control the activity of M1 neurons using visual feedback and reward based cortical learning [47]. In 1970, Humphrey et. al. showed using simple quantitative procedures

that a small sets of M1 neurons are adequate to predict the accurate time course of kinematic and kinetic measurements during simple arm movements [75]. In mid-1970s, several studies by Fetz and colleagues demonstrated the existence of a linear correlation between the firing rate of M1 neurons and the force generated by the joints during the arm movements [48, 49, 87]. These early studies demonstrated (1) the encoding of the kinetics of arm movement in the activity of M1 neurons, and (2) control of the activity of single M1 neurons using biofeedback and learning.

In 1980, Edvard Schmidt raised the possibility to control external devices by extracting voluntary motor commands from raw cortical neural activity of M1 neurons [144]. After two decades, in 1999, the first experiment in BMI research was performed in which the control of a robotic arm with a single degree of freedom was demonstrated using the recorded neural activity of a population of M1 neurons from behaving rats [24]. During 2000-2005, several laboratories performed experiments on rhesus monkeys and reported BMIs that reproduced trajectories of primate arm movements during reaching and grasping using either computer cursors or robotic manipulators as actuators [169, 146, 154, 22, 99, 52]. In 2006, the first human experiment was reported in which BMI was developed to control prosthetic devices (computer cursor and physical devices) using intracortical neural ensemble spiking activity in human with tetraplegia [70] (see [170] for noninvasive BMI in human). These early results suggested that BMIs could be used to restore motor impairments in humans with paralysis. Since then, BMI based motor intended neural prosthetic systems have been studied extensively towards their clinical deployment (see [176, 111, 11, 165, 66, 68, 30, 110, 38, 71, 43]). The major effort in most of these studies has been centered on enhancing the performance of BMIs by developing better decoding and learning

algorithms.

### 1.1.2 Need of a Closed-loop Control-theoretic Framework

In most of BMIs studies, healthy subjects (mostly non-human primates) are trained to perform a specific motor task such as reaching or grasping. Recorded data (brain activities as well as task relevant kinematic and kinetic information) during the performance of the task are then used to develop a mathematical model called decoder. The decoder extracts the kinetic as well as the kinematic motor information from a continuously recorded activity of motor relevant cortical neurons. The performance of the decoder is typically measured by applying the decoded information to a prosthetic arm. The online movement based error correction during the reaching task is accomplished by the subject using the available visual feedback information as well as sensorimotor learning in the absence of the natural proprioception. Therefore these BMIs are considered as partially closed-loop systems in their current formulations where the incorporation of artificial proprioception and texture information are neglected in their designs.

In the absence of tactile feedback, these BMIs can fail to differentiate visually similar textures. Similarly in the absence of proprioception, these BMIs are unable to provide the natural sensation of the arm movement which are both experienced and used by healthy subjects in controlling their natural limb movements. It has recently been recognized that inclusion of sensory feedback from the actuated artificial limb is necessary to improve the versatility of motor-based BMIs [141].

Very recently, attempts have been made towards closing the BMI loop by incorporating artificial texture [38] and proprioception [168] information. In these studies,

intra-cortical micro-stimulation (ICMS) technique has been investigated as a promising approach in providing artificial sensation of motor tasks to the brain. Even though the technique is promising for developing future BMIs, the experimental trial and error approach in designing appropriate stimulating sensory input currents may change the natural functionality of the brain. Therefore, a systematic approach that uses optimal feedback control theory is highly desirable towards developing stimulation enhanced next generation BMIs. This approach provides flexibility in designing optimal stimulating sensory input currents and analyzing the closed-loop BMI under various feedback scenarios.

The intellectual merit of studying such systems by taking control-theoretic approach is to exploit all the available degrees of freedom in developing the next generation of feedback-enabled neuroprosthetic devices. The development of these feedback-enabled neuroprosthetic devices is necessary for making the prosthetic devices less prone to error in decoding and targeting the intended action of the neurons. Negligible attention has been devoted to study these feedback-enabled neuroprosthetic devices and impact of natural and surrogate feedback paths on the overall functioning of neural prosthesis systems (see [14] as exception). For example, these feedback paths have never been considered in studies conducted on developing a robotic device, in primates for reproducing reaching and grasping movements and in motor impaired human subjects. At a broader level, a control-theoretic framework that considers the various interacting systems and feedback paths in a true multi-variable context from a systems-perspective is entirely missing from the literature. As a result, issues such as prosthetic system stability, system reliability, impact of transmission loss, latency and time delays, impact of model complexity and uncertainty, optimality of the modeling

and control framework, etc. which are critical to the success of this emerging discipline remain open problems. This provides the main motivation for control-theoretic system level analysis of feedback-enabled neuroprosthetic devices for successful transition of BMIs-based neural prostheses and assistive devices to stable extended use in human subjects.

Figure 1.2 shows our proposed design of a closed-loop BMI using an optimal receding horizon controller. As shown in this design, an optimal receding horizon controller replaces the encoding block shown in Figure 1.1 in designing artificial tactile and proprioception information.

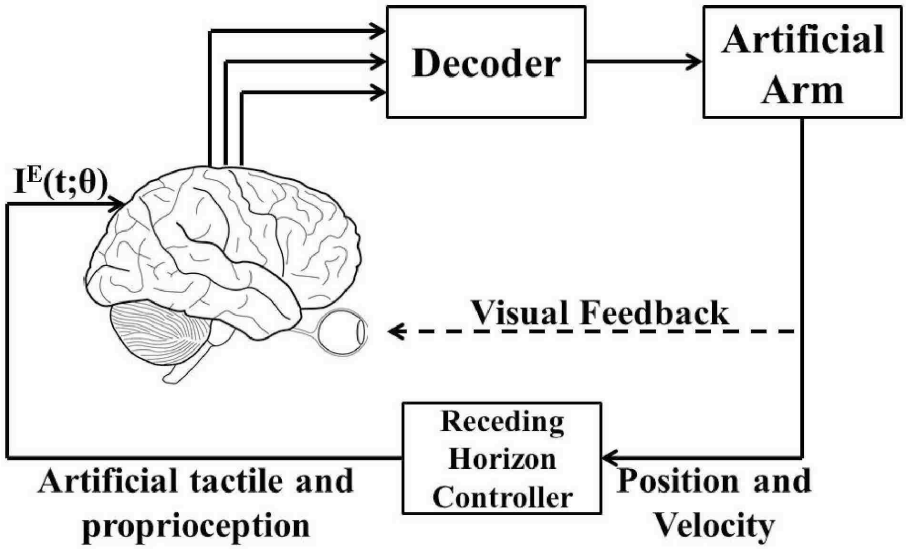



Figure 1.2: Receding horizon controller based closed-loop brain-machine interface (BMI). Brain shot courtesy of wikipedia. Here, the receding horizon controller designs optimal artificial sensory feedback currents $I^E(t;\boldsymbol{\theta})$ and stimulates neurons of the appropriate cortical sensory areas such that the closed-loop (natural) performance of the BMI can be recovered for a given motor task.

### 1.1.3 Model-based Receding Horizon Controller

As shown in Figure 1.2, the optimal artificial sensory feedback currents $I^E(t;\boldsymbol{\theta})$ are designed using a model based receding horizon controller (RHC). A model based RHC [95], also known as model predictive control (MPC), is an optimal control strategy that explicitly incorporates a dynamic model of the system as well as constraints in determining control actions. At each time $k$, the system measurements are obtained and a model of the system is used to predict future outputs of the system $y_{k+l+1|k}$, $l = 0, 1, 2, \cdots, N_p - 1$ as a function of current and future control moves $u_{k+l|k}$, $l = 0, 1, 2, \cdots, N_c - 1$. How far ahead in the future the predictions are computed is called the prediction horizon $N_p$ and how far ahead the control moves are computed is called the control horizon $N_c$. Figure 1.3 illustrates the idea of prediction and control horizon in a model based receding horizon control strategy.

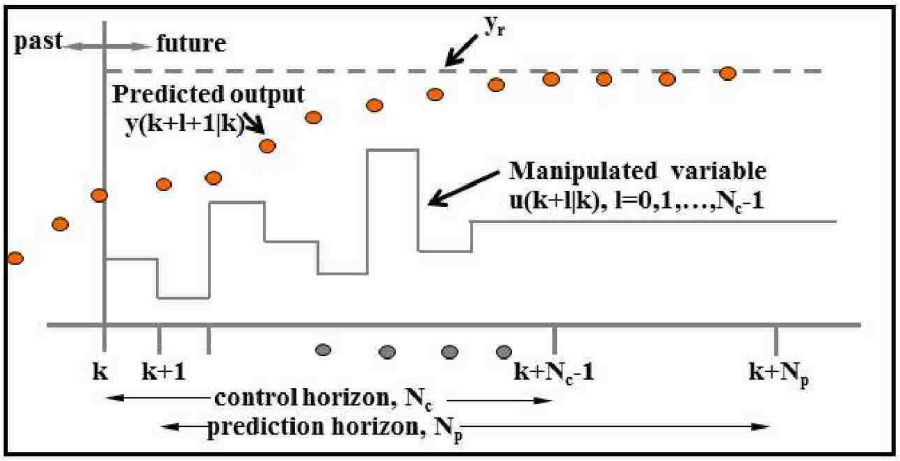



Figure 1.3: Prediction and controller move optimality in MPC

Using the predictions from the model, the $N_c$ control moves $u_{k+l|k}$, $l = 0, 1, \cdots, N_c - 1$ are optimally computed by minimizing a cost function $\mathbb{J}_k$ over the prediction horizon $N_p$ subject to constraints on the control inputs as well as any other constraints

on the internal states and outputs of the system as follows:

$$\min_{u_{k+l|k}, l=0,1,\cdots,N_c-1} \mathbb{J}_k \tag{1.1}$$

subjects to constraints on control inputs and the system. A typical quadratic objective cost function may be of the form

$$\mathbb{J}_k = \sum_{l=0}^{N_p-1} \{[y_{k+l+1|k} - y_r]^T Q [y_{k+l+1|k} - y_r]\} + \sum_{l=0}^{N_c-1} \{u_{k+l|k}^T R u_{k+l|k}\}. \tag{1.2}$$

Here, $y_r$ is the output to be tracked. Only the first optimally computed move $u_{k|k}$ is implemented out of $m$ computed optimal moves at time $k$. At the next time $k+1$, new system measurements are obtained and the optimization problem is solved again with the new measurements. Thus, the control and prediction horizon recede by one step as time moves ahead by one step. The measurements at each sampling time provide feedback for rejecting inter-sample disturbances, model uncertainty and noises, all of which cause the model predictions to be different from the true system output.

### 1.1.4 System Dynamics

As mentioned in the previous section, a model-based receding horizon controller uses a dynamical model of the system in computing optimal control moves. For BMI applications, a dynamical model of the system includes a network model of cortical neurons which captures essential cortical circuit relevant to a given motor task, a decoder, and a model of prosthetic arm.

**Dynamical Models of Single Neurons and Cortical Networks**

Figure 1.4 shows a typical structure of a single neuron. As shown in this figure, dendrites of the neuron make synaptic connections with axons of other neurons and thus receive input information through synaptic transmission.

Briefly, inputs received by the neuron at dendrite terminals in specific time windows are integrated at the cell body which induces a voltage jump in the membrane potential of the cell body. A positive voltage jump increases the membrane potential and leads to depolarization of the membrane potential. A negative voltage jump decreases the membrane potential and leads to repolarization or hyperpolarization of the membrane potential. When the depolarization of the membrane potential reaches a critical level called threshold, the neuron fires all-or-none action potential as shown in Figure 1.5 [87].

1. **Single Neuron Models:**

   Neurons communicate with each other in a network by generating and transferring action potentials. The exact phenomenon through which these action potentials encode the information contained in presynaptic activities is still unknown. To understand this phenomenon of neural coding using rigorous mathematical analysis is an important problem in computational and theoretical neuroscience [32]. One of the challenges in attempting such analysis is to develop computationally efficient mathematical models of neurons which can predict action potentials observed in experimental conditions, both qualitatively and quantitatively. Several theoretical models, deterministic as well as stochastic, have been developed in this direction to describe the biophysical

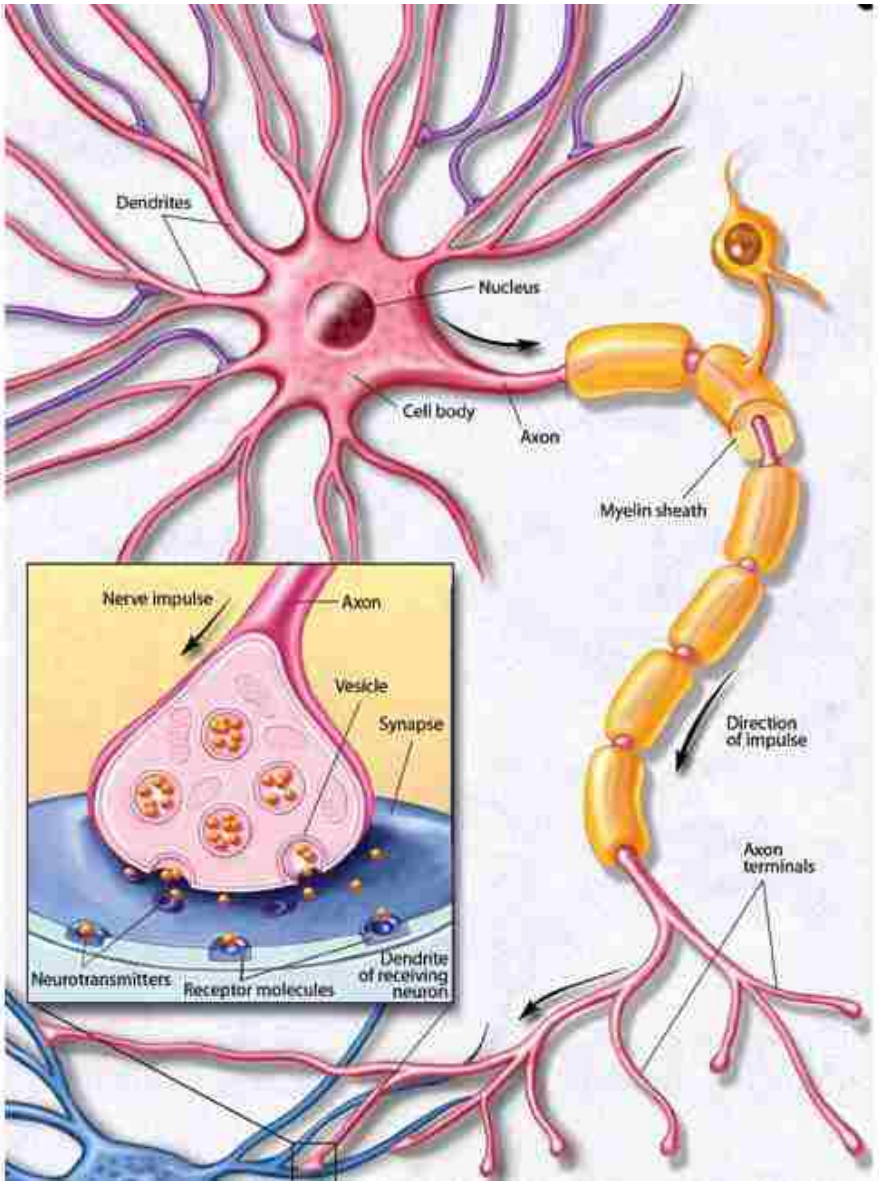



Figure 1.4: A single neuron, Figure taken from Brain Facts, Society for Neuroscience

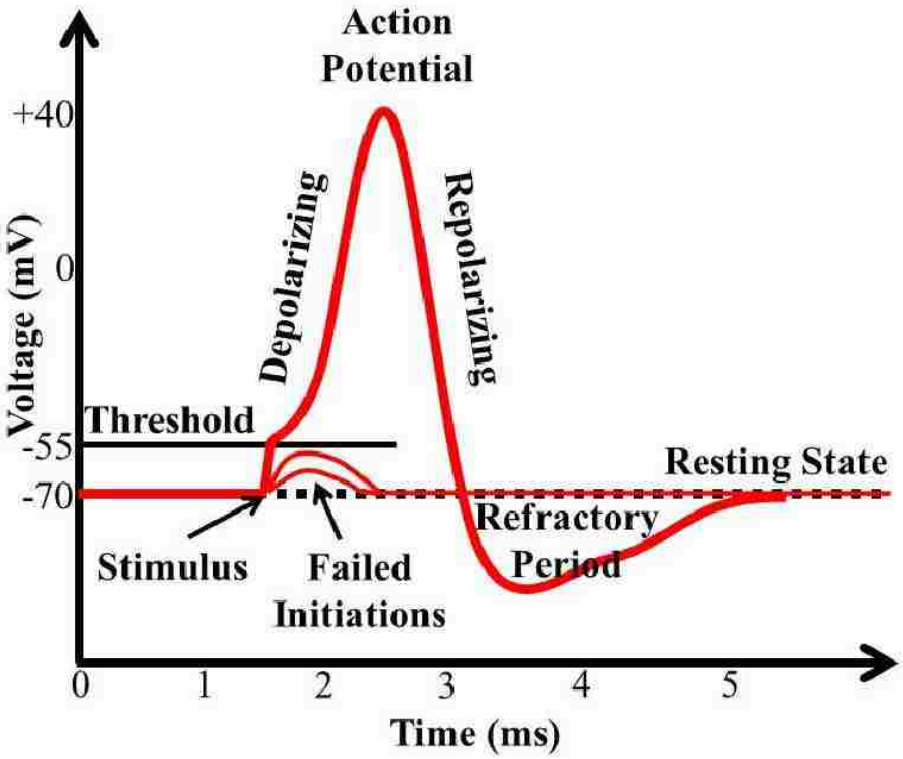



Figure 1.5: A schematic of an action potential.

behavior of single neurons. Based on the level of description of single neurons structure included in modeling their dynamical behavior, these models are classified into detailed compartmental models, reduced compartmental models, single-compartmental models, cascade models, and black-box models [69]. Detailed compartmental models and reduced compartmental models include spatial structure of a neuron (dendrites) and focus on how the spatial structure contributes to the dynamical and functional behavior of a neuron. Although these models can capture neuron's specific electronic structure and provide detailed understanding on spatial aspects of synaptic integration in dendrites, these models are extremely computationally expensive and are not suitable for modeling large-scale neuronal networks. Cascade models and black-box models ignore biophysics of single neurons and provide an appropriate framework for distilling key processing steps from experimentally measured data. These models are often based on probabilistic framework and are suitable for understanding the encoding of sensory information in single neurons and networks. On the other hand, single compartmental models neglect the spatial structure of a neuron and focus on the contribution of various ionic currents to neurons's subthreshold behavior and spike generation. These models capture essential biophysics of single neurons and provide quantitative understanding of various dynamical phenomena of single neurons including bursting, bistability, phasic spiking, and adaptation to spike frequency [78]. At the same time, these models offer computationally inexpensive simulation of large neuronal networks. Since our ultimate goal is to be able to use neuronal network models for real time prediction in an optimal feedback control framework, we will focus on a particular

class of single compartment models in this dissertation.

In the category of single compartment models, the simplest and most influential model is Integrate-and-Fire (I&F) model [1] introduced by Lapicque in 1907. Although mathematical simplicity of this model offers analytical analysis of the dynamical behavior, the model is incapable in capturing the dynamical richness of spiking behaviors in single neurons observed in experimental studies. On the other hand, the classic Hodgkin-Huxley model, introduced by Hodgkin and Huxley in 1952 [72], is based on biophysics of single cells and captures the exact shape of action potential shown in Figure 1.5. Despite the dynamical richness of spiking behaviors shown by this model, this model is still not very computationally efficient for its use in large-scale network simulations.

In the last decade, a class of models has been derived through the systematic mathematical reductions of the Hodgkin-Huxley model. In this class, specific models are the spike response model [56], the Izhikevich model [77], the adaptive exponential integrate-and-fire model [16], the Touboul model [158], and variations of linear integrate-and-fire model [137, 107]. These models ignore the detail biophysics of single cells and focus on computational efficiency in large-scale network simulations while retaining the dynamical richness of single cells observed in experimental studies. Moreover, these models have shown their capability in capturing broad range of qualitative behavior of biological neurons [74, 78, 157]. Most of these models are described by coupled first-order ordinary differential equations which show discontinuous behavior through reset conditions:

$$C\frac{\mathrm{d}v(t)}{\mathrm{d}t} = f(v(t), u(t)) + I(t), \tag{1.3a}$$

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} = g(v(t), u(t)), \tag{1.3b}$$

$$\begin{aligned} &if \quad v(t) \geq v_p(t), \quad then \\ &\quad v(t) \leftarrow c \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{1.3c}$$

Here, $v(t)$ and $u(t)$ are the time-varying membrane potential and the membrane recovery variable of a neuron respectively. $C$ is the membrane capacitance. $I(t)$ is the total input current delivered to the neuron. $v_p(t)$ is a firing threshold. $c$ and $d$ are the model parameters. Typically, $c$ is set to the membrane resting potential $v_r$. The time at which the membrane potential $v(t)$ reaches the firing threshold $v_p(t)$, starting from the resting state, is called the time of the occurrence of an action potential or the spike time. At this time, $v(t)$ is reset to $c$ and $u(t)$ is reset to $u(t) + d$. Again starting from this reset time and taking the reset values of $v(t)$ and $u(t)$ as initial conditions, the time of occurrence of the next action potential is determined through equation (1.3). The time difference between the occurrence of two consecutive action potentials is called inter-spike interval (ISI).

The obvious benefits of using these low-order models in the analysis of network of neurons for neural coding are their mathematical simplicity and existence of small number of model parameters which can easily be extracted from experimental data. Other benefits are in implementing neural prostheses where

these models can be simulated at low costs. In this dissertation, we will use the Izhikevich model [78, 76] to represent the dynamics of single neurons. The Izhikevich model is given by equation (1.3) with $f(v,u) = k(v-v_r)(v-v_t) - u$ and $g(v,u) = a(b(v-v_r) - u)$ where $a$, $b$ and $k$ are model parameters, $v_r$ is the membrane resting potential, and $v_t$ is the membrane instantaneous threshold. Depending on the choice of model parameters, this model is efficient in generating most of the neuronal spikes patterns observed in single neurons [78]. This model has also recently been used to simulate large number of mammalian thalamocortical neurons with synaptic connectivities [79].

It is well known that single cells are intrinsically noisy [161, 151, 125, 167, 45, 57]. Typically, this noisy characteristic of single neurons is captured in deterministic dynamical models of spiking neurons such as equation (1.3) by including an additive or multiplicative noise term in form of an external input current to the model. In this case, the dynamical model takes the form of stochastic differential equation [123] and the resultant sequence of ISIs (also known as the first passage times) becomes a stochastic process [133, 134, 159, 160, 150]. As an example, the leaky integrate-and-fire model takes the form of the well known Ornstein-Uhlenbeck process in the presence of Gaussian noise [21, 20, 10]. In this dissertation, we will use the Izhikevich model with Gaussian noise as a stochastic single neuron model in explaining experimentally observed single cell spike trains.

2. **Neuronal Network Models:**

   It is clear from Figure 1.4 that the dendrites of a postsynaptic neuron establish synaptic connections with the axons of its presynaptic neurons and thus

receive the input information in form of synaptic currents. Mathematically, the synaptic current $I^s(t)$ is modeled as

$$I^s(t) = -g_e(t)(v(t) - E_e) - g_i(t)(v(t) - E_i). \tag{1.4}$$

Here, $g_e(t)$ and $g_i(t)$ are the excitatory and inhibitory synaptic conductances respectively. $E_e$ and $E_i$ are the excitatory and inhibitory membrane reversal potentials respectively. Typically, the synaptic conductance $g_x(t),\ \ x \in \{e, i\}$ is modeled by taking the weighted sum of all presynaptic neuronal activities and is represented in the following form [56]:

$$g_x(t) = \sum_{j=1}^{N_x} \sum_f w_j K(t - t_j^f). \tag{1.5}$$

Here, $N_x$ is the total number of presynaptic neurons of type $x$. $w_j$ is the weight of the synapse $j$ to the post-synaptic neuron. $t_j^f$ is the time of the $f^{th}$ incoming action potential from the synapse $j$ to the post-synaptic neuron. $K(t - t_j^f)$ models the stereotypical time course of postsynaptic conductances following presynaptic spikes. Typical forms of $K(t - t_j^f)$ are

$$K(t - t_j^f) = \frac{q_j}{\tau_s} \exp(-(t - t_j^f)/\tau_s)\Theta(t - t_j^f), \tag{1.6}$$

and

$$K(t - t_j^f) = \frac{q_j}{\tau_s}(t - t_j^f) \exp(-(t - t_j^f)/\tau_s)\Theta(t - t_j^f). \tag{1.7}$$

Here, $q_j$ is the maximum conductance transmitted by the $j^{th}$ synapse through one action potential. $\tau(\cdot)$ is the time constant and $\Theta(\cdot)$ is the Heavyside function

with $\Theta(t - t_j^f) = 1$ for $t > t_j^f$ and $\Theta(t - t_j^f) = 0$ otherwise. With appropriate choices for the weight, type and number of synapses, equations (1.4), (1.5) and (1.7) along with single neuron dynamics such as equation (1.3) have shown the capability of a neuronal network of synaptically connected spiking neurons in predicting the qualitative behavior of various cortical areas under different dynamical regimes [17, 103, 166, 104, 64].

When the number of neurons present in a network is large and the exact timing of the occurrence of action potentials is irrelevant, a direct simulation of equations (1.3), (1.4), (1.5) and (1.7) is computationally expensive. An alternative approach is to represent the dynamics of neurons in terms of their firing rate i.e. rate at which spike occurs [108]. Several mathematical models based on population density approach [124, 90, 119, 65, 56, 51, 33, 149, 109] and mean-field density approach [46] have been developed in this direction. These models are typically represented by a nonlinear integral-differential equation and assume that the biophysical properties of neurons within a population are same for all neurons and each neuron within a population receives synaptic input with the same average rate. These assumptions make these models highly computationally efficient for simulating large number of synaptically connected similar neurons.

In this dissertation, we will use both the direct simulation approach and the rate-based approach to represent the dynamics of neuronal networks for BMIs applications.

### Decoder

The problem of determining the information (such as motor intent) contained in a spike train measured from an ensemble of neurons is referred to as the neural decoding problem [126]. There exists two approaches, complementary to each other, for extracting information encoded in neuronal populations, namely decoding algorithms and information theory [132]. Decoding algorithms are applicable for making prediction of a given stimulus or behavior from the pattern of ensemble neuronal responses while information theory is applicable for determining the quantity of information carried by neurons about a given stimuli.

It should be evident that the general neuronal decoding problem defies a mechanistic or physically motivated modeling paradigm and lends itself much more naturally to a range of signal processing, probabilistic and black-box techniques for the generation of models [145, 139]. These techniques typically employ input measurements, such as spike trains from ensemble of neurons, and the corresponding outputs such as motor commands, grasping, etc. to develop decoding models. In BMIs research, a wealth of tools and a variety of model families have been shown to give reasonable predictive models for motor parameters from neural spike trains (for example see [89, 91]). Typical model forms in this category are generative models based on population vector methods [54, 55], Kalman filters and their variations [173, 174, 112, 101, 31, 60], maximum likelihood estimators [11], Bayesian regression [102], Poisson processes [176], Wiener filters [154, 89], multi-layer feed-forward artificial neural networks (ANNs) [2, 4] and recurrent neural networks (RNNs) [153]. Among these decoding models, the most popular are Weiner filters and Kalman filters.

In this dissertation, we will use the following adaptive Weiner filter [89] and

Kalman filter [59] in designing BMIs:

1. **Weiner Filter [89]:**

   Figure 1.6 shows a schematic of a discrete-time Weiner filter for BMIs. Mathe-

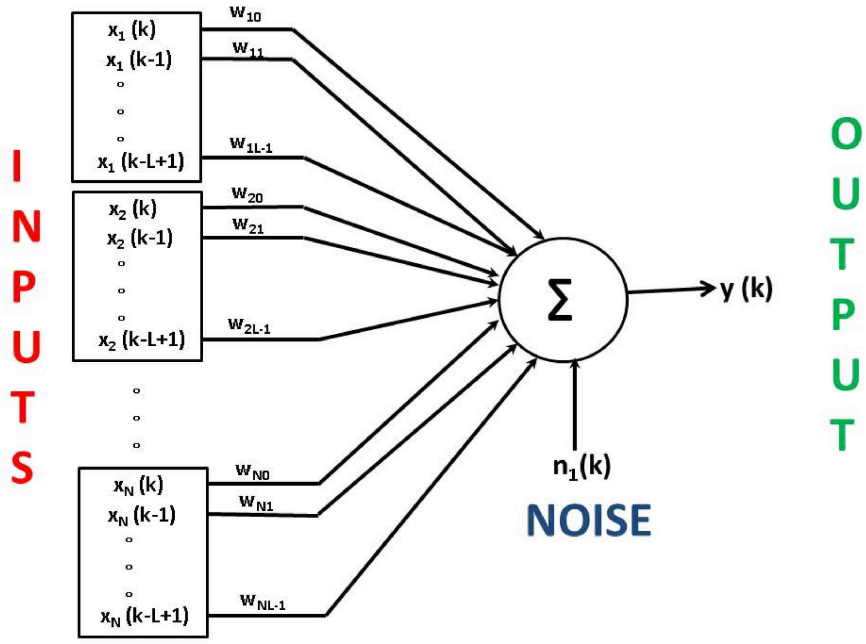



Figure 1.6: Schematic of a Discrete-time Weiner Filter. $x_m(k-l)$ represents the firing rate of neuron $m$ at time $k$ delayed by $l$ samples. $w_{m,l}$ is a weight on $x_m(k-l)$. $y(k)$ is the decoded output. $n_1(k)$ is the measurement noise.

matically, the decoded output $y(k)$ can be expressed in terms of the firing rate of neuronal ensemble as

$$y(k) = \sum_{l=0}^{L-1} \sum_{m=1}^{N} x_m(k-l) w_{ml} + n_1(k). \tag{1.8}$$

Here, $y(k)$ is a scalar decoded output. In BMIs, it can be prediction of hand position co-ordinate, velocity or acceleration at discrete time $k$. $L$ is the number

of delay elements. $N$ is the total number of neurons in the population. $x_m(k-l)$ represents the firing rate of the neuron $m$ delayed by $l$ samples. $w_{ml}$ is a weight on $x_m(k-l)$. In a vector form, equation (1.8) can be expressed as

$$y(k) = \mathbf{w}^T \mathbf{x}(k) + n_1(k). \tag{1.9}$$

Here, $\mathbf{w}$ is a $N.L \times 1$ dimensional weight vector of the current and $L-1$ previous spikes counts of $N$ neurons. $(\cdot)^T$ is the transpose of a vector. $x(k) = [x_1(k), x_1(k-1), \cdots, x_1(k-L+1), x_2(k), \cdots, x_N(k-L+1)]^T$.

In a typical BMI experiment, a part of the experimentally recorded data is used to train a chosen decoder model and thus estimate the unknown model parameters of the decoder. The rest of the data is then used to validate the performance of the decoder. Using the training data, the weighting matrix can be computed using the following normalized least mean square algorithm [89]:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\eta}{\beta + ||\mathbf{x}(k)||^2} \mathbf{e}(k)\mathbf{x}(k). \tag{1.10}$$

Here, $\eta \in (0, 2)$. $\beta$ is a small positive constant. $||\cdot||$ represents the Euclidean norm. $\mathbf{e}(k)$ is the error between the recorded $y(k)$ and the estimated value through equation (1.9). After the training, the weight vector '$\mathbf{w}$' is frozen to the final adapted value which is then used in equation (1.9) to validate the performance of the decoder. It should be noted that the scalar $y(k)$ in equation (1.9) can easily be extended to vector.

2. **Kalman Filter [86, 59]:**

The Kalman filter [86] uses a dynamical model of the system's state and noisy measurements available from sensors to estimate the future state of the system. For example, if the state of the system is known perfectly at a given time $k$ and the noisy measurements of outputs (or observations) are available from the system then the Kalman filter can be used to estimate the state of the system at time $k+1$. The basic algorithm of the Kalman filter can be described as follows [172]:

$$y(k+1) = Ay(k) + r(k), \tag{1.11a}$$

$$x(k) = Cy(k) + q(k). \tag{1.11b}$$

Here, $y(k)$ is the state vector of dimension $p \times 1$. $A \in \mathbb{R}^{p \times p}$ represents the state matrix. $x(k)$ is the observation vector of dimension $r \times 1$. $C \in \mathbb{R}^{r \times p}$ represents the observation matrix. $r(k) \sim \mathcal{N}(0, R)$ and $q(k) \sim \mathcal{N}(0, Q)$ are Gaussian noise sources.

The state matrix $A$ and the observation matrix $C$ can be estimated by least squares:

$$A = \arg\min_{A} \sum_{k=1}^{D-1} ||y(k+1) - Ay(k)||, \tag{1.12a}$$

$$C = \arg\min_{C} \sum_{k=1}^{D} ||x(k) - Cy(k)||. \tag{1.12b}$$

Here, $D$ is the total time steps in the training data. The solutions of equation (1.12) result in

$$A = Y_2 Y_1^T (Y_1 Y_1^T)^{-1}, \tag{1.13a}$$

$$C = X Y^T (Y Y^T)^{-1}. \tag{1.13b}$$

Here, $X = \begin{pmatrix} x_{1,1} & \cdots & x_{1,D} \\ \vdots & \ddots & \vdots \\ x_{r,1} & \cdots & x_{r,D} \end{pmatrix}$, $Y = \begin{pmatrix} y_{1,1} & \cdots & y_{1,D} \\ \vdots & \ddots & \vdots \\ y_{p,1} & \cdots & y_{p,D} \end{pmatrix}$, $Y_1 = \begin{pmatrix} y_{1,1} & \cdots & y_{1,D-1} \\ \vdots & \ddots & \vdots \\ y_{p,1} & \cdots & y_{p,D-1} \end{pmatrix}$, and $Y_2 = \begin{pmatrix} y_{1,2} & \cdots & y_{1,D} \\ \vdots & \ddots & \vdots \\ y_{p,2} & \cdots & y_{p,D} \end{pmatrix}$. Now using the estimated $A$ and $C$, one can estimate $R$ and $Q$ by

$$R = \frac{1}{D-1}(Y_2 - AY_1)(Y_2 - AY_1)^T, \tag{1.14a}$$

$$Q = \frac{1}{D}(X - CY)(X - CY)^T. \tag{1.14b}$$

The reconstruction of the state $y(k)$ using the Kalman filter can be described as follows:

At time $k$, a previous estimate of the state $y(k)$ i.e. $\hat{y}(k \mid k-1)$ and a new observation $y(k)$ are obtained. The first step is to compute *a priori* estimate of the state $y(k)$ using

$$\hat{y}(k \mid k-1) = A\hat{y}(k-1) \tag{1.15}$$

where $\hat{y}(k \mid k-1)$ is *a priori* estimate of $y(k)$ at time $k$ and $\hat{y}(k-1)$ is the estimate of $y(k)$ at time $k-1$. The estimate of *a priori* covariance of $\hat{y}(k \mid k-1)$ is computed using

$$\hat{P}(k \mid k-1) = A\hat{P}(k-1)A^T + R. \tag{1.16}$$

The final step is to compute *a posteriori* estimate $\hat{y}(k)$ (the final estimation for the state) of the state $y(k)$ and *a posteriori* estimate $\hat{P}(k)$ of the covariance matrix using

$$\hat{y}(k) = \hat{y}(k \mid k-1) + K_k(x(k) - C\hat{y}(k \mid k-1)), \tag{1.17a}$$

$$\hat{P}(k) = (I - K_k C)\hat{P}(k \mid k-1). \tag{1.17b}$$

Here, $I$ is an identity matrix. $K_k$ is the Kalman gain which is given by

$$K_k = \hat{P}(k \mid k-1)C^T(C\hat{P}(k \mid k-1)C^T + Q)^{-1}. \tag{1.18}$$

For neuroprosthetic applications, the state vector $y(k)$ can be expressed as $y(k) = [p_x(k), p_y(k), p_z(k), v_x(k), v_y(k), v_z(k), a_x(k), a_y(k), a_z(k), 1]^T$. Here, $p_i(k)$, $v_i(k)$ and $a_i(k)$ represent the position, velocity and acceleration of the arm effector in the $i^{th}$ direction at time $k$. The constant 1 is the fixed offset (e.g.

baseline firing rate) in the observations. $x(k)$ represents the measured neural signal which can be firing rate at time $k$ or the binned spike counts. The position, velocity and acceleration of the arm effector can be reconstructed from the firing activity of cortical neurons using equations (1.13)-(1.18). The typical bin width (in case of spike counts) ranges from 10 milliseconds (ms) to 300 ms. It has been shown in BMIs studies that the shorter bin widths result in better performance of the designed decoder [30].

In **Chapter 2**, **Chapter 3**, **Chapter 4**, and **Chapter 5** of this dissertation, we aim to develop feedback-enabled BMIs in an optimal receding horizon control framework towards the next generation BMIs.

## 1.2 Small Length Scale Dynamical Systems

By a small length scale dynamical system, we mean that the system is submicroscopic and the dynamical behavior of the system is governed by a stochastic process. Examples include but are not limited to particles driven by Brownian motion, nanobots, subcellular biological structures suspended in a fluidic environment, and a swarm of stochastically behaving agents.

Since the last decade, the attention of many control engineers has been focused in developing control strategies for (1) regulating trajectories of micro and nano level objects in a specific medium, and (2) manipulating the dynamical behaviors of individual entities in a swarm of identical agents with a goal of achieving a desired aggregate behavior of the ensemble. Emerging applications in this direction include but are not limited to

1. understanding of dynamical behaviors of biological objects and systems such as cellular interactions, DNA etc. in their natural environments for facilitating biomedical applications ([29],[61]),

2. regulation of submicroscopic particles in fluidic systems for applications in drug delivery systems and laboratory on a chip technologies ([42],[9],[62]),

3. design of smart robots and multi-agent intelligent systems for applications in automation (see [162],[25] and references therein).

The key challenge in facilitating applications such as (1) and (2) lies in the control over trajectories of these submicroscopic objects which are under continuous random fluctuations, also called Brownian fluctuations, induced by interactions of these objects with the medium in which they are placed. Thus the main question arises here is the design of optimal control strategies for suppressing or canceling these Brownian effects while achieving the desired performance of the system. Similarly, the key challenge in facilitating applications such as (3) lies in the control over the states of stochastically behaving agents for stabilizing the aggregate behavior of a vast number of identical agents when a limited feedback information is available from the system.

In **Chapter 6** and **Chapter 7** of this dissertation, we aim to develop stochastic optimal control strategies in a receding horizon framework to facilitate these emerging applications. Particular systems of study are multi-agent system (**Chapter 6**) and an ensemble of particles driven by purely Brownian motion (**Chapter 7**).

### 1.2.1 Multi-Agent System

By an agent, we mean an autonomous entity which is capable in making its own decision in response to the input. Examples include biological cell, bacteria, bird, human, intelligent robot etc. An agent is called simple if it possesses discrete simple states. As an example, the states of a simple agent can be represented by switching behaviors such as "ON" and "OFF". Based on the input, the agent can decide to stay at the present state or to switch to the other available state.

In Chapter 6 of this dissertation, we consider a multi-agent system consists of a vast number of identical simple agents. Examples include an ensemble of endothelial cells, artificial muscle actuators, a colony of E. coli bacteria, and a swarm of simple agents. In all these examples, the common goal for agents is to achieve a desired number of particular state. The underlying complexity in assigning individual controllers to each agent in the ensemble with very few available actuators has limited direct applicability of traditional control strategies such as decentralized and distributed control policies for achieving such goals.

Recently, Asada and his group [162] have introduced the concept of broadcast feedback control, a centralized control strategy, for stabilizing the aggregate behavior of a vast number of identical agents when limited feedback information is available from the system. The authors have applied this control framework to the problems of endothelial cell migration [171] and artificial muscle actuators [162], [121]. Similar control architecture has later been applied to achieve desired aggregate behavior in a colony of E. coli bacteria [85] and supervising a swarm of simple agents [25]. Figure 1.7 shows a broadcast feedback control architecture employed in these studies.

In all these works, the dynamical behavior of individual agents in the system has

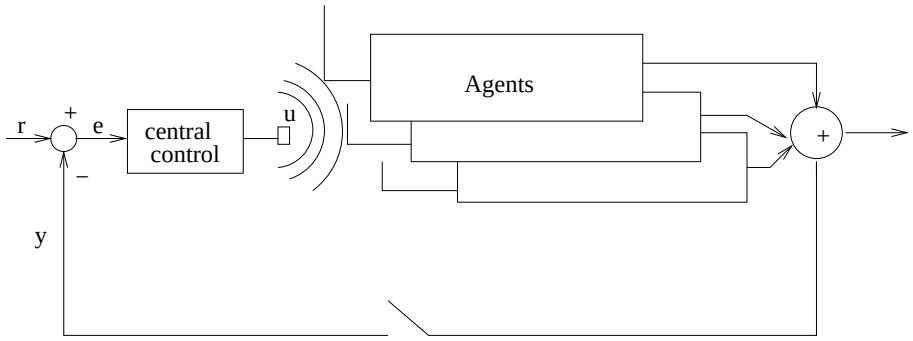



Figure 1.7: Broadcast feedback control architecture: Here $y$ is the measured output, $r$ is the desired output, $e = r - y$ is the system error, and $u$ is the designed control input. The central controller designs and broadcasts a single control input to all the agents in the system.

been represented by the finite state Markov chain model and their state transition probabilities have been used as manipulated variables to achieve the desired system behavior. The computed transition probabilities in most of these works are non-optimal (exception [121] and [122]) which may lead to a poor performance of the overall system. This provides the main motivation for us to develop an optimal centralized control framework for such systems in Chapter 6 of this dissertation.

### 1.2.2 Brownian Motion and Control

Brownian motion refers to the memoryless and non-differentiable path generated by random movements of the particle placed in the fluid. These random movements are the result of random collisions of a particle with the surrounding molecules of the fluid. Typical mathematical models used to describe the trajectory of a particle undergoing Brownian motion are random walk based models [41, 28], the Langevin equation and its variants [136, 88].

Suppression of Brownian fluctuations to facilitate the tracking, transport, sorting

and assembly of micro-particles as well as biological entities in solution has always been a non-trivial problem. In recent years, experimental efforts have focused on enhancing optical (see [62], [9] and [116]) and magnetic tweezers (see [61] and [175]) to obtain a more precise and reliable manipulation of individual particles. Feedback control is essential to make appropriate real-time corrections in the trajectories of these microscopic entities [6]. Recent applications of classical feedback controllers in the design of Anti-Brownian Electrophoretic (ABEL) trap [29] and electrokinetic tweezers (see [130] and references therein) have shown promise in this direction. Optimal feedback control algorithms can greatly enhance the capability of these devices for their widespread applicability [130]. More importantly, an optimal feedback control framework should be able to address control of not just individual entities but Brownian ensembles, an issue that is currently not fully resolved. This is the main motivation for us to develop an optimal feedback control strategy for regulating an ensemble of particles driven by Brownian motion in Chapter 7 of this dissertation.

### 1.2.3 Optimal Control Policies for Stochastic Dynamical Systems

Historically, theory of optimal control policies [140, 94] in stabilizing stochastic dynamical systems traces back to the seminal results of Kalman [86] and Kushner [92]. These results provided a theoretical framework which allows incorporation of probabilistic tools, such as conditional expectations and the theory of martingales [41], in designing closed-loop control policies for such systems [93]. Incorporation of receding horizon based predictive control policy [95] such as shown in Figure 1.3 within this framework can potentially result in a unique optimal control strategy for stabilizing

stochastic dynamical systems. Recent applications on stabilization of linear stochastic systems with constraints (see [129, 73] and references therein) have shown the capability of this framework for designing a model-based receding horizon control (RHC) policy. In Chapter 6 and Chapter 7 of this dissertation, we extend this framework for stabilizing dynamical systems consisted of an ensemble of stochastic entities.

## 1.3 Dissertation Overview

This dissertation is organized as follows: In **Chapter 2**, the problem of estimating model parameters using limited experimental data is addressed for both the deterministic and the stochastic Izhikevich single neuron model. Reasonable model parameters are estimated by formulating and solving optimization problems. The approach used here can serve as a template for validating the efficacy of single neuron models in predicting experimental data. In **Chapter 3**, we derive conditions for continuous differentiability of inter-spike intervals of spiking neurons with respect to parameters (decision variables) of an external stimulating input current which drives a recurrent network of synaptically connected neurons. The derived theoretical results in this chapter allow us to use a local gradient-based optimization algorithm in solving optimal control problems formulated for closed-loop brain-machine interfaces (BMIs). In **Chapter 4**, an optimal control framework using receding horizon control policy is developed to facilitate rigorous control-theoretic analysis of stimulus-based closed-loop BMIs. The framework allows spiking models of cortical neurons and networks, both deterministic and stochastic, as well as dynamical models of decoders in analyzing closed-loop performance of BMIs. Under this framework, optimal control problems are formulated using a single neuron model to investigate the role of sensory feedback

in closed-loop BMIs. In **Chapter 5**, we design optimal artificial sensory feedback to recover the closed-loop performance of a BMI for voluntary single joint extension task in the absence of the natural proprioceptive feedback pathways. The closed-loop BMI framework presented in this chapter allows firing rate-based cortical network models in designing optimal artificial sensory feedback. The study presents the first systematic approach to incorporate artificial proprioception in BMIs towards stimulation enhanced next generation BMIs. Next, in **Chapter 6**, a broadcast stochastic receding horizon control framework is developed using a limited feedback information available from the system in stabilizing a swarm of identical simple agents. Theoretical as well as simulation results are presented to demonstrate the efficacy of the designed controller over traditional controller designs. In **Chapter 7**, we investigate the optimal trapping of an ensemble of particles driven by Brownian motion in a minimum possible trapping region. Finally, we summarize the work presented in this dissertation and present future work directions in **Chapter 8**.

# Chapter 2

# Investigation on Parameter Estimation of Single Neuron Models

## 2.1 Introduction

Neurons communicate with each other in a network by generating and transferring action potentials [87]. The exact phenomenon through which these action potentials encode the information contained in presynaptic activities is still unknown. To understand this phenomenon of neural coding using rigorous mathematical analysis is an important problem in computational and theoretical neuroscience [32]. One of the challenges in attempting such analysis is to develop computationally efficient mathematical models of neurons which can predict action potentials observed in experimental conditions, both qualitatively and quantitatively. Recent efforts towards

developing single neuron models have shown the capability of mathematically simplified low-order models such as spike response model [56], the Izhikevich single neuron model [76, 77], the adaptive exponential integrate-and-fire model [16], and a generalized integrate-and-fire model [106] in capturing broad range of qualitative behavior of biological neurons. The obvious benefits of using these low-order models in the analysis of network of neurons for neural coding are their mathematical simplicity and existence of small number of model parameters which can easily be extracted from experimental data. Other benefits are in implementing neural prostheses where these models can be simulated at low costs. To choose any of these models for applications, it is necessary to analyze the quantitative capability of these models in predicting experimental data [83].

In the past, several results have been reported on estimating parameters of both deterministic and stochastic single neuron models using either detailed experimental data such as subthreshold voltage traces or inter-spike interval (ISI), the time difference between two consecutive action potentials, data [80, 128, 67, 35, 5, 36, 97, 81, 26, 37, 113, 96, 53, 82, 40, 137, 18, 39, 138]. Surprisingly, none of these studies have reported results on the deterministic and the stochastic Izhikevich single neuron model [78] with a full set of model parameters.

In this chapter, we first estimate parameters of a reduced form of the conductance-based representation of the deterministic Izhikevich model using single cortical neuron experimental data obtained from a primate study [142]. The approach presented to estimate model parameters of this model is appropriate when the only available experimental data is ISIs. Next, we estimate parameters of both the deterministic and the

stochastic Izhikevich single neuron model with full set of parameters using a benchmark set of experimental data available from "Quantitative single-neuron modeling competition 2009" [114] and investigate the quantitative predictive capability of this model in explaining experimental data. The approach presented to estimate model parameters of these models is appropriate when both ISIs and the input current delivered to a single neuron are available from experiments. Our analysis may serve as a benchmark comparison of the performance of this model against other existing single neuron models.

## 2.2 Single Neuron Models

We consider two forms of the Izhikevich single neuron model [76, 77, 78] namely, "Model I" and "Model II" to represent the dynamics of a single neuron. "Model I" is a reduced form of the conductance-based representation of the Izhikevich model ("Model II") with a full set of model parameters. The mathematical description of both models is given as follows:

### 2.2.1 Model I

In a reduced form, the Izhikevich model [76, 77] is represented as

$$\frac{\mathrm{d}v(t)}{\mathrm{d}t} = 0.04v^2(t) + 5v(t) + 140 - u(t) + RI(t), \tag{2.1a}$$

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} = a(bv(t) - u(t)), \tag{2.1b}$$

$$\begin{aligned} &if \quad v(t) \geq 30, \quad then \\ &\quad v(t) \leftarrow c \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{2.1c}$$

Here, $v(t)$ and $u(t)$ are the time-varying membrane potential and the membrane recovery variable of a neuron respectively in millivolts (mV). $I(t)$ is an external input current in nano-Ampere (nA). $R$ is the membrane resistance. Throughout this work, we assume that $R = 10^6\ \Omega$. $a, b, c, d$ are the model parameters. The occurrence of an action potential (spike) is assumed whenever $v(t)$ exceeds 30 mV. At this time, $v(t)$ is reset to $c$ and $u(t)$ is reset to $u(t) + d$. In this model, unknown parameters are $a, b, c, d$ which can be estimated using experimental data. The presence of small number of parameters makes this model appropriate for simulating single neurons.

The recovery variable $u(t)$ accounts for the activation and inactivation of $K^+$ and $Na^+$ ionic currents respectively and provides a negative feedback to the membrane potential $v(t)$. The time scale of the recovery variable $u(t)$ is described by the parameter $a$ whereas the sensitivity of $u(t)$ to the subthreshold fluctuations of the membrane potential $v(t)$ is characterized by the parameter $b$. After-spike reset value of the membrane potential caused by the fast high-threshold $K^+$ conductances is described by the reset parameter $c$. After-spike reset of the recovery variable $u(t)$ caused by slow $Na^+$ and $K^+$ conductances is given by parameter $d$ [76]. Detailed geometric characterization of these variables are given in [78]. In this work, the membrane potential reset value $c$ is chosen to be the membrane resting potential or the equilibrium potential $v_{eq}$ and is calculated here as

$$v_{eq} = \frac{(b-5) - ((5-b)^2 - 22.4)^{0.5}}{0.08}. \tag{2.2}$$

### 2.2.2 Model II

The Izhikevich model [78] with a full set of model parameters is given by

$$C\mathrm{d}v(t) = (k(v(t) - v_r)(v(t) - v_t) - u(t) + I(t))\mathrm{d}t + \sigma \mathrm{d}W(t), \tag{2.3a}$$

$$\mathrm{d}u(t) = a(b(v(t) - v_r) - u(t))\mathrm{d}t, \tag{2.3b}$$

$$\begin{aligned} &if \quad v(t) \geq v_p, \quad then \\ &\quad v(t) \leftarrow c \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{2.3c}$$

Here, $v(t)$ and $u(t)$ are the time-varying membrane potential and the membrane recovery variable of a neuron respectively in mV. $C$ is the membrane capacitance in micro-Farad ($\mu$F). $v_r$ is the membrane resting potential in mV. $v_t$ is the instantaneous threshold in mV. $I(t)$ is the deterministic total input current, external as well as synaptic, delivered to the neuron in nano-Ampere (nA). $\sigma^2$ is the variance. $W(t)$ is the standard Wiener process. $v_p$ is a firing threshold in mV. $a, b, c, d$ are the model parameters. With $\sigma = 0$, the model is deterministic and with $\sigma > 0$, the model is stochastic. The occurrence of an action potential is assumed whenever $v(t)$ exceeds $v_p$. At this time, $v(t)$ is reset to $c$ and $u(t)$ is reset to $u(t) + d$. An inter-spike interval (ISI) is defined as the time difference between the occurrence of two consecutive action potentials. In this model, unknown parameters are $a, b, c, d, v_r, v_t, v_p, k, C, \sigma$. The model is appropriate for simulating large scale networks of conductance-based synaptically connected neurons.

## 2.3 Parameter Estimation: Model I

In this section, we estimate unknown parameters of "Model I" using experimental inter-spike intervals (ISIs) of a single cortical neuron obtained from a primate study. For this, we formulate two optimization problems, namely unconstrained and constrained. Since the only available data for model fitting is ISIs, we also estimate the input current $I(t)$ to the model.

### 2.3.1 Problem Statement

Our goal is to estimate optimal values of unknown parameters $a, b, c, d$ and the input current $I(t)$ of "Model I" such that the estimated ISIs using the model matches the experimental ISIs. We assume that the input current $I(t)$ within an ISI is constant. The value of $c$ is given by equation (2.2). With this, we formulate the following non-linear optimization problems to estimate the remaining parameters of the model as well as the input current $I(t)$:

1. **Unconstrained Non-Linear Optimization Problem** We formulate the unconstrained non-linear optimization problem as follows:

$$\min_{a,b,d,I_j} N(T_j^m - T_j^e)^2. \tag{2.4}$$

Here, $\mathbb{J}$ is the optimization cost function. $j = 1, \cdots, n$ represents the index of the experimental ISI data. $T_j^e$ and $T_j^m$ are the $j^{th}$ experimental ISI data and $j^{th}$ ISI estimated by the model respectively. $a, b, d$ are unknown model parameters. $I_j$ is the input current corresponding to the $j^{th}$ ISI. $N$ is the number of replications of the $j^{th}$ ISI.

2. **Constrained Non-Linear Optimization Problem** We formulate the constrained non-linear optimization problem as follows:

$$\min_{a,b,d,I_j} N(T_j^m - T_j^e)^2, \tag{2.5a}$$

such that

$$22.4 - (5-b)^2 < 0, \tag{2.5b}$$

$$\Re(0.08v_{eq} + 5 - a + ((a - 0.08v_{eq} - 5)^2 - 4a(b - 0.08v_{eq} - 5))^{0.5}) < 0. \tag{2.5c}$$

Here, $\Re$ represents the real part of a complex number. The constraint defined by equation (2.5b) ensures the existence of feasible equilibrium points of equation (2.1) in the absence of any external input current ($I_j = 0$). The constraint defined by equation (2.5c) ensures the existence of a stable equilibrium point of equation (2.1) in the absence of any external input current.

### 2.3.2 Primal-Dual Interior-Point Method

In this section, we describe a basic primal-dual interior-point algorithm [15] for solving constrained optimization problems. In later sections, we will apply this algorithm in solving equation (2.5). To begin with, let us consider the following optimization problem:

$$\min_{\mathbf{x}} f_0(\mathbf{x}), \tag{2.6a}$$

such that

$$f_i(\mathbf{x}) \leq 0, \qquad i = 1, 2, \cdots, m \tag{2.6b}$$

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \tag{2.6c}$$

Here, $\mathbf{x}$ is a vector of parameters (decision variables) of dimension $n \times 1$. $f_i(\mathbf{x}) : \mathbb{R}^n \longrightarrow \mathbb{R}$, $i = 0, 1, \cdots, m$ is a convex function on the real space and is two times continuously differentiable with respect to (w.r.t.) $\mathbf{x}$. $\mathbf{A}$ is a real matrix of dimension $p \times n$ with $\mathbf{rank}(\mathbf{A}) = p < n$. $\mathbf{b}$ is a vector of dimension $p \times 1$.

The modified Karush-Kuhn-Tucker (KKT) conditions [15] which lead to a feasible optimal solution of equation (2.6) are given by

$$\nabla f_0(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \nabla f_i(\mathbf{x}) + \mathbf{A}^T \nu = 0, \tag{2.7a}$$

$$-\lambda_i f_i(\mathbf{x}) = 1/t, \qquad i = 1, \cdots, m, \tag{2.7b}$$

$$\mathbf{A}\mathbf{x} = \mathbf{b}. \tag{2.7c}$$

Here, $\nabla$ is a gradient operator. $\lambda_i$, $i = 1, 2, \cdots, m$ and $\nu$ are Lagrange multipliers associated with equation (2.6b) and equation (2.6c) respectively. $m/t$ with $t > 0$ is the duality gap. In a vector form, equation (2.7) can be expressed as

$$\mathbf{r}_t(\mathbf{x}, \lambda, \nu) = \begin{bmatrix} \nabla f_0(\mathbf{x}) + D\mathbf{f}(\mathbf{x})^T \lambda + \mathbf{A}^T \nu \\ -\mathbf{diag}(\lambda)\mathbf{f}(\mathbf{x}) - (1/t)\mathbf{1} \\ \mathbf{A}\mathbf{x} - \mathbf{b} \end{bmatrix} = 0. \tag{2.8}$$

Here, $(\cdot)^T$ represents the transpose of a vector. $\mathbf{f}(\mathbf{x})$ and $D\mathbf{f}(\mathbf{x})$ are given by

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{bmatrix}, \tag{2.9a}$$

$$D\mathbf{f}(\mathbf{x}) = \begin{bmatrix} \nabla f_1(\mathbf{x})^T \\ \vdots \\ \nabla f_m(\mathbf{x})^T \end{bmatrix}. \tag{2.9b}$$

$\mathbf{x} = \mathbf{x}^*(t)$, $\lambda = \lambda^*(t)$ and $\nu = \nu^*(t)$ are called the optimal solution of equation (2.6) if $\mathbf{x} = \mathbf{x}^*(t)$, $\lambda = \lambda^*(t)$ and $\nu = \nu^*(t)$ satisfy equation (2.8) along with $f_i(\mathbf{x}) < 0$ for $i = 1, 2, \cdots, m$. In the absence of an analytical expression for the solution of equation (2.8), equation (2.8) is solved numerically using the Newton method:

$$\begin{bmatrix} \nabla^2 f_0(\mathbf{x}) + \sum_{i=1}^m \lambda_i \nabla^2 f_i(\mathbf{x}) & D\mathbf{f}(\mathbf{x})^T & \mathbf{A}^T \\ -\mathbf{diag}(\lambda) D\mathbf{f}(\mathbf{x}) & -\mathbf{diag}(\mathbf{f}(\mathbf{x})) & \mathbf{0} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x} \\ \Delta\lambda \\ \Delta\nu \end{bmatrix} = - \begin{bmatrix} r_{dual} \\ r_{cent} \\ r_{pri} \end{bmatrix}. \tag{2.10}$$

Here, $r_{dual}$, $r_{pri}$, and $r_{cent}$ are given by

$$r_{dual} = \nabla f_0(\mathbf{x}) + D\mathbf{f}(\mathbf{x})^T \lambda + \mathbf{A}^T \nu, \tag{2.11a}$$

$$r_{pri} = \mathbf{A}\mathbf{x} - \mathbf{b}, \tag{2.11b}$$

$$r_{cent} = -\mathbf{diag}(\lambda)\mathbf{f}(\mathbf{x}) - (1/t)\mathbf{1}. \tag{2.11c}$$

**Primal-dual interior-point algorithm:**

Choose an initial guess $\mathbf{x} = \mathbf{x}_0$ such that $f_i(\mathbf{x}) < 0$ for $i = 1, 2, \cdots, m$. Choose values of $\lambda$, $\mu$, $\epsilon_{feas}$, and $\epsilon$ such that $\lambda \succ 0$, $\mu > 1$, $\epsilon_{feas} > 0$, and $\epsilon > 0$. Define $\hat{\eta}(\mathbf{x}, \lambda) = -\mathbf{f}(\mathbf{x})^T\lambda$. Let $\mathbf{y} = (\mathbf{x}, \lambda, \nu)^T$ and $\Delta\mathbf{y} = (\Delta\mathbf{x}, \Delta\lambda, \Delta\nu)^T$. Now apply the following steps to compute $\mathbf{x}^*(t)$, $\lambda^*(t)$ and $\nu^*(t)$:

1. Set $t := \mu m/\hat{\eta}$.
2. Compute the primal-dual search direction $y$ by solving equations (2.10) and (2.11).
3. Use a line search method to compute step length $s > 0$ and update $\mathbf{y} := \mathbf{y} + s\Delta\mathbf{y}$.
4. Repeat 1, 2 and 3 until $||r_{pri}|| \leq \epsilon_{feas}$, $||r_{dual}|| \leq \epsilon_{feas}$, and $\hat{\eta} \leq \epsilon$.

### 2.3.3 Algorithm

We apply the following strategy in solving the unconstrained and the constrained optimization problem defined by equation (2.4) and equation (2.5) respectively:

1. Minimize $N(T_1^m - T_1^e)^2$ w.r.t. $a, b, d, I_1$ and obtain the optimal value of $a, b, d$ and $I_1$;
2. Fix the optimal value of $a, b, d$ obtained in the previous step and estimate the input current $I_j$ for $j > 1$ by minimizing objective function $N(T_j^m - T_j^e)^2$ w.r.t. $I_j$.

**Unconstrained Optimization Problem (equation (2.4))**

Using the above strategy, the solution of the unconstrained optimization problem (equation (2.4)) leads to the following conditions:

$$\min_{a,b,d,I_1} \mathbb{J} \qquad \Longrightarrow \nabla\mathbb{J} = 0 \qquad and \qquad \nabla^2\mathbb{J} > 0. \tag{2.12}$$

Here, $\mathbb{J} = N(T_1^m - T_1^e)^2$. $\nabla\mathbb{J}$ represents the gradient of $\mathbb{J}$ w.r.t. $a, b, d, I_1$. $\nabla^2\mathbb{J}$ represents the second derivative of $\mathbb{J}$ w.r.t. $a, b, d, I_1$. Given the initial guess of $a$, $b$, $d$, and $I_1$, $\nabla\mathbb{J}$ and $\nabla^2\mathbb{J}$ are computed numerically using finite difference schemes. At each iteration, $a$, $b$, $d$, and $I_1$ are updated using the following Newton method:

$$\mathbf{x}^{new} = \mathbf{x}^{old} - t(\nabla^2\mathbb{J})^{-1}\nabla\mathbb{J}. \tag{2.13}$$

Here, $\mathbf{x} = [a\ b\ d\ I_1]^T$. At each iteration, the value of $c$ is updated using equation (2.2). The parameter $t$ is updated using the backtracking line search method. Stopping criterion is determined by the Newton decrement function which is defined as $\beta(x) = (\nabla\mathbb{J}(x)^T\nabla^2\mathbb{J}(x)^{-1}\nabla\mathbb{J}(x))^{0.5}$ [15].

**Constrained Optimization Problem (equation (2.5))**

The Primal-dual interior-point method given in section (2.3.2) is applied to solve the constrained optimization problem. The following modified Karush-Kuhn-Tucker (KKT) conditions are used for the implementation of the primal-dual interior-point

algorithm:

$$r_t(x, \lambda_1, \lambda_2) = \begin{bmatrix} \nabla \mathbb{J}(x) + \lambda_1 \nabla g(x) + \lambda_2 \nabla h(x) \\ -\lambda_1 g(x) - \frac{1}{t} \\ -\lambda_2 h(x) - \frac{1}{t} \end{bmatrix} = 0. \tag{2.14}$$

Here, $g(x) = 22.4 - (5-b)^2$. $h(x) = \Re(0.08v_{eq} + 5 - a + ((a - 0.08v_{eq} - 5)^2 - 4a(b - 0.08v_{eq} - 5))^{0.5})$. $v_{eq} = \frac{(b-5)-((5-b)^2-22.4)^{0.5}}{0.08}$. $\lambda_1$ and $\lambda_2$ are the Lagrange multipliers. $t = \frac{\mu}{\hat{\eta}}$. $\hat{\eta} = -\lambda_1 g(x) - \lambda_2 h(x)$ is the surrogate gap [15].

### 2.3.4 Experimental Data

We use the experimentally obtained cortical neuron ISI data from a primate study to estimate the unknown parameters of "Model I". All the experiments were carried out at the University of Rochester Medical Center by Prof. Marc H. Schieber. Experimental set-up and details are discussed in [142, 3]. Briefly, a male rhesus monkey - K -, trained to perform visually-cued finger and wrist movements, placed his right hand in a pistol-grip manipulandum in order to extend or flex a digit by a few millimeters. For instructing the monkey to close the switch by extension or flexion, visual cues were presented to the monkey using LEDs. Experimental data acquisition was accomplished using single unit recordings of neuronal activities in the primary motor cortex (M1) area of a monkey. Self-made, glass-coated, Pt-Ir microelctrodes [3] were implanted through surgical procedure to record the time of single neuron action potentials. A schematic diagram of spikes recording from a single neuron is shown in Figure 2.1.

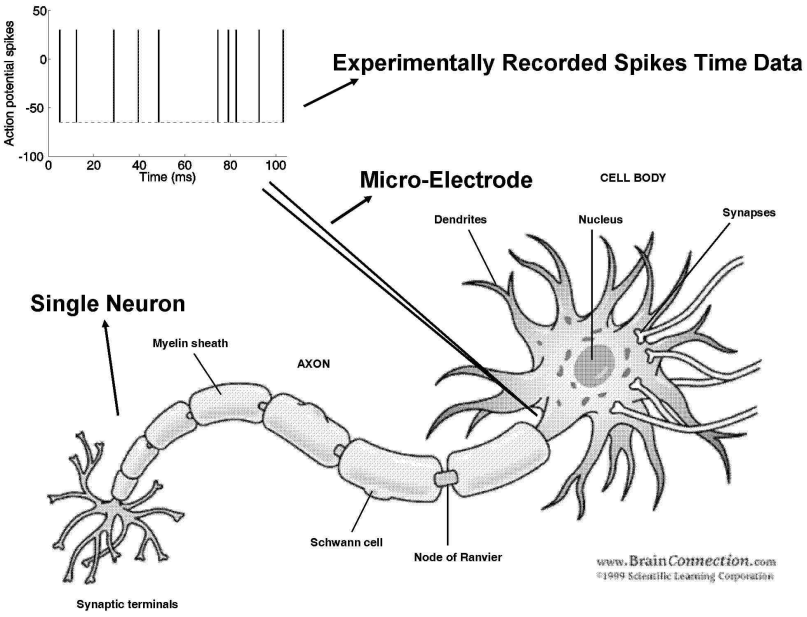



Figure 2.1: Experimental recordings of spike timings from a single neuron. The neuron figure is taken from www.brainconnections.com

### 2.3.5 Numerical Results

In this section, we solve the optimization problems defined by equations (2.4) and (2.5) numerically and estimate the unknown parameters of "Model I" using experimental ISI data. In particular, we use the ISIs of two neurons namely "K11404" and "K15906" which were recorded during an extension of the right hand index finger of a primate [4]. The finger movement trials selected for the present analysis lasted in an approximate time duration of $700ms - 1300ms$ [142]. The total number of ISIs occurred during this time period was 34 and 43 for the neuron "K11404" and the neuron "K15906" respectively.

#### Unconstrained Optimization Problem

To solve the unconstrained optimization problem defined by equation (2.4) numerically, we set $N = 20$ in equation (2.4) and applied the algorithm described in section 2.3.3 for the unconstrained optimization problem. $a = 0.02$, $b = 0.25$, $d = 6$ and $I_1 = 20$ were chosen as the initial guess for performing numerical optimization. We set $\frac{\beta^2}{2} < 10^{-4}$ as the stopping criteria for the optimization algorithm. Using the first ISI of the neuron "K11404", we estimated $a = 0.08$, $b = 0.59$, and $d = 15.57$. Using these values, we computed the input current $I_j$ for the remaining 33 ISIs. Figure 2.2 shows our simulation results.

In Figure 2.2, the top plot represents the experimentally obtained action potentials. The second plot shows action potentials predicted by the model using the estimated parameters and the input current. Almost exact matching of theoretical and experimental ISIs shows that the estimated parameters are appropriate to

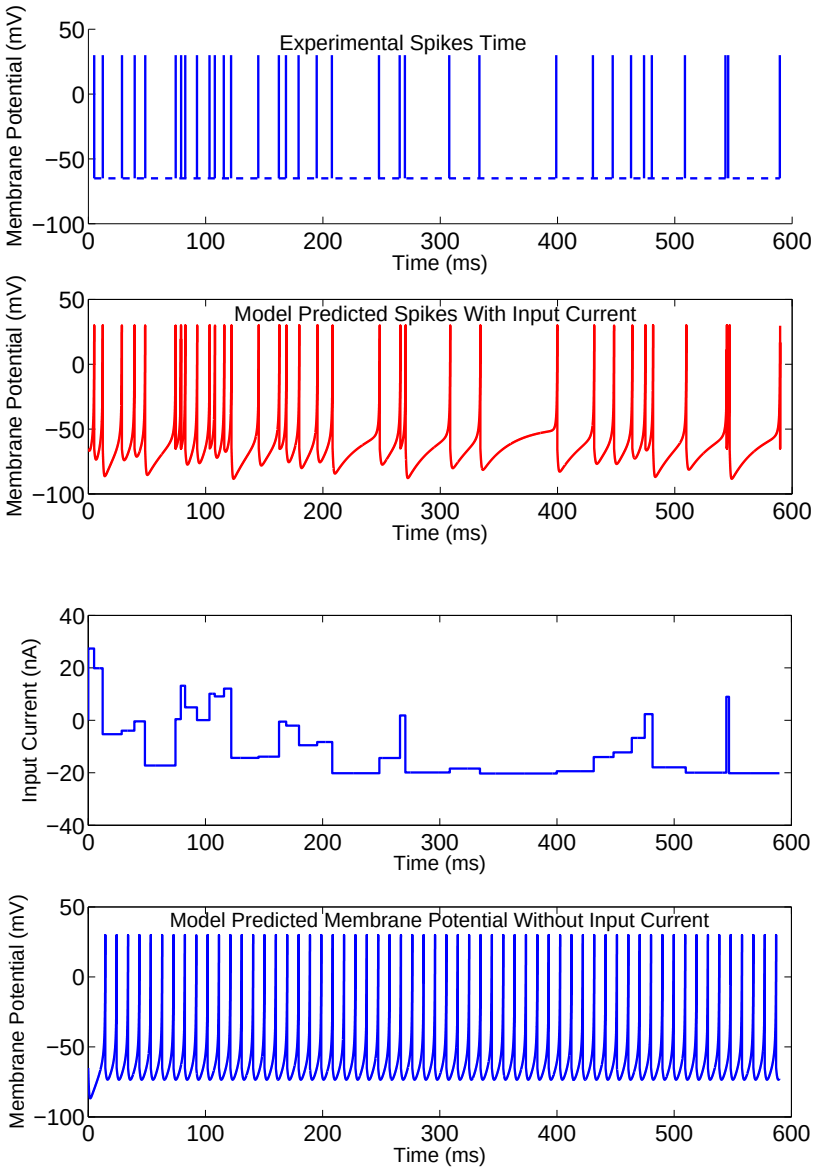



Figure 2.2: Unconstrained parameter estimation for the neuron "K11404".

characterize properties of the "K11404" neuron. The third plot represents the trajectory of the membrane potential in the absence of input currents i.e. $I_j = 0$ for $j = 1, 2, \cdots, 34$. The variation in the input current from one ISI to another is shown in the last plot. A positive current represents an excitatory and a negative current represents an inhibitory input to the neuron.

As shown in the third plot of Figure 2.2, the model predicts a sequence of action potentials even in the absence of any external input currents. It is well known that an isolated single neuron fires action potentials only in the presence of input currents. This indicates that the estimated parameters for this model are not appropriate to characterize the dynamical behavior of a biophysical single neuron.

Mathematical analysis of "Model I" (equations (2.1) and (2.2)) with $I(t) = 0$ shows that $(5 - b)^2 - 22.4 \geq 0$ i.e. $b \in (-\infty, 0.27] \cup [9.73, \infty)$ is necessary to obtain real equilibrium points of equation (2.1). Solution of the unconstrained optimization problem clearly shows that the estimated value of $b$ i.e. $b = 0.59$ is outside the domain of the feasible equilibrium space. This clearly indicates that the above estimated model parameters are not appropriate to represent the dynamical behavior of a single neuron.

These analyses suggest that there is a need to impose constraints on the parameters of the Izhikevich model such that feasible equilibrium points exist for the model in the absence of any external input currents.

### Constrained Optimization Problem

To solve the constrained optimization problem defined by equation (2.5) numerically, we set $N = 5$ in equation (2.5) and implemented the primal-dual interior-point

method in MATLAB. $a = 0.02$, $b = 0.25$, and $d = 6$ were chosen as the initial guess for performing numerical optimization. $I_1 = 20$ and $I_1 = 25$ were chosen as an initial guess of the input current to the neuron "K11404" and the neuron "K15906" respectively. The stopping criteria for the primal-dual interior-point algorithm was chosen based on $||r_t(x, \lambda)||_2 \leq 0.3$. $\mu = 10$ and $\mu = 4$ were chosen to initialize the parameter $t$ for the neuron "K11404" and the neuron "K15906" respectively. Using the first ISI of the neuron "K11404" and the neuron "K15906", we estimated $a = 0.23$, $b = 0.24$ and $d = 27.19$ for the neuron "K11404" and $a = 0.21$, $b = 0.25$ and $d = 16.71$ for the neuron "K15906". Using these values, we computed the input current $I_j$ for the remaining 33 ISIs of the neuron "K11404" and 42 ISIs of the neuron "K15906". Figures 2.3 and 2.4 show our simulation results for the neuron "K11404" and the neuron "K15906" respectively.

In both Figure 2.3 and Figure 2.4, the top plot shows the experimentally recorded action potentials. The second plot shows action potentials predicted by the Izhikevich model in the presence of input currents. The third plot shows the trajectory of the membrane potential predicted by the Izhikevich model in the absence of input currents. As clearly shown in this plot, the predicted trajectory of the membrane potential by the Izhikevich model is at the membrane resting potential in the absence of input currents. This indicates that the estimated parameters are appropriate to characterize the dynamical behavior of both neurons. The fourth plot shows the estimated input currents. The last plot shows the percentage error between the experimental and the model predicted spikes time. This error was computed as $\frac{100|(T_j^e - T_j^m)|}{T_j^e}$ where $T_j^e$ and $T_j^m$ are the experimental and the model predicted spike time for the $j^{th}$ action potential.

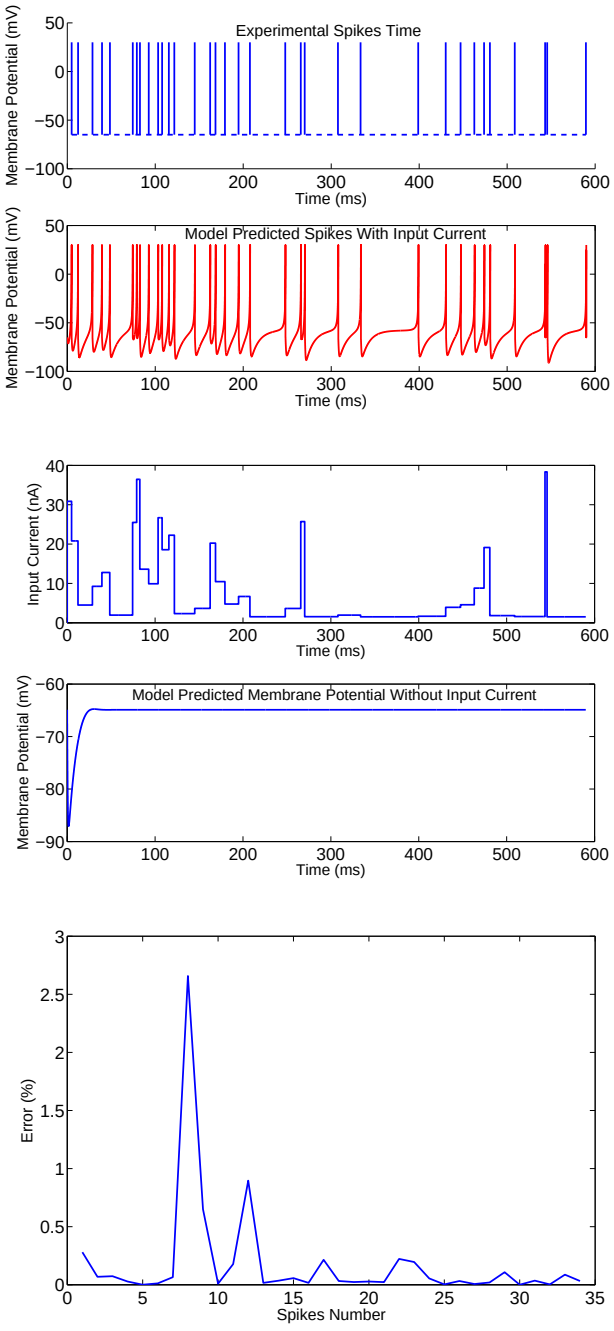



Figure 2.3: Constrained parameter estimation for the neuron "K11404".

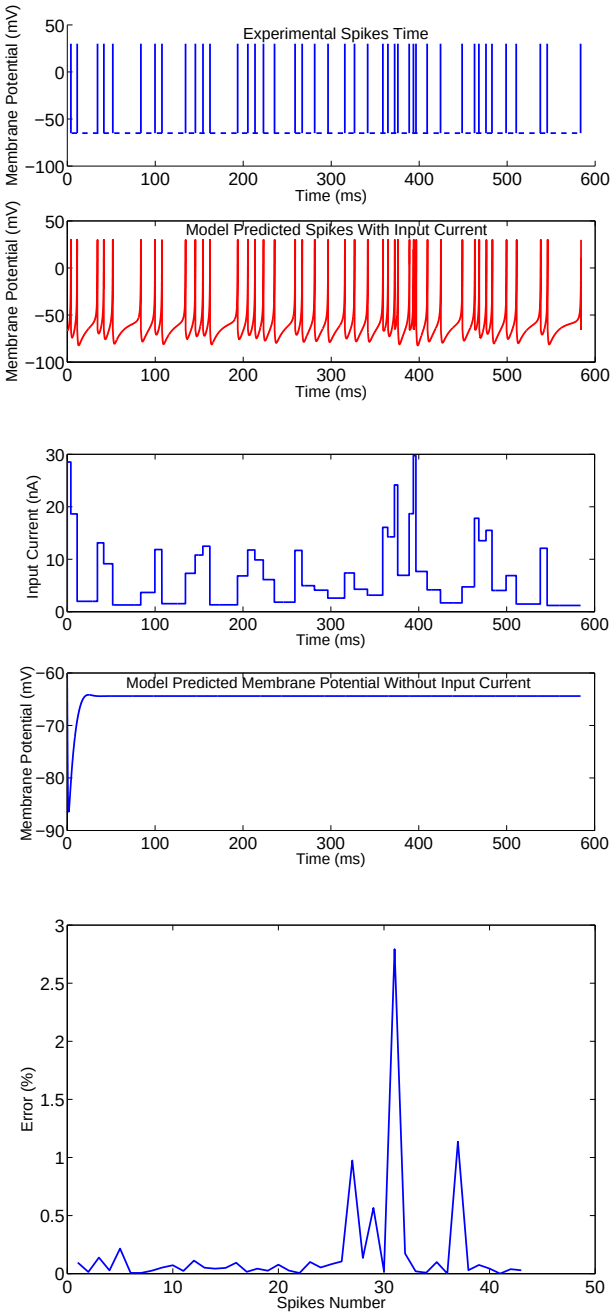



Figure 2.4: Constrained parameter estimation for the neuron "K15906".

To support our simulation results, we performed mathematical analysis of the Izhikevich model for both neurons, "K11404" and "K15906", using the estimated model parameters along with $I(t) = 0$. The computed equilibrium points for the neuron "K11404" i.e. $(-53.16, -12.76)$ and $(-65.84, -15.80)$ and the computed equilibrium points for the neuron "K15906" i.e. $(-53.77, -13.19)$ and $(-65.09, -15.98)$ clearly show that the model possesses real equilibrium states in the absence of any external input currents which was not true in the case of the unconstrained optimization problem. Each pair of equilibrium points represents $(v, u)$, the membrane potential and the membrane recovery variable respectively, at the equilibrium. $v = -65.84mV$ and $v = -53.16mV$ for the neuron "K11404" and $v = -65.09mV$ and $v = -53.77mV$ for the neuron "K15906" are the membrane resting potential and the neuron firing threshold respectively.

The computed eigenvalues at the membrane resting potential are $-0.25 \pm 0.23i$ for the neuron "K11404" and $-0.21 \pm 0.23i$ for the neuron "K15906". Here, $i = \sqrt{-1}$. These eigenvalues show that the membrane resting potential is a stable focus in both neurons. The existence of this stable focus supports experimental evidences that a neuron stays at its membrane resting potential in the absence of any input currents. The computed eigenvalues at the neuron firing threshold are 0.69 and $-0.17$ for the neuron "K11404" and 0.64 and $-0.15$ for the neuron "K15906". These eigenvalues clearly suggests the existence of a saddle at the firing threshold in both neurons.

Mathematical analysis of the Izhikevich model using estimated parameters for neurons "K11404" and "K15906" supports the experimentally observed dynamical behaviors of spiking neurons. Based on the parameter estimation results from other M1 neurons, we have found that this approach can be used to estimate parameters

of the Izhikevich single neuron model when the only available information is the inter-spike intervals (ISIs) data.

## 2.4 Parameter Estimation: Model II

In this section, we estimate unknown parameters of "Model II" using a benchmark set of experimental data available from "quantitative single-neuron modeling competition 2009" [114]. Figure 2.5 shows a framework for analyzing the quantitative predictive capability of the stochastic Izhikevich single neuron model in explaining experimental data.

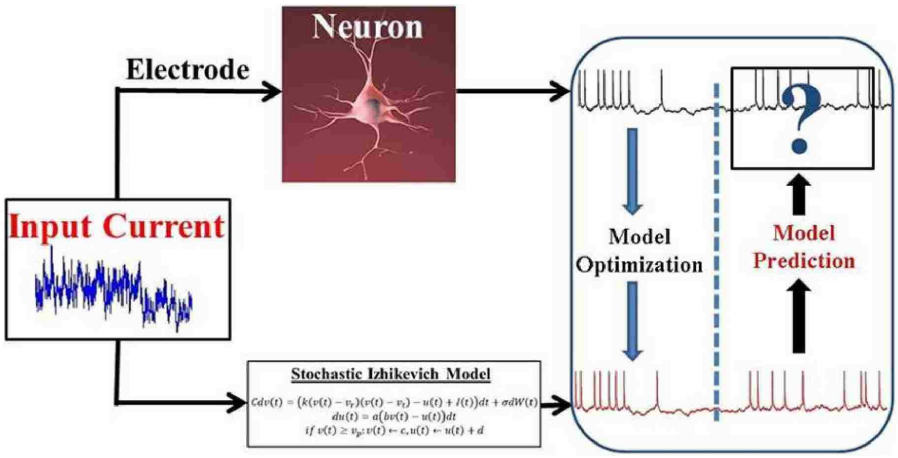



Figure 2.5: A framework for quantitative assessment of the stochastic Izhikevich single neuron model. The diagram has been modified and redrawn from Gerstner et al. [57].

As shown in this figure, an input current is injected to the soma of a single neuron. As a result, the neuron fires a sequence of action potentials (also called spike train). The time of the occurrence of action potentials are recorded to generate an experimental data set. The same input current is then used to simulate the

stochastic Izhikevich single neuron model. A part of the experimental data is used to optimize the model parameters. The remaining data is then used to validate the model predictability against the experimental data.

### 2.4.1 Problem Statement

In this section, we consider both deterministic ($\sigma = 0$) and stochastic ($\sigma > 0$) form of the Izhikevich single neuron model (see equation (2.3)) and formulate two problems in the framework shown in Figure 2.5 for analyzing the quantitative predictive capability of the model in explaining experimental data.

#### Problem I

It is clear that equation (2.3) is a two dimensional stochastic differential equation for $\sigma > 0$. Therefore, the time at which an action potential occurs (also known as the first passage time of the process) is a stochastic process. We define $\tau_i$, the time of occurrence of the $i^{th}$ action potential starting from $t = 0$ or the first passage time after the occurrence of the $(i-1)^{th}$ action potential, as

$$\tau_i = \inf\{t : v(t) \geq v_p \mid v(\tau_{i-1}) = c, u(\tau_{i-1}) = u(\tau_{i-1}) + d, t > \tau_{i-1}\}. \tag{2.15}$$

Here, $\tau_0 = 0$. We formulate the following optimization problem using the maximum likelihood framework to estimate parameters of the stochastic Izhikevich single neuron model given by equation (2.3):

$$\max_{a,b,c,d,v_r,v_t,v_p,k,C,\sigma} \sum_{i=1}^{n} \log(p_i(\tau_i \mid \tau_k, k = 1, \cdots, i-1)). \tag{2.16}$$

Here, $n$ is the total number of spikes in a given experimental spike train. In words, given an experimentally observed sequence of the first passage times $\{\tau_1, \tau_2, \cdots, \tau_n\}$, estimate the unknown parameters $a, b, c, d, C, k, v_r, v_t, v_p, \sigma$ optimally such that the joint probability of the occurrence of this sequence is maximum over these parameters. $p_i(\tau_i \mid \tau_k, k = 1, \cdots, i-1)$ is a conditional first passage time probability density given by

$$p_i(\tau_i \mid \tau_k, k = 1, \cdots, i-1) = -[\frac{d}{dt} \int_{-v_m}^{v_p} \int_{-u_{min}}^{u_{max}} P(t, v, u) du dv]_{t=\tau_i}. \tag{2.17}$$

Here, $P(t, v, u)$ is the joint probability density for the time evolution of the membrane potential $v(t)$ and the membrane recovery variable $u(t)$ and satisfies the following two-dimensional Fokker-Planck equation:

$$\frac{\partial P(t, v, u)}{\partial t} = -\frac{\partial J_v(t, v, u)}{\partial v} - \frac{\partial J_u(t, v, u)}{\partial u}, \tag{2.18}$$

with

$$P(t, v_p, u) = 0 \qquad \text{(Boundary condition)}, \tag{2.19a}$$

$$P(t, v_m, u) = 0 \qquad \text{(Boundary condition)}, \tag{2.19b}$$

$$P(\tau_{i-1}^{+} v, u) = \frac{1}{Z} \delta(v - c) R(-\frac{\partial J_v(\tau_{i-1}^{-}, v, u - d)}{\partial v} \mid_{v=v_p}) \qquad \text{(Initial condition)}. \tag{2.19c}$$

Here, $J_v(t, v, u) = \frac{1}{C}(k(v - v_r)(v - v_t) - u + I(t)))P(t, v, u) - \frac{\sigma^2}{2C^2} \frac{\partial P(t,v,u)}{\partial v}$ is the probability flux in the direction of $v$ and $J_u(t, v, u) = a(bv - u)P(t, v, u)$ is the probability flux in the direction of $u$. $\delta(\cdot)$ is the Dirac Delta function. $R(x) = x$ for $x > 0$,

otherwise 0. $Z$ is a normalization factor give by

$$Z = \int_{u_{min}}^{u_{max}} R(-\frac{\partial J_v(v, u-d, \tau_{i-1}^-)}{\partial v} \mid_{v=v_p}) \mathrm{d}u. \tag{2.20}$$

To compute the objective function in equation (2.16) for one iteration of numerical optimization, one requires the solution of equations (2.17), (2.18), (2.19) and (2.20) at each $i$ for $i = 1, 2, \cdots, n$. In the absence of a closed form solution of equations (2.17), (2.18), (2.19) and (2.20), solving these equations numerically at each $i$ for a large $n$ (say $n = 100$) are computationally expensive.

One way to reduce this computational complexity is to use a deterministic membrane recovery variable $u_d(t)$ in equation (2.3) for $\sigma > 0$, which can be computed using equation (2.3) with $\sigma = 0$ [39]. This leads to the following one dimensional approximation of the stochastic Izhikevich single neuron model:

$$C\mathrm{d}v(t) = (k(v(t) - v_r)(v(t) - v_t) - u_d(t) + I(t))\mathrm{d}t + \sigma \mathrm{d}W(t), \tag{2.21a}$$

$$C\mathrm{d}v_d(t) = (k(v_d(t) - v_r)(v_d(t) - v_t) - u_d(t) + I(t))\mathrm{d}t, \tag{2.21b}$$

$$\mathrm{d}u_d(t) = a(bv_d(t) - u_d(t))\mathrm{d}t, \tag{2.21c}$$

$$\begin{aligned} &if \quad v(t) \geq v_p, \quad then \\ &\quad v(t) \leftarrow c \quad and \quad u_d(t) \leftarrow u_d(t) + d, \end{aligned} \tag{2.21d}$$

$$\begin{aligned} &if \quad v_d(t) \geq v_p, \quad then \\ &\quad v_d(t) \leftarrow c. \end{aligned} \tag{2.21e}$$

Equation (2.21) approximates the dynamical evolution of the membrane potential

$v(t)$ given in equation (2.3) for an arbitrary small $\sigma > 0$ using one dimensional stochastic differential equation. Moreover, equation (2.21) reduces to equation (2.3) when $\sigma = 0$. The probability density $P(t, v)$ describes the time evolution of the membrane potential $v(t)$ in probability and satisfies the following one dimensional Fokker-Planck equation:

$$\frac{\partial P(t,v)}{\partial t} = -\frac{\partial((F(v) - \frac{1}{C}u_d(t) + \frac{1}{C}I(t))P(t,v))}{\partial v} + \frac{\sigma^2}{2C^2}\frac{\partial^2 P(t,v)}{\partial v^2}, \tag{2.22a}$$

Initial Condition:

$$P(t,v) \mid_{t=\tau_{i-1}} = \delta(v - c), \tag{2.22b}$$

Boundary Conditions:

$$P(t,v) \mid_{v=v_m} = 0, \tag{2.22c}$$

$$\frac{\partial P(t,v)}{\partial t} \mid_{v=v_p} = -\frac{\partial((F(v) - \frac{1}{C}u_d(t) + \frac{1}{C}I(t))P(t,v))}{\partial v} \mid_{v=v_p} . \tag{2.22d}$$

Here, $F(v) = \frac{k}{C}(v - v_r)(v - v_t)$. Equation (2.22c) represents an absorbing boundary condition at the left boundary and (2.22d) represents a no diffusion boundary condition at the right boundary of the $v$-space. We modify the first passage time probability density in equation (2.17) as

$$p_i(\tau_i \mid \tau_k, k = 1, \cdots, i-1) = -[\frac{d}{dt}\int_{v_m}^{v_p} P(t,v)dv]_{t=\tau_i}. \tag{2.23}$$

Equations (2.16), (2.21), (2.22) and (2.23) define the parameter estimation problem

for the stochastic Izhikevich single neuron model in the maximum likelihood framework.

To solve the one dimensional Fokker-Planck equation given by equation (2.22) numerically, we implemented the method of lines [143] with a 5-point upwind scheme in MATLAB. For this, we approximated the Dirac Delta function given in the initial condition defined by equation (2.22b) by a Gaussian distributed function and redefined the initial condition as

$$P(t,v)\mid_{t=\tau_{i-1}}= \frac{1}{5\sqrt{(2\pi)}}\exp(\frac{-(v-c)^2}{50}). \tag{2.24}$$

We discretized the $v$ space by defining $\Delta v = (v_p - v_m)/(N-1)$ and $v_j = v_m + (j-1)\Delta v$ for $j = 1, 2, \cdots, N$. At each $j$, we defined $P_j(t) = P(t, v_j)$. With this, we approximated the first derivative $\frac{\partial P_j(t)}{\partial v}$ by a 5-point upwind scheme as follows: When $F(v_j) - \frac{1}{C}u_d(t) + \frac{1}{C}I(t) > 0$,

$$\frac{\partial P_j(t)}{\partial v} = \frac{3P_{j+1} + 10P_j - 18P_{j-1} + 6P_{j-2} - P_{j-3}}{12\Delta v} \qquad for\ \ j = 3, 4, \cdots, N-1, \tag{2.25a}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{-25P_j + 48P_{j+1} - 36P_{j+2} + 16P_{j+3} - 3P_{j+4}}{12\Delta v} \qquad for\ \ j = 1, \tag{2.25b}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{-3P_{j-1} - 10P_j + 18P_{j+1} - 6P_{j+2} + P_{j+3}}{12\Delta v} \qquad for\ \ j = 2, \tag{2.25c}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{P_{j-2} - 8P_{j-1} + 0P_j + 8P_{j+1} - P_{j+2}}{12\Delta v} \qquad for\ \ j = 3, \tag{2.25d}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{3P_{j-4} - 16P_{j-3} + 36P_{j-2} - 48P_{j-1} + 25P_j}{12\Delta v} \qquad for\ \ j = N. \tag{2.25e}$$

When $F(v_j) - \frac{1}{C}u_d(t) + \frac{1}{C}I(t) < 0$,

$$\frac{\partial P_j(t)}{\partial v} = \frac{-3P_{j-1} - 10P_j + 18P_{j+1} - 6P_{j+2} + P_{j+3}}{12\Delta v} \qquad for\ \ j = 2, 3, \cdots, N-2, \tag{2.26a}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{-25P_j + 48P_{j+1} - 36P_{j+2} + 16P_{j+3} - 3P_{j+4}}{12\Delta v} \qquad for\ \ j = 1, \tag{2.26b}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{P_{j-2} - 8P_{j-1} + 0P_j + 8P_{j+1} - P_{j+2}}{12\Delta v} \qquad for\ \ j = N-2, \tag{2.26c}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{3P_{j+1} + 10P_j - 18P_{j-1} + 6P_{j-2} - P_{j-3}}{12\Delta v} \qquad for\ \ j = N-1, \tag{2.26d}$$

$$\frac{\partial P_j(t)}{\partial v} = \frac{3P_{j-4} - 16P_{j-3} + 36P_{j-2} - 48P_{j-1} + 25P_j}{12\Delta v} \qquad for\ \ j = N. \tag{2.26e}$$

We approximated the second order derivative $\frac{\partial^2 P_j(t)}{\partial v^2}$ by the standard second order central finite difference scheme. For $j = 2, 3, \cdots, N-1$, equation (2.22a) was discretized as

$$\frac{\mathrm{d}P_j}{\mathrm{d}t} = -(F(v_j) - \frac{1}{C}u_d(t) + \frac{1}{C}I(t))\frac{\partial P_j}{\partial v} - \frac{\partial F(v_j)}{\partial v}P_j + \frac{\sigma^2}{2}\frac{\partial^2 P_j}{\partial v^2}. \tag{2.27}$$

For $j = 1$, we implemented the boundary condition given by equation (2.22c) as

$$\frac{\mathrm{d}P_1}{\mathrm{d}t} = 0. \tag{2.28}$$

For $j = N$, we implemented the boundary condition given by equation (2.22d) as

$$\frac{\mathrm{d}P_N}{\mathrm{d}t} = -(F(v_N) - \frac{1}{C}u_d(t) + \frac{1}{C}I(t))\frac{\partial P_N}{\partial v} - \frac{\partial F(v_N)}{\partial v}P_j. \tag{2.29}$$

For solving these $N$ set of ordinary differential equation at each time $t$, we used MATLAB ODE solver "ode15s". We set $N = 201$. To compute the first passage time

given by equation (2.23), we used MATLAB numerical integration function "trapz".

Figure 2.6 shows a numerical solution of equations (2.22) and (2.23) with $i = 1$ for a constant and a time varying input current $I(t)$. Here, we used $C = 100$, $k = 0.07$, $v_t = -40$, $v_r = -60$, $a = 0.003$, $b = -2$, $c = -50$, $d = 100$, $v_p = 35$, $\sigma = 1000$.

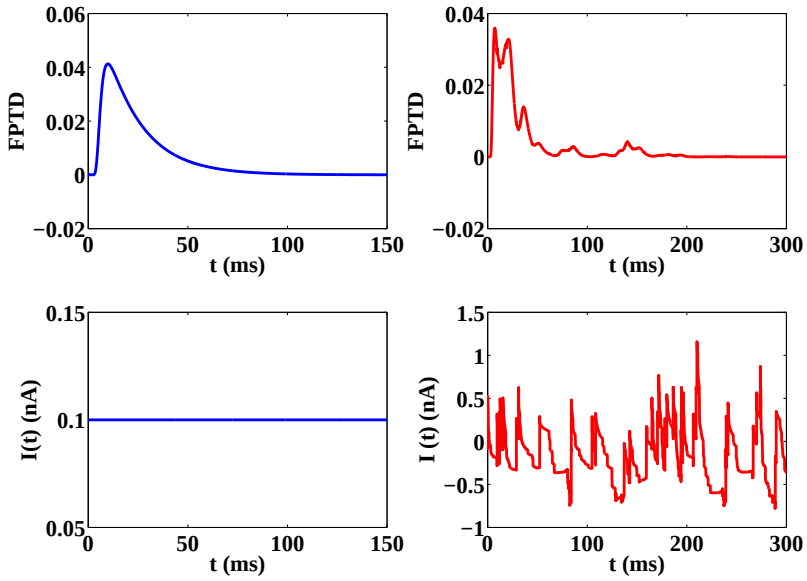



Figure 2.6: Numerical solution of the first passage time density function. Here, "FPTD" stands for first passage time density and represents $p_1(t)$. $I(t)$ is the net current to the neuron at time $t$.

### Problem II

As an alternative to the maximum likelihood estimation approach described in the previous section, here we formulate the following optimization problem to estimate unknown parameters of the one dimensional approximation of "Model II" given by

equation (2.21):

$$\max_{a,b,c,d,v_r,v_t,v_p,k,C,\sigma} \frac{1}{n}\frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m}\Gamma_{i,j} + \frac{1}{n}\sum_{i=1}^{n}(1 - \frac{|\max_j N^j_{data} - N^i_{model}|}{\max_j N^j_{data}}). \tag{2.30}$$

Here, $\Gamma_{i,j}$ is the coincident factor between two spike trains [83] defined as

$$\Gamma_{i,j} = \frac{N^{i,j}_{coinc} - < N^{i,j}_{coinc} >}{(N^{i,j}/2)(N^j_{data} + N^i_{model})}. \tag{2.31}$$

$n$ is the number of realizations of spike train generated by the model. $m$ is the total number of spike trains in the experimental data set. $N^j_{data}$ is the number of spikes (action potentials) in the $j^{th}$ spike train of the experimental data set in a given time interval. $N^i_{model}$ is the number of spikes (action potentials) in the $i^{th}$ realization of the spike train predicted by the model in the same time interval. $N^{i,j}_{coinc}$ is the number of coincidences of spikes with precision $\Delta = 4$ ms between the $j^{th}$ spike train of the experimental data set and the $i^{th}$ realization of the spike train predicted by the model. $< N^{i,j}_{coinc} >= 2f_i\Delta N^j_{data}$ is the expected number of coincidences generated by a homogeneous Poisson process with the same firing rate $f_i$ as the model. $N^{i,j} = 1 - 2f_i\Delta$ is a normalization factor.

Equation (2.30) maximizes the average coincidence factor defined by the first term while keeping the average number of spikes predicted by the model in a given time interval close to the maximum number of spikes in the experimental spike trains of the same duration. When the model is deterministic i.e. $\sigma = 0$, $n = 1$ and the maximization in equation (2.30) is over $a, b, c, d, v_r, v_t, v_p, k, C$. Moreover for $\sigma = 0$, the modified form of "Model II" given by equation (2.21) is same as the "Model II" i.e. equation (2.3).

The average coincidence factor is defined as $\Gamma = \frac{1}{n}\frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m}\Gamma_{i,j}$ with $\Gamma \in [0,1]$. $\Gamma = 0$ means that the model is not better than if the spike trains were drawn from a homogeneous Poisson process with mean firing rate same as the model. On the other hand, $\Gamma = 1$ means that the model is optimal [114].

### 2.4.2 Results

In this section, we present our results on the efficacy of the modified Izhikevich single neuron model given by (2.21) in explaining experimental data by solving "Problem I" and "Problem II" numerically. In "Problem I", we used MATLAB based local optimization algorithm "fminsearch" to solve equation (2.16) for the modified stochastic Izhikevich single neuron model given by equation (2.21). The numerical solution of the one dimensional Fokker-Planck equation (equations (2.21) and (2.22)) was obtained by solving equations (2.24) - (2.29) in MATLAB. In "Problem II", we used a direct optimization algorithm MATLAB routine [84, 50] to solve equation (2.30). All the computations were performed on Lehigh high performance computing cluster machines.

To estimate the model parameters in both "Problem I" and "Problem II", we used the experimental data from "quantitative single-neuron modeling competition 2009: Challenge A" [114]. These data were obtained from a regular spiking L5 pyramidal neuron in *in vivo* like conditions. Details of the experimental protocol and the method applied to record data are discussed in [114]. Figure 2.7 shows a part of the injected input current to the regular spiking L5 pyramidal neuron and the recorded spike trains for two repeated trials with the same input current.

The data set was consisted of 13 independent spike trains, each of 38 seconds (s)

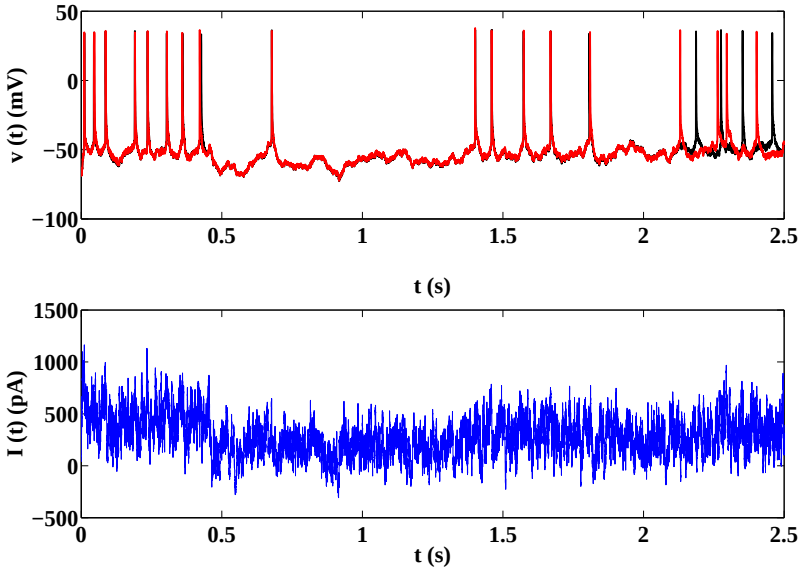



Figure 2.7: Recording of spike trains from a regular spiking L5 pyramidal neuron in response to *in vivo* like current injection [114].

duration, obtained from the regular spiking L5 pyramidal neuron in response to the same (deterministic) input current. For estimating parameters in "Problem I", we used the data from only 11 spike trains. For estimating parameters in "Problem II", we used the data from all the 13 spike trains. In both "Problem I" and "Problem II", we used the data recorded in each independent trials during 17.5 s and 28 s (10.5 s duration) to optimize the model parameters (training data) and the data recorded in each independent trials during 28 s and 38 s (10 s duration) to compare the model performance against the experimental data (performance data). The model performance on the training data and the performance data was computed using the coincidence factor defined by equation (2.31) and

$$\Gamma = \mathbb{E}\Gamma \pm SD(\Gamma). \tag{2.32}$$

Here, $\mathbb{E}\Gamma = \frac{1}{n}\frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m}\Gamma_{i,j}$ and $SD(\Gamma) = \sqrt{\frac{1}{m}\sum_{j=1}^{m}(\frac{1}{n}\sum_{i=1}^{n}\Gamma_{i,j} - \mathbb{E}\Gamma)^2}$.

**Problem I**

In this section, we studied the parameter estimation problem in the maximum likelihood framework described in "Problem I" of section 2.4.1 and investigated the predictive capability of the modified stochastic Izhikevich single neuron model (equation (2.21)) using the optimized parameters obtained from this framework. Before optimizing the model parameters using experimental data, we validated the maximum likelihood approach used in estimating model parameters against a synthetic data set.

To generate a synthetic data set, we simulated equation (2.21) with $C = 100$, $k = 0.07$, $v_t = -40$, $v_r = -60$, $a = 0.003$, $b = -2$, $c = -50$, $d = 100$, $v_p = 35$, $\sigma = 1000$. Figure 2.8 shows the input current $I(t)$ used to simulate equation (2.21).

We generated 52 independent spike trains, each consisting of $322 \pm 7$ spikes over a period of 39 seconds. We fixed $v_p = 35$ and $\sigma = 1000$. and used the first 200 spikes from each spike trains to optimize the remaining 8 model parameters $C, k, v_t, v_r, a, b, c, d$ using the maximum likelihood approach. For this, we solved the maximum likelihood optimization problem defined by equation (2.16) for each spike trains and obtained 52 estimated values of the same parameter. Using the mean values of the optimized parameters, we validated the efficacy of the model in explaining the remaining spikes in each spike trains. Table 2.1 shows the optimized parameters using the synthetic data set.

It is clear from Table 2.1 that the mean values of parameters estimated using the maximum likelihood approach are significantly different from the values of parameters

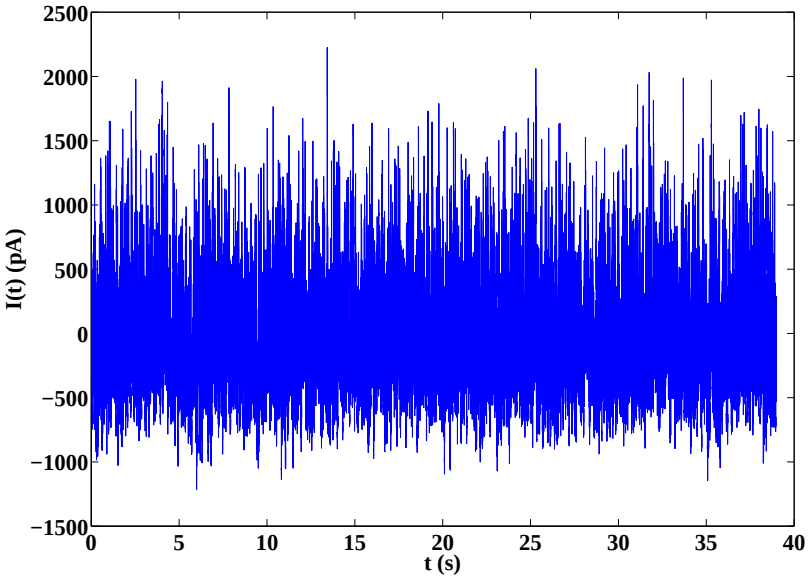



Figure 2.8: Input current $I(t)$ used for generating the synthetic data set.

| Parameter | True | Guess | Estimated (mean $\pm$ standard deviation) |
|---|---|---|---|
| $C$ | 100.00 | 80.00 | $66.06 \pm 14.50$ |
| $k$ | 0.70 | 0.50 | $0.50 \pm 0.38$ |
| $v_t$ | $-40.00$ | $-45.00$ | $-44.46 \pm 8.62$ |
| $v_r$ | $-60.00$ | $-65.00$ | $-88.54 \pm 8.95$ |
| $a$ | 0.0030 | 0.0025 | $0.0036 \pm 0.0023$ |
| $b$ | $-2.00$ | $-4.00$ | $-6.59 \pm 3.29$ |
| $c$ | $-50.00$ | $-55.00$ | $-67.68 \pm 12.05$ |
| $d$ | 100.00 | 80.00 | $48.28 \pm 20.15$ |

Table 2.1: Estimated parameters of the modified stochastic Izhikevich single neuron model (equation (2.21)) for "Problem I" using a synthetic data set. Here, the mean value of a given parameter was computed by taking the average of its estimated value over 52 trials. The standard deviation of a given parameter was computed by taking the square root of its variance over 52 trials.

used to generate the data ("True"). One of the reason for this may be the limited data (52 spike trains with 200 spikes in each) used in estimating parameters. In the maximum likelihood approach, the mean estimated values of parameters match the true parameters of the model if the number of data used in estimating parameters are large enough (theoretically infinite). Since our data set was small, we wondered if the average coincidence factor (a benchmark measure of a single neuron model performance) computed using the "True" parameters are closed to the one computed using the mean values of the estimated parameters. For this, we computed $\mathbb{E}\Gamma$ over the training data set (consists of 200 spikes from each spike trains) and the performance data set (the remaining spikes,approximately 100, from each spike trains) using equation (2.32) with $n = 16000$ and $m = 52$. On the training data set, the computed $\mathbb{E}\Gamma$ with "True" parameters was 0.2748 and with the mean values of the estimated parameters was 0.2965. On the performance data set, the computed $\mathbb{E}\Gamma$ with "True" parameters was 0.4985 and with the mean values of the estimated parameters was 0.5381. Clearly, the difference between the computed coincident factor with "True" parameters and the mean values of the estimated parameters is small over both the training and the performance data set. These results suggest that the maximum likelihood approach can be used to estimate parameters of the modified stochastic Izhikevich single neuron model given by equation (2.21) using a small data set.

Next, we optimized the parameters of the model given by equation (2.21) using a part of the experimental data in the maximum likelihood framework. Here, we optimized all 10 parameters $C, k, v_t, v_r, v_p a, b, c, d, \sigma$. For this, we solved the maximum likelihood optimization problem defined by equation (2.16) for 11 spike trains and obtained 11 estimated values of the same parameter. To initialized the optimization,

we computed the initial values ("Guess") of parameters by matching the average firing rate in an experimental spike train with the average firing rate predicted by model in the same time duration. Moreover, we fixed the non-positive value, if occurred, of the first passage time by assigning it a value of $10^{-15}$ during the optimization. Table 2.2 shows the optimized parameters using the experimental data set.

| Parameter | Guess | Estimated (mean $\pm$ standard deviation) |
|---|---|---|
| $C$ | 94.52 | $237.78 \pm 24.46$ |
| $k$ | 0.84 | $0.66 \pm 0.03$ |
| $v_p$ | 34.77 | $44.91 \pm 7.29$ |
| $v_t$ | $-26.37$ | $-19.05 \pm 1.82$ |
| $v_r$ | $-72.20$ | $-73.89 \pm 1.09$ |
| $a$ | 0.03 | $0.04 \pm 0.01$ |
| $b$ | $-1.84$ | $-1.26 \pm 0.26$ |
| $c$ | $-58.01$ | $-123.36 \pm 21.16$ |
| $d$ | 118.72 | $126.67 \pm 24.07$ |
| $\sigma$ | 500.00 | $1414.10 \pm 153.39$ |

Table 2.2: Estimated parameters of the modified stochastic Izhikevich single neuron model (equation (2.21)) for "Problem I" using a part of the experimental data set. Here, the mean value of a given parameter was computed by taking the average of its estimated value over 11 trials. The standard deviation of a given parameter was computed by taking the square root of its variance over 11 trials.

Using the mean values of the estimated parameters, we computed $\mathbb{E}\Gamma$ over the training data set and the performance data set using equation (2.32) with $n = 10000$ and $m = 11$. On the training data set, the computed $\mathbb{E}\Gamma$ with the mean values of the estimated parameters was 0.0364. On the performance data set, the computed $\mathbb{E}\Gamma$ with the mean values of the estimated parameters was 0.0427. These results clearly show that the modified stochastic Izhikevich single neuron model (equation (2.21)) performs poor on the experimental data set with the optimized parameters obtained

through the maximum likelihood approach.

### Problem II

In this section, we studied the parameter estimation problem using the approach of maximizing the average coincident factor described in "Problem II" of section 2.4.1. We investigated the predictive capability of the deterministic Izhikevich single neuron model (equation (2.3 with $\sigma = 0$) and the modified stochastic Izhikevich single neuron model (equation (2.21)) using the optimized parameters obtained through this approach.

We first considered the deterministic Izhikevich single neuron model and optimized 9 parameters of this model. Table 2.3 shows the optimized value of these parameters.

| Parameter | Minimum bound | Maximum bound | Optimum |
|---|---|---|---|
| $C$ | 10 | 400 | 75.00 |
| $k$ | 0.001 | 5 | 2.50 |
| $v_p$ | 0 | 60 | 0.37 |
| $v_t$ | $-50$ | $-20$ | $-45.00$ |
| $v_r$ | $-100$ | $-50$ | $-75$ |
| $a$ | 0.0001 | 5 | 0.83 |
| $b$ | $-10$ | $-0.01$ | $-1.68$ |
| $c$ | $-100$ | $-50$ | $-75$ |
| $d$ | 0 | 400 | 330.04 |

Table 2.3: Estimated parameters of equation (2.3) with $\sigma = 0$ for "Problem II".

Using the estimated value of parameters given in Table 2.3, we computed $\Gamma$ for the training data set and the performance data set using equation (2.32) with $n = 1$ and $m = 13$. On the training data set, the computed $\Gamma$ was $0.40 \pm 0.06$. The average number of spikes in the training data was 120 and the number of spikes predicted by the model was 124. On the performance data set, the computed $\Gamma$ was $0.38 \pm 0.09$.

The average number of spikes in the performance data was 80 and the number of spikes predicted by the model was 66. These results are consistent with the reported study in literature where 6 parameters were optimized for a modified form of "Model I" [137].

Clearly, the deterministic Izhikevich single neuron model (equation (2.3) with $\sigma = 0$) with estimated parameters using the coincidence factor approach shows a better performance in predicting experimental data compared to the modified stochastic model (equation (2.21)) with estimated parameters using the maximum likelihood approach. We wondered if this approach can also provide at least the similar performance of the modified stochastic model (equation (2.21)) in predicting experimental data.

Next we optimized 10 parameters of the modified stochastic Izhikevich single neuron model (equation (2.21) with $\sigma > 0$) including $\sigma$ and investigated the capability of the model in predicting experimental data. Table 2.4 shows the obtained value of these optimized parameters.

| Parameter | Minimum bound | Maximum bound | Optimum |
|---|---|---|---|
| $C$ | 10 | 400 | 75.00 |
| $k$ | 0.001 | 5 | 2.50 |
| $v_p$ | 0 | 60 | 3.33 |
| $v_t$ | $-50$ | $-20$ | $-45.00$ |
| $v_r$ | $-100$ | $-50$ | $-75$ |
| $a$ | 0.0001 | 5 | 2.50 |
| $b$ | $-10$ | $-0.01$ | $-1.68$ |
| $c$ | $-100$ | $-50$ | $-91.67$ |
| $d$ | 0 | 400 | 66.67 |
| $\sigma$ | 0.5 | 100 | 6.03 |

Table 2.4: Estimated parameters of equation (2.21) with $\sigma > 0$ for "Problem II".

Using the estimated value of parameters given in Table 2.4, we computed $\Gamma$ for the training data set and the performance data set using equation (2.32) with $n = 100$ and $m = 13$. On the training data set, the computed $\Gamma$ was $0.40 \pm 0.02$. The average number of spikes in the training data was 120 and the average number of spikes predicted by the model was 126. On the performance data set, the computed $\Gamma$ was $0.39 \pm 0.02$. The average number of spikes in the performance data was 80 and the average number of spikes predicted by the model was 66. These results clearly shows that the performance of the modified stochastic Izhikevich model (equation (2.21)) on experimental data is comparable to the deterministic model (equation (2.3) with $\sigma = 0$) using this approach. Moreover, the model with estimated parameters using this approach performs much better than the one with the maximum likelihood approach.

## 2.5 Concluding Remarks

In this chapter, we have investigated the estimation of parameters of both the deterministic and the stochastic Izhikevich single neuron model using experimentally obtained inter-spike intervals (ISIs) of single neurons. For a dimensionless form of the deterministic Izhikevich model ("Model I"), 4 parameters as well as the input current to the neuron have been estimated using experimentally recorded single neuron ISI data from a primate study. Non-linear constrained and unconstrained optimization problems have been formulated to characterize the neuron physical properties appropriately. Numerical results have been presented for two cortical neurons ("K11404" and "K15906"). Need for the constrained optimization problem has been emphasized and discussed through a rigorous mathematical analysis. We believe that this

approach can be used to efficiently estimate parameters of the single neuron models when the only available information is the experimentally obtained ISIs.

For the conductance-based detailed Izhikevich model ("Model II"), we have estimated 10 parameters of the stochastic Izhikevich single neuron model using the approach of maximizing the likelihood of the joint probability distribution of the occurrence of an experimentally observed sequence of action potentials in a benchmark set of experimental data available from "quantitative single-neuron modeling competition 2009" [114]. The data set included the input current delivered to a regular spiking L5 pyramidal neuron and the recorded ISIs in response to the input current for 13 repeated trials. For computational efficiency, the two dimensional stochastic Izhikevich model has been approximated by a one dimensional stochastic model coupled with the deterministic Izhikevich model. One dimensional Fokker-Planck equation has been derived to compute the first passage time density. We have found that the model with estimated parameters using this approach shows poor predictability in explaining experimental data (approximately 4% of predictability). We have also found that one of the possible reasons for this poor performance is the large value of the noise term $\sigma$ ($1414.10 \pm 153.39$) estimated by the maximum likelihood method (since the optimization was unconstrained) which makes the predicted spike trains by the model extremely unreliable. This suggests further investigations of the maximum likelihood estimation approach by formulating constrained optimization problems.

Finally, we compared the results obtained using the maximum likelihood approach with the approach of maximizing the average coincident factor between the experimental and the model predicted spike trains. For this, we estimated 8 and 10 parameters of the deterministic and the one dimensional approximation of the stochastic

Izhikevich single neuron model respectively using the approach of coincident factor. We have found that both form of the model can explain approximately 38% of the experimental data using the parameters estimated using this approach.

In conclusion, both the deterministic and the one dimensional approximation of the stochastic Izhikevich single neuron model with estimated parameters using the coincident factor approach shows better predictability of these models in explaining experimental data as opposed to the maximum likelihood estimation approach.

# Chapter 3

# Investigation on Continuity and Differentiability of Inter-Spike Intervals

## 3.1 Motivation

As discussed in Chapter 1 (see Figure 1.2 in Chapter 1), the receding horizon controller designs optimal artificial sensory feedback currents and stimulates neurons of the appropriate cortical sensory areas such that the closed-loop performance of the BMI can be recovered for a given motor task. Briefly, the controller computes optimal stimulating input currents $I^E(t;\boldsymbol{\theta})$ by solving an optimal control problem in a predictive framework [95]. Here $\boldsymbol{\theta}$ is a set of parameters or decision variables. At a given time, system measurements are obtained and a dynamic model of the cortical network of spiking neurons, decoder, and the prosthetic arm is used to predict the outputs of the system. An optimal control problem is then formulated to minimize

the difference between the predicted and the desired outputs. The design of artificial sensory feedback currents is constrained by experimentally observed minimum and maximum limit on the instantaneous firing rate or the corresponding ISI of neurons in the cortical network model as well as any constraints on the system outputs. These constraints appear explicitly in the resultant nonlinear optimization problem. A local gradient-based optimization algorithm for solving the optimization problem requires at least the first-order continuous differentiability of the objective and constraint functions with respect to (w.r.t.) decision variables. The reset of state variables at the occurrence of each action potential in typical single neuron models such as the Izhikevich model [78] makes it nontrivial to determine the continuous differentiability of ISIs.

ISIs and their gradients have also been used in developing gradient-based learning rules for training spiking neural networks (for example see [127, 105] and references there in). In most of these works, the Leaky Integrate-and-Fire model or the theta neuron model has been used where analytical solution for the error gradient has been derived. Extension of these learning rules to models such as the Izhikevich neuron model [78] require investigation of differentiability of ISIs w.r.t. the synaptic weights since analytical solutions in these models are difficult to obtain.

With this motivation, in this chapter, we consider a recurrent network of $n$ interconnected neurons. We assume that each neuron is connected with the remaining $n-1$ neurons. The dynamical behavior of each neuron is represented by a class of discrete event-based discontinuous single neuron models. The flow of synaptic information from one neuron to another is modeled using a synaptic conductance model. Stimulation of an input neuron using an external input current drives the network.

Measured inter-spike intervals of a single output neuron defines the output of the network. In this set-up, we derive sufficient conditions under which the first-order continuous differentiability of ISIs of this output neuron w.r.t. parameters (decision variables) of the external stimulating current can be guaranteed.

## 3.2 Mathematical Tools

In this chapter, we make use of the following well known existence and uniqueness theorems for ordinary differential equations (Theorems 3.2.1 and 3.2.2) and the implicit function theorem (Theorem 3.2.3) in deriving our results.

**Theorem 3.2.1.** *[27] Suppose $D$ is a domain in the $(t, x)$ space. $I_\mu$ be the domain of $\mu$ space, i.e. $|\mu - \mu_0| < c$ for $c > 0$ where $\mu_0$ is fixed. $D_\mu$ is defined as $D_\mu := (t, x) \in D, \mu \in I_\mu$. Let $f \in \mathcal{C}$ on $D_\mu$ and satisfy a Lipschitz condition in $x$ uniformly on $D_\mu$. For a given fixed $\mu = \mu_0$, let $\psi$ be the solution of*

$$\dot{x} = f(t, x, \mu) \tag{3.1}$$

*on an interval $a \leq t \leq b$. There exists a $\delta > 0$ such that for any $(\tau, \xi, \mu) \in U_\mu$ where $U_\mu := \{(\tau, \xi, \mu) \mid a < \tau < b, |\xi - \psi(\tau)| + |\mu - \mu_0| < \delta\}$, there exists a unique solution $\phi$ of equation (3.1) on $a \leq t \leq b$ satisfying $\phi(\tau, \tau, \xi, \mu) = \xi$. Moreover, $\phi \in \mathcal{C}$ on $n + k + 2$ dimensional domain $V_\mu := \{(t, \tau, \xi, \mu) \mid a < t < b, (\tau, \xi, \mu) \in U_\mu\}$.*

**Theorem 3.2.2.** *[27] Let the hypothesis of Theorem 3.2.1 be satisfied. Suppose that $f_x \in \mathcal{C}$, $f_\mu \in \mathcal{C}$ on $D_\mu$. Then the solution defined in Theorem 3.2.1 is of class $\mathcal{C}^1$ on $V_\mu$.*

**Theorem 3.2.3.** *[8] Let* $\mathbf{f} = (f_1, f_2, \cdots, f_n)$ *be a vector-valued function defined on an open set* $S$ *in* $E_{n+k}$ *with values in* $E_n$*. Suppose* $\mathbf{f} \in \mathcal{C}^1$ *on* $S$*. Let* $(\mathbf{x_0}; \mathbf{t_0})$ *be a point in* $S$ *for which* $\mathbf{f}(\mathbf{x_0}; \mathbf{t_0}) = \mathbf{0}$ *and for which the* $n \times n$ *determinant* $det[D_j f_i(\mathbf{x_0}; \mathbf{t_0})] \neq 0$*. Then there exists a* $k-$ *dimensional neighborhood* $T_0$ *of* $\mathbf{t_0}$ *and one, and only one, vector-valued function* $\mathbf{g}$*, defined on* $T_0$ *and having values in* $E_n$*, such that*

1. $\mathbf{g} \in \mathcal{C}^1$ *on* $T_0$*,*
2. $\mathbf{g}(\mathbf{t_0}) = \mathbf{x_0}$*,*
3. $\mathbf{f}(\mathbf{g}(\mathbf{t}); \mathbf{t}) = \mathbf{0}$ *for every* $\mathbf{t} \in T_0$*.*

## 3.3 Problem Statement

Let the dynamics of a single neuron is given by the following dynamical equations:

$$C\frac{\mathrm{d}v(t)}{\mathrm{d}t} = f(v(t), u(t)) + I(t), \tag{3.2a}$$

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} = g(v(t), u(t)), \tag{3.2b}$$

$$\begin{aligned} &if \quad v(t) \geq v_p(t), \quad then \\ &\quad v(t) \leftarrow c \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{3.2c}$$

Here, $v(t)$ and $u(t)$ are the time-varying membrane potential and the membrane recovery variable of a neuron respectively. $C$ is the membrane capacitance. $I(t)$ is the total input current delivered to the neuron. $v_p(t)$ is a firing threshold. $c$ and $d$ are the model parameters.

We define $t^f$, the time of occurrence of the $f^{th}$ action potential starting from $t = 0$, as

$$t^f = \{t : v(t) = v_p(t) \mid v(t^{f-1}) = c, u(t^{f-1}) = u(t^{f-1}) + d\}. \tag{3.3}$$

An inter-spike interval (ISI) is defined as the time difference between the occurrence of two consecutive action potentials. Thus, the $f^{th}$ ISI is $\Delta t^f = t^f - t^{f-1}$ for $f \geq 1$ with the convention $t^0 = 0$. As a result of this, $t^f$ can be expressed in terms of the summation of ISIs as follows:

$$t^f = \sum_{k=1}^{f} \Delta t^k. \tag{3.4}$$

The input current $I(t)$ can be an external current $I^E(t; \boldsymbol{\theta})$ to the neuron or a synaptic current $I^s(t)$ delivered to the neuron in a network or a combination of both. Here $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_r]^T$ is a vector of parameters or decision variables over the real space. Mathematically, the synaptic current $I^s(t)$ is modeled as

$$I^s(t) = -g_e(t)(v(t) - E_e) - g_i(t)(v(t) - E_i). \tag{3.5}$$

Here, $g_e(t)$ and $g_i(t)$ are the excitatory and inhibitory synaptic conductances respectively. $E_e$ and $E_i$ are the excitatory and inhibitory membrane reversal potentials respectively. Typically, the synaptic conductance $g_x(t),\ x \in \{e, i\}$ is modeled by taking the weighted sum of all presynaptic neuronal activities and is represented in the following form [56]:

$$g_x(t) = \sum_{j=1}^{N_x} \sum_{f} w_j K(t - t_j^f). \tag{3.6}$$

$N_x$ is the total number of presynaptic neurons of type $x$. $w_j$ is the weight of the synapse $j$ to the post-synaptic neuron. $t_j^f$ is the time of the $f^{th}$ incoming action

potential from the synapse $j$ to the post-synaptic neuron. $K(t - t_j^f)$ models the stereotypical time course of postsynaptic conductances following presynaptic spikes. We assume that each presynaptic neuron establishes only one synapse with each of its postsynaptic neurons and all synaptic connections within the network are excitatory i.e. $g_i = 0$ and $g_e \geq 0$ with $E_e \geq v_p$. Moreover, we assume that there is no time delay in the feedback loop. With this, we state the problem as follows:

Consider a network of $n(\geq 1)$ cortical spiking neurons denoted by $S_1, S_2, \cdots, S_n$. The neuron $S_j, 1 \leq j \leq n$ in this network is synaptically connected with the remaining $n-1$ neurons and thus receives and transfers synaptic information within the network in the form of synaptic input currents $I^s(t)$ according to equations (3.5) and (3.6). The dynamics of neuron $S_j$ are given by equation (3.2). External input current $I^E(t;\boldsymbol{\theta})$ drives the network by stimulating the neuron $S_1$. Here $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_r]^T$ is a vector of parameters or decision variables over the real space. The measured output of the neuron $S_j$ is represented as a sequence of ISIs $\{\Delta t_j^1, \Delta t_j^2, \cdots\}$. With this setup, we formulate the problem as follows: Under what conditions is the $f^{th}$ ISI $\Delta t_j^f$ of the $j^{th}$ neuron $S_j$ expressible as

$$\Delta t_j^f = h_{j,f}(\boldsymbol{\theta}) \tag{3.7}$$

for all $f \geq 1$, where $h_{j,f}$ is a continuously differentiable function of $\boldsymbol{\theta}$ in a $\delta(> 0)$ neighborhood of $\boldsymbol{\theta_0}$ i.e. $h_{j,f} \in \mathcal{C}^1$ on a domain $N_\delta(\boldsymbol{\theta_0})$ with $N_\delta(\boldsymbol{\theta_0}) := \{\boldsymbol{\theta} : |\boldsymbol{\theta} - \boldsymbol{\theta_0}| < \delta\}$?

## 3.4 Theoretical Results

We first consider a single neuron which is stimulated by an input current $I(t;\boldsymbol{\theta})$. The dynamics of the neuron are given by equation (3.2) with $I(t) = I(t;\boldsymbol{\theta})$. We establish

conditions under which the $i^{th}$ ISI of this neuron is expressible as

$$\Delta t^i = h_i(\boldsymbol{\theta}). \tag{3.8}$$

Here, $h_i \in \mathcal{C}^1$ on a domain $N_\delta(\boldsymbol{\theta_0})$ for some $\delta > 0$ and $i \geq 1$. We call this "**Problem 1**". Then we consider a single neuron stimulated by synaptic currents induced by spikes of a presynaptic neuron. We represent the $i^{th}$ spike time of the presynaptic neuron by $t_p^i$. Here, the subscript "p" stands for the presynaptic neuron. The synaptic current $I^s(t; t_p^1, t_p^2, \cdots, t_p^r)$ induced by a sequence of spikes $\{t_p^1, t_p^2, \cdots, t_p^r\}$ of a presynaptic neuron (see equations (3.5) and (3.6)) stimulates the postsynaptic neuron. Here, $r \geq 1$. We define $\Delta t_p^i$ as the $i^{th}$ ISI of the presynaptic neuron. Since $\Delta t_p^i = t_p^i - t_p^{i-1}$ by definition, we can write $I^s(t; t_p^1, t_p^2, \cdots, t_p^r) = I^s(t; \boldsymbol{\beta})$ where $\boldsymbol{\beta} = [\Delta t_p^1, \Delta t_p^2, \cdots, \Delta t_p^r]^T$ is a vector of first $r$ ISIs of the presynaptic neuron. With this, we establish conditions under which the $i^{th}$ ISI $\Delta t^i, i \geq 1$ of the postsynaptic neuron is expressible as

$$\Delta t^i = h_i^*(\boldsymbol{\beta}) \tag{3.9}$$

where $h_i^* \in \mathcal{C}^1$ on a domain $N_\delta(\boldsymbol{\beta_0})$ for some $\delta > 0$. Here $N_\delta(\boldsymbol{\beta_0}) := \{\boldsymbol{\beta} : |\boldsymbol{\beta} - \boldsymbol{\beta_0}| < \delta\}$. We call this "**Problem 2**". Finally, we use results of Problem 1 and 2 to establish conditions for the existence of equation (3.7).

It is clear from section 3.3 that $S_1$ is the only neuron in the network which is stimulated by both the external input current $I^E(t; \boldsymbol{\theta})$ and the synaptic currents available from the remaining $n-1$ neurons in the network. The other $n-1$ neurons in the network are driven by only the synaptic currents and thus have indirect effect of $I^E(t; \boldsymbol{\theta})$ through the ISIs of the neuron $S_1$. Also, $I^E(t; \boldsymbol{\theta})$ is the only stimulating

current to the neuron $S_1$ till the time at which synaptic currents from the remaining neurons in the network arrive to the neuron $S_1$. Therefore, the problem described by equation (3.7) can also be viewed as the combination of the following subproblems:

1. Under what conditions are ISIs of neuron $S_1$ continuously differentiable w.r.t. $\boldsymbol{\theta}$ when the only available stimulating input current to the neuron $S_1$ is the external input current $I^E(t;\boldsymbol{\theta})$?

2. Under what conditions are ISIs of neuron $S_j, j \neq 1$ continuously differentiable w.r.t. $\boldsymbol{\theta}$?

3. Under what conditions are ISIs of neuron $S_1$ continuously differentiable w.r.t. $\boldsymbol{\theta}$ when the stimulating input current to the neuron $S_1$ is the sum of the external input current $I^E(t;\boldsymbol{\theta})$ and the synaptic currents from the remaining $n-1$ neurons of the network?

Let us first consider an example of a recurrent network of two synaptically connected neurons and show that the establishment of conditions for the existence of equations (3.8) and (3.9) is sufficient to solve subproblems (a), (b) and (c) and thus guarantee the existence of equation (3.7) in case of two neurons.

**Example 3.4.1.** *Fig. 3.1 shows a network of two synaptically connected neurons which is driven by an external stimulating input current* $I^E(t;\boldsymbol{\theta})$.

*At this point, let us assume that equations (3.8) and (3.9) hold. For simplicity, let us further assume that* $\Delta t_1^1 < \Delta t_2^1 < \Delta t_1^2$ *i.e. the first spike of the neuron* $S_2$ *occurs in between the time of the first and the second spike of the neuron* $S_1$. *Thus the neuron* $S_1$ *receives the feedback information, in form of the synaptic input current* $I^s(t;t_2^1)$, *from the neuron* $S_2$ *at time* $t = t_2^1$.

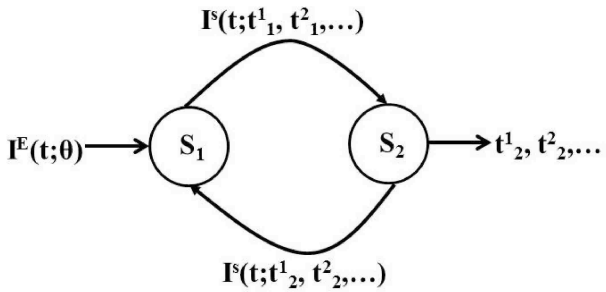



Figure 3.1: A recurrent network of two neurons: Here $I^E(t;\boldsymbol{\theta})$ is the external input current with parameter $\boldsymbol{\theta}$. $I^s(t;t^1_1,t^2_1,\cdots)$ and $I^s(t;t^1_2,t^2_2,\cdots)$ are synaptic input currents to the neuron $S_2$ and $S_1$ respectively. Here $t^1_1,t^2_1,\cdots$ and $t^1_2,t^2_2,\cdots$ are the spike trains of neurons $S_1$ and $S_2$ respectively.

*We first consider the subproblem (a) and show that this subproblem can be solved using equation (3.8) i.e. Problem 1. It is clear that $I^E(t;\boldsymbol{\theta})$ is the only stimulating input current to the neuron $S_1$ for $t < t^1_2$. Therefore by setting $I(t;\boldsymbol{\theta}) = I^E(t;\boldsymbol{\theta})$ in Problem 1 till $t = \Delta t^1_1$, the continuous differentiability of the ISI $\Delta t^1_1$ of the neuron $S_1$ w.r.t. $\boldsymbol{\theta}$ can be established using equation (3.8).*

*Next we consider the subproblem (b) and show that this subproblem can be solved using equations (3.8) and (3.9) i.e. Problem 1 and 2. Since $\Delta t^1_1$ is the only ISI of the neuron $S_1$ till the time $t = t^1_2$, equation (3.9) establishes that $\Delta t^1_2$ is a continuously differentiable function in a small neighborhood of $\Delta t^1_1$ (Problem 2). Also from equation (3.8), we know that $\Delta t^1_1$ is a continuously differentiable function of $\boldsymbol{\theta}$ (Problem 1). Now using the fact that a continuously differentiable function of a continuously differentiable function is also continuously differentiable w.r.t. the underlying arguments, equations (3.8) and (3.9) together establish that $\Delta t^1_2$ is a continuously differentiable function of $\boldsymbol{\theta}$.*

*Finally, we consider the effect of feedback on $S_1$ and show that the subproblem (c) can be solved using equation (3.8). It is clear that for $t \in [t^1_1, t^1_2)$, $I^E(t;\boldsymbol{\theta})$ is the only*

*input to the neuron $S_1$. For $t \in [t_2^1, t_1^2]$, the total input to the neuron $S_1$ is the sum of $I^E(t;\boldsymbol{\theta})$ and $I^s(t;t_2^1)$ (the synaptic current from $S_2$ to $S_1$). By definition $t_2^1 = \Delta t_2^1$. From subproblem (b), we know that $\Delta t_2^1$ is a continuous differentiable function of $\boldsymbol{\theta}$. Thus, $I^s(t;t_2^1)$ is an implicit function of $\boldsymbol{\theta}$ and the total input current to the neuron $S_1$ for $t \in [t_2^1, t_1^2]$ i.e. the sum of $I^E(t;\boldsymbol{\theta})$ and $I^s(t;t_2^1)$ is a function of $\boldsymbol{\theta}$. Since the input current to the neuron $S_1$ is continuous in $t$ at $t = t_2^1$, equation (3.8) establishes the continuous differentiability of $\Delta t_1^2$ w.r.t. $\boldsymbol{\theta}$.*

*If we continue the arguments presented above for $t \geq \Delta t_1^2$, we find that the ISIs of neurons $S_1$ and $S_2$ beyond the time $t = \Delta t_1^2$ are also continuously differentiable function of $\boldsymbol{\theta}$. Thus, the establishment of the existence of equations (3.8) and (3.9) is sufficient to guarantee the existence of equation (3.7) for a network of two neurons.*

Example 3.4.1 demonstrates that equations (3.8) and (3.9) along with the fact "a continuously differentiable function of a continuously differentiable function is also continuously differentiable w.r.t. the underlying arguments" establishes the existence of equation (3.7) for a recurrent network of two neurons. By following the arguments in this example, it is not difficult to show that the existence of equations (3.8) and (3.9) is sufficient to guarantee the existence of equation (3.7) for a recurrent network of $n > 2$ synaptically connected neurons.

We next establish the existence of equations (3.8) and (3.9) by solving Problem 1 and Problem 2 respectively. In both problem 1 and problem 2, we first make use of the well known existence and uniqueness theorems (Theorems 3.2.1 and 3.2.2) for ordinary differential equations. These existence theorems ensure the existence of a continuously differentiable solution of equation (3.2). Then we apply the implicit function theorem (Theorem 3.2.3) and establish conditions under which equations

(3.8) and (3.9) exist.

### 3.4.1 Problem 1

In this section, we establish conditions under which equation (3.8) exists. The description of the underlying problem is as follows:

An input current $I(t;\boldsymbol{\theta})$, which is a function of $t$ and $\boldsymbol{\theta}$, stimulates a single neuron. For $\boldsymbol{\theta} = \boldsymbol{\theta_0}$, the neuron fires a sequence of action potentials according to equation (3.2). The goal is to find conditions under which equation (3.8) exists in a small neighborhood (may be infinitesimally small) of $\boldsymbol{\theta_0}$.

**Notation 3.4.2.** *In this section, $t_0^i$ and $\Delta t_0^i$ represent the time of the $i^{th}$ action potential and the $i^{th}$ ISI of a single neuron respectively when $\boldsymbol{\theta} = \boldsymbol{\theta_0}$. $\Delta t^i$ represents the $i^{th}$ ISI of a single neuron otherwise. $\mathcal{C}$ stands for continuity. $\mathcal{C}^1$ stands for continuous differentiability.*

**Definition 3.4.3.** *We define a domain $D_i$ on the $(t, \boldsymbol{x}, \boldsymbol{\theta})$ space with $\boldsymbol{x} = [v, u]^T$ as $D_i := (t_0^{i-1} - \epsilon, t_0^i + \epsilon) \times (v_{min}, v_{max}) \times (u_{min}, u_{max}) \times (\boldsymbol{\theta_0} - \delta, \boldsymbol{\theta_0} + \delta)$ where $\epsilon > 0$, $\delta > 0$ and $t_0^0 = 0$. Here $v_{min} > -\infty$, $u_{min} > -\infty$, $v_{max} < \infty$, and $u_{max} < \infty$.*

**Definition 3.4.4.** *We define $N_\delta(\boldsymbol{\theta_0}) := \{\boldsymbol{\theta} : |\boldsymbol{\theta} - \boldsymbol{\theta_0}| < \delta\}$.*

**Assumption 3.4.5.** *There exists an $\epsilon > 0$ such that the solution of equation (3.2) exists on the domain $D_i$ for $\boldsymbol{\theta} = \boldsymbol{\theta_0}$.*

**Assumption 3.4.6.** *The solution of equation (3.2) is non-chaotic.*

The following result states conditions under which equation (3.8) exists.

**Lemma 3.4.7.** *Under Assumptions 3.4.5 and 3.4.6, there exists a* $\delta^*(\leq \delta) > 0$ *and a unique function* $h_i$ *such that* $\Delta t^i = h_i(\boldsymbol{\theta})$ *with* $h_i \in \mathcal{C}^1$ *and* $i \geq 1$ *on* $N_{\delta^*}(\boldsymbol{\theta_0})$ *if*

1. $\frac{\partial f(v,u)}{\partial v} \in \mathcal{C}$ *on* $D_i$ *and* $\frac{\partial f(v,u)}{\partial u} \in \mathcal{C}$ *on* $D_i$,
2. $\frac{\partial g(v,u)}{\partial v} \in \mathcal{C}$ *on* $D_i$ *and* $\frac{\partial g(v,u)}{\partial u} \in \mathcal{C}$ *on* $D_i$,
3. $I(t;\boldsymbol{\theta}) \in \mathcal{C}$ *on* $D_i$ *and* $\frac{\partial I(t;\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \in \mathcal{C}$ *on* $D_i$,
4. $\frac{\partial v(t,\boldsymbol{x}(t_0^{i-1}),\boldsymbol{\theta_0})}{\partial t} \mid_{t=t_0^i} \neq \frac{\partial v_p(t)}{\partial t} \mid_{t=t_0^i}$ *i.e. the partial derivative of the membrane potential* $v(t,\boldsymbol{x}(t_0^{i-1}),\boldsymbol{\theta_0})$ *w.r.t.* $t$ *is not equal to the partial derivative of the firing threshold* $v_p(t)$ *w.r.t.* $t$ *at the time of the* $i^{th}$ *action potential* $t = t_0^i$.

**Remark 3.4.8.** *If the firing threshold* $v_p(t)$ *is time invariant i.e.* $\frac{\partial v_p(t)}{\partial t} = 0$ *for all* $t \geq 0$ *(as is the case for most of the spiking neuron models), the required condition of Lemma 3.4.7 reduces to the nonzero slope of* $v(t,\boldsymbol{x}(t_0^{i-1}),\boldsymbol{\theta_0})$ *at the time of the* $i^{th}$ *action potential i.e.* $\frac{\partial v(t,\boldsymbol{x}(t_0^{i-1}),\boldsymbol{\theta_0})}{\partial t} \mid_{t=t_0^i} \neq 0$.

*Proof.* We first consider equation (3.2) with an initial condition $\boldsymbol{x}(0) \equiv [v(0), u(0)]^T$ at $t = 0$ where $v(0) \in D_1$ and $u(0) \in D_1$ are constants. We define $f_1(t,\boldsymbol{x},\boldsymbol{\theta}) = f(v,u) + I(t;\boldsymbol{\theta})$ where $\boldsymbol{x} \equiv [v,u]^T$. From the hypothesis of Lemma 3.4.7, $f_1(t,\boldsymbol{x},\boldsymbol{\theta}) \in \mathcal{C}$ on $D_1$ and $g(\boldsymbol{x}) \in \mathcal{C}$ on $D_1$. Also, the partial derivative of $f_1(t,\boldsymbol{x},\boldsymbol{\theta})$ w.r.t. $\boldsymbol{x}$ and $\boldsymbol{\theta}$ is continuous on $D_1$ and the partial derivative of $g(\boldsymbol{x})$ w.r.t. $\boldsymbol{x}$ is continuous on $D_1$. Thus by Theorems 3.2.1 and 3.2.2 along with Assumption 3.4.5, there exists a $\delta_1(\leq \delta) > 0$ such that for any $\boldsymbol{\theta} \in N_{\delta_1}(\boldsymbol{\theta_0})$, there exist unique solutions $v(t) = v(t,\boldsymbol{x}(0),\boldsymbol{\theta}) \in \mathcal{C}^1$ and $u(t) = u(t,\boldsymbol{x}(0),\boldsymbol{\theta}) \in \mathcal{C}^1$ of equation (3.2) for all $t \in [0, t_0^1 + \epsilon)$.

We define a function $H(t,\boldsymbol{x}(0),\boldsymbol{\theta}) = v(t,\boldsymbol{x}(0),\boldsymbol{\theta}) - v_p(t)$. Clearly, $H(t,\boldsymbol{x}(0),\boldsymbol{\theta})$ is continuously differentiable w.r.t. $t$ and $\boldsymbol{\theta}$ for $t \in [0, t_0^1 + \epsilon)$ and $\boldsymbol{\theta} \in N_{\delta_1}(\boldsymbol{\theta_0})$. Also,

$H(t_0^1, \boldsymbol{x}(0), \boldsymbol{\theta_0}) = 0$. In addition, the partial derivative of $H(t, \boldsymbol{x}(0), \boldsymbol{\theta})$ w.r.t. $t$ at $t = t_0^1$ and $\boldsymbol{\theta} = \boldsymbol{\theta_0}$ is nonzero i.e. $\frac{\partial v(t, \boldsymbol{x}(0), \boldsymbol{\theta_0})}{\partial t} \mid_{t=t_0^1} \neq \frac{\partial v_p(t)}{\partial t} \mid_{t=t_0^1}$ from the hypothesis of Lemma 3.4.7. Now by applying the implicit function theorem (Theorem 3.2.3) on the function $H(t, \boldsymbol{x}(0), \boldsymbol{\theta})$ for $t \in [0, t_0^1 + \epsilon)$ and $\boldsymbol{\theta} \in N_{\delta_1}(\boldsymbol{\theta_0})$, we find that there exists a $\delta_1^* (\leq \delta_1) > 0$ and a unique function $h_1$ on the domain $N_{\delta_1^*}(\boldsymbol{\theta_0})$ such that

- $h_1 \in \mathcal{C}^1$ on $N_{\delta_1^*}(\boldsymbol{\theta_0})$,
- $\Delta t_0^1 = h_1(\boldsymbol{\theta_0})$,
- $H(h_1(\boldsymbol{\theta}), \boldsymbol{x}(0), \boldsymbol{\theta}) = 0$ i.e. $\Delta t^1 = h_1(\boldsymbol{\theta})$ for every $\boldsymbol{\theta} \in N_{\delta_1^*}(\boldsymbol{\theta_0})$.

Next we consider equation (3.2) with $\boldsymbol{x}(h_1(\boldsymbol{\theta})) \equiv [c, u(h_1(\boldsymbol{\theta}), \boldsymbol{x}(0), \boldsymbol{\theta}) + d]^T$ as the initial condition at $t = h_1(\boldsymbol{\theta})$, where $\boldsymbol{\theta} \in N_{\delta_1^*}(\boldsymbol{\theta_0})$. It should be noted that $t = h_1(\boldsymbol{\theta})$ is the time at which the reset of $v(t, \boldsymbol{x}(0), \boldsymbol{\theta})$ and $u(t, \boldsymbol{x}(0), \boldsymbol{\theta})$ occurs according to equation (3.2).

From the hypothesis of Lemma 3.4.7, $f_1(t, \boldsymbol{x}, \boldsymbol{\theta}) \in \mathcal{C}$ on $D_2$ and $g(\boldsymbol{x}) \in \mathcal{C}$ on $D_2$. Also, the partial derivative of $f_1(t, \boldsymbol{x}, \boldsymbol{\theta})$ w.r.t. $\boldsymbol{x}$ and $\boldsymbol{\theta}$ is continuous on $D_2$ and the partial derivative of $g(\boldsymbol{x})$ w.r.t. $\boldsymbol{x}$ is continuous on $D_2$. We know from the reset conditions of equation (3.2) that at $t = h_1(\boldsymbol{\theta})$, $\boldsymbol{x}(t) \equiv [c, u(h_1(\boldsymbol{\theta}), \boldsymbol{x}(0), \boldsymbol{\theta}) + d]^T$. We also know that $u(h_1(\boldsymbol{\theta}), \boldsymbol{x}(0), \boldsymbol{\theta}) + d \in \mathcal{C}^1$ on $N_{\delta_1^*}(\boldsymbol{\theta_0})$. Thus again by Theorems 3.2.1 and 3.2.2 along with Assumptions 3.4.5 and 3.4.6, there exists a $\delta_2 (\leq \delta_1^*) > 0$ such that for any $\boldsymbol{\theta}$ which satisfies

$$|c - v(h_1(\boldsymbol{\theta}), \boldsymbol{x}(0), \boldsymbol{\theta_0})| + |u(h_1(\boldsymbol{\theta}), \boldsymbol{x}(0), \boldsymbol{\theta}) - u(h_1(\boldsymbol{\theta}), \boldsymbol{x}(0), \boldsymbol{\theta_0})| + |\boldsymbol{\theta} - \boldsymbol{\theta_0}| < \delta_2, \quad (3.10a)$$

$$h_1(\boldsymbol{\theta}) \geq t_0^1 - \epsilon, \quad (3.10b)$$

there exist unique solutions $v(t) = v(t, \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta}) \in \mathcal{C}^1$ and $u(t) = u(t, \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta}) \in \mathcal{C}^1$ of equation (3.2) for all $t \in [t_0^1 - \epsilon_1, t_0^2 + \epsilon)$.

We define a function $H(t, \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta}) = v(t, \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta}) - v_p(t)$. It is clear that $H(t, \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta})$ is continuously differentiable w.r.t. $t$ and $\boldsymbol{\theta}$ for all $t \in [t_0^1 - \epsilon_1, t_0^2 + \epsilon)$ and for all $\boldsymbol{\theta}$ satisfying equation (3.10). Also, $H(t_0^2, \boldsymbol{x}(h_1(\boldsymbol{\theta_0})), \boldsymbol{\theta_0}) = 0$. In addition, the derivative of $H(t, \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta})$ w.r.t. $t$ at $t = t_0^2$ and $\boldsymbol{\theta} = \boldsymbol{\theta_0}$ is nonzero i.e. $\frac{\partial v(t, \boldsymbol{x}(h_1(\boldsymbol{\theta_0})), \boldsymbol{\theta_0})}{\partial t} \mid_{t=t_0^2} \neq \frac{\partial v_p(t)}{\partial t} \mid_{t=t_0^2}$ from the hypothesis of Lemma 3.4.7. Now by applying the implicit function theorem (Theorem 3.2.3) on the function $H(t, \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta})$ for $t \in [t_0^1 - \epsilon_1, t_0^2 + \epsilon)$ and $\boldsymbol{\theta}$ satisfying equation (3.10), we find that there exists a $\delta_2^* (\leq \delta_2) > 0$ and a unique function $h_2$ on $N_{\delta_2^*}(\boldsymbol{\theta_0})$ such that

- $h_2 \in \mathcal{C}^1$ on $N_{\delta_2^*}(\boldsymbol{\theta_0})$,
- $\Delta t_0^2 = h_2(\boldsymbol{\theta_0})$,
- $H(h_2(\boldsymbol{\theta}), \boldsymbol{x}(h_1(\boldsymbol{\theta})), \boldsymbol{\theta}) = 0$ i.e. $\Delta t^2 = h_2(\boldsymbol{\theta})$ for every $\boldsymbol{\theta} \in N_{\delta_2^*}(\boldsymbol{\theta_0})$.

Now the claim in Lemma 3.4.7 for $i \geq 3$ follows directly from mathematical induction.

□

### 3.4.2 Problem 2

In this section, we establish conditions under which equation (3.9) exists. The description of the underlying problem is as follows:

The synaptic current $I^s(t; \boldsymbol{\beta})$ stimulates the postsynaptic neuron. Here $\boldsymbol{\beta} = [\Delta t_p^1, \Delta t_p^2, \cdots, \Delta t_p^r]^T$ is a vector of first $r$ ISIs of the presynaptic neuron. For $\boldsymbol{\beta} = \boldsymbol{\beta_0}$,

the postsynaptic neuron fires its $i^{th}$ action potentials at the time $t^i$ according to equation (3.2). The goal is to find conditions under which equation (3.9) exists in a small neighborhood (may be infinitesimally small) of $\boldsymbol{\beta_0}$.

**Notation 3.4.9.** *In this section, $t_0^i$ and $\Delta t_0^i$ represent the time of the $i^{th}$ action potential and the $i^{th}$ ISI of a single neuron respectively when $\boldsymbol{\beta} = \boldsymbol{\beta_0}$. $\Delta t^i$ represents the $i^{th}$ ISI of a single neuron otherwise.*

**Definition 3.4.10.** *We define a domain $F_i$ on the $(t, \boldsymbol{x}, \boldsymbol{\beta})$ space with $\boldsymbol{x} = [v, u]^T$ as $F_i := (t_0^{i-1} - \epsilon, t_0^i + \epsilon) \times (v_{min}, v_{max}) \times (u_{min}, u_{max}) \times (\boldsymbol{\beta_0} - \delta, \boldsymbol{\beta_0} + \delta)$ where $\epsilon > 0$, $\delta > 0$ and $t_0^0 = 0$. Here $v_{min} > -\infty$, $u_{min} > -\infty$, $v_{max} < \infty$, and $u_{max} < \infty$.*

**Definition 3.4.11.** *We define $N_\delta(\boldsymbol{\beta_0}) := \{\boldsymbol{\beta} : |\boldsymbol{\beta} - \boldsymbol{\beta_0}| < \delta\}$.*

**Assumption 3.4.12.** *There exists an $\epsilon > 0$ such that the solution of equation (3.2) exists on the domain $F_i$ for $\boldsymbol{\beta} = \boldsymbol{\beta_0}$.*

The following result states conditions under which equation (3.9) exists.

**Lemma 3.4.13.** *Under Assumptions 3.4.6 and 3.4.12, there exists a $\delta^*(\leq \delta) > 0$ and a unique function $h_i^*$ such that $\Delta t^i = h_i^*(\boldsymbol{\beta_0})$ with $h_i^* \in \mathcal{C}^1$ and $i \geq 1$ on $N_{\delta^*}(\boldsymbol{\beta_0})$ if*

1. *$\frac{\partial f(v,u)}{\partial v} \in \mathcal{C}$ on $F_i$ and $\frac{\partial f(v,u)}{\partial u} \in \mathcal{C}$ on $F_i$,*
2. *$\frac{\partial g(v,u)}{\partial v} \in \mathcal{C}$ on $F_i$ and $\frac{\partial g(v,u)}{\partial u} \in \mathcal{C}$ on $F_i$,*
3. *$I^s(t; \boldsymbol{\beta}) \in \mathcal{C}$ on $F_i$ and $\frac{\partial I^s(t;\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \in \mathcal{C}$ on $F_i$,*
4. *$\frac{\partial v(t, \boldsymbol{x}(t_0^{i-1}), \boldsymbol{\beta_0})}{\partial t} \mid_{t=t_0^i} \neq \frac{\partial v_p(t)}{\partial t} \mid_{t=t_0^i}$ i.e. the partial derivative of the membrane potential $v(t, \boldsymbol{x}(t_0^{i-1}), \boldsymbol{\beta_0})$ w.r.t. $t$ is not equal to the the partial derivative of the firing threshold $v_p(t)$ w.r.t. $t$ at the time of the $i^{th}$ action potential $t = t_0^i$.*

By following the steps provided in proving the claim of Lemma 3.4.7, it is straightforward to show the claim of Lemma 3.4.13.

**Remark 3.4.14.** *If synaptic currents induced by spikes of more than one presynaptic neuron stimulate the postsynaptic neuron, then $I^s(t;\boldsymbol{\beta}))$ in Lemma 3.4.13 can be replaced by the sum of synaptic currents from all the presynaptic neurons.*

With the establishment of results for Problem 1 and 2, next we state our main result on the existence of equation (3.7).

**Notation 3.4.15.** *From here onwards, $t^i_{j,0}$ and $t^i_j$ represent the time of the $i^{th}$ action potential of the neuron $S_j$ in the network defined in section 3.3 for $\boldsymbol{\theta} = \boldsymbol{\theta_0}$ and $\boldsymbol{\theta} \neq \boldsymbol{\theta_0}$ respectively.*

**Theorem 3.4.16.** *Let us assume that neurons in the network defined in section 3.3 fire sequence of action potentials for $\boldsymbol{\theta} = \boldsymbol{\theta_0}$. Let us also assume that the conditions of Lemma 3.4.7 are satisfied for the neuron $S_1$ and the conditions of Lemma 3.4.13 are satisfied for the neuron $S_j$ for $j \in \{2, 3, \cdots, n\}$ till time $t = t^i_{j,0} + \epsilon$ for some $\epsilon > 0$. Then there exists a $\delta(> 0)$ neighborhood $N_\delta(\boldsymbol{\theta_0})$ such that the $i^{th}$ ISI $\Delta t^i_j$ of the neuron $S_j$ is expressible as $\Delta t^i_j = h_{j,i}(\boldsymbol{\theta})$ for all $i \geq 1$ and $j \in \{1, 2, \cdots, n\}$, where $h_{j,i} \in \mathcal{C}^1$ on $N_\delta(\boldsymbol{\theta_0})$.*

*Proof.* Let us define a time $t^* = \min\{t^1_{2,0}, t^1_{3,0}, \cdots, t^1_{n,0}\}$. Without loss of generality, lets say that the total number of action potentials occurred in neuron $S_1$ till time $t = t^*$ is $k_1 \geq 1$. It is clear from section 3.3 that the only stimulating input current to the neuron $S_1$ till time $t = t^*$ is the external input current $I^E(t;\boldsymbol{\theta})$ with $\boldsymbol{\theta}$ in a small neighborhood of $\boldsymbol{\theta_0}$. From the hypothesis of Theorem 3.4.16, the conditions of Lemma 3.4.7 are satisfied for the neuron $S_1$ till time $t = t^* + \epsilon$ for some (arbitrarily

small) $\epsilon > 0$. Thus from the conclusion of Lemma 3.4.7, there exists a $\delta_1(> 0)$ and a continuously differentiable function $h_{1,i} = h_i$ on a domain $N_{\delta_1}(\boldsymbol{\theta_0})$ such that the $i^{th}$ ISI $\Delta t_1^i$ of the neuron $S_1$ is expressible as $\Delta t_1^i = h_{1,i}(\boldsymbol{\theta})$ for all $i \in \{1, 2, \cdots, k_1\}$ and $\boldsymbol{\theta} \in N_{\delta_1}(\boldsymbol{\theta_0})$.

We now consider the remaining $n-1$ neurons in the network. Till time $t = t^* + \epsilon$, the total stimulating input current to individual neurons in the remaining $n-1$ neurons is the synaptic current $I^s(t; t_1^1, t_1^2, \cdots, t_1^{k_1})$ from the neuron $S_1$ for $\boldsymbol{\theta} \in N_{\delta_1}(\boldsymbol{\theta_0})$. Defining $\boldsymbol{\beta} = [\Delta t_1^1, \Delta t_1^2, \cdots, \Delta t_1^{k_1}]^T$, we can write $I^s(t; t_1^1, t_1^2, \cdots, t_1^{k_1})$ as $I^s(t; \boldsymbol{\beta})$. Since $t^* = \min\{t_{2,0}^1, t_{3,0}^1, \cdots, t_{n,0}^1\}$ by definition, lets say that $t^* = t_{j,0}^1$ for some $j \in \{2, 3, \cdots, n\}$. From the hypothesis of Theorem 3.4.16, the conditions of Lemma 3.4.13 are satisfied for the neuron $S_j$ till time $t = t^* + \epsilon$. Thus from the conclusions of Lemma 3.4.13, there exists a $\delta_2(> 0) \leq \delta_1$ and a continuously differentiable function $h_1^*$ such that the first ISI $\Delta t_j^1$ of the neuron $S_j$ is expressible as $\Delta t_j^1 = h_1^*(\boldsymbol{\beta})$ for $\boldsymbol{\beta} \in N_{\delta_2}(\boldsymbol{\beta_0})$. We know that $\Delta t_1^i$ for $i \in \{1, 2, \cdots, k_1\}$ is continuously differentiable w.r.t. $\boldsymbol{\theta}$ for $\boldsymbol{\theta} \in N_{\delta_1}(\boldsymbol{\theta_0})$. Now using the fact that a continuously differentiable function of a continuously differentiable function is also continuously differentiable w.r.t. the underlying arguments, it is easy to see that $\Delta t_j^1$ is also continuously differentiable w.r.t. $\boldsymbol{\theta}$ for $\boldsymbol{\theta} \in N_{\delta_3}(\boldsymbol{\theta_0})$ where $\delta_3(> 0) \leq \delta_2$. Thus there exists a $\delta_3(> 0)$ and a continuously differentiable composite function $h_{j,1}$ on $N_{\delta_3}(\boldsymbol{\theta_0})$ such that the first ISI $\Delta t_j^1$ of the neuron $S_j$ is expressible as $\Delta t_j^1 = h_{j,1}(\boldsymbol{\theta})$.

It is clear that the synaptic current $I^s(t; t_j^1)$, induced by the first action potential of the neuron $S_j$, contributes in the total stimulating input current to the remaining $n-1$ neurons beyond time of this action potential. Since the hypothesis of Lemma 3.4.13 is satisfied by neuron $S_l$ with $l \neq j$ and $l \neq 1$, $I^s(t; t_j^1)$ is continuously differentiable

w.r.t. $\Delta t_j^1$. We know that $\Delta t_j^1$ is continuously differentiable w.r.t. $\boldsymbol{\theta}$ on $N_{\delta_3}(\boldsymbol{\theta_0})$. Thus, $I^s(t; t_j^1)$ is continuously differentiable w.r.t. $\boldsymbol{\theta}$ and the total input current to the neuron $S_1$ i.e. $I^E(t; \boldsymbol{\theta}) + I^s(t; \Delta t_j^1)$ is a function of $t$ and $\boldsymbol{\theta}$ on $N_{\delta_3}(\boldsymbol{\theta_0})$. Now by applying the arguments presented in previous paragraphs to subsequent action potentials in the network, it is easy to see that the conclusion of Theorem 3.4.16 hold for all $j \in \{1, 2, \cdots, n\}$ and $i \geq 1$.

□

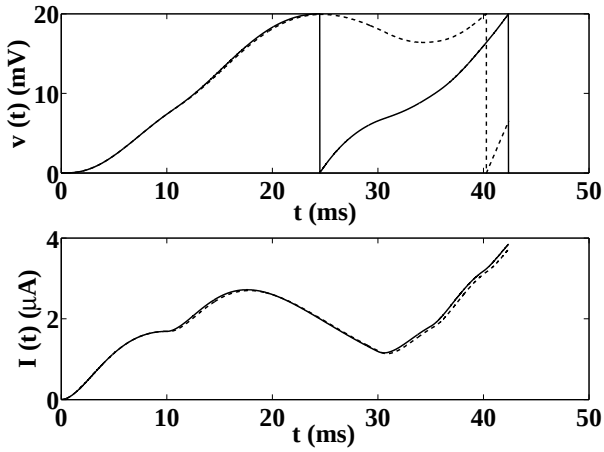
## 3.5 Simulation Results

In this section, we use simulations to demonstrate our theoretical results for a recurrent network of synaptically connected neurons. We begin with a single neuron whose dynamics is represented by the following Leaky Integrate-and-Fire model:

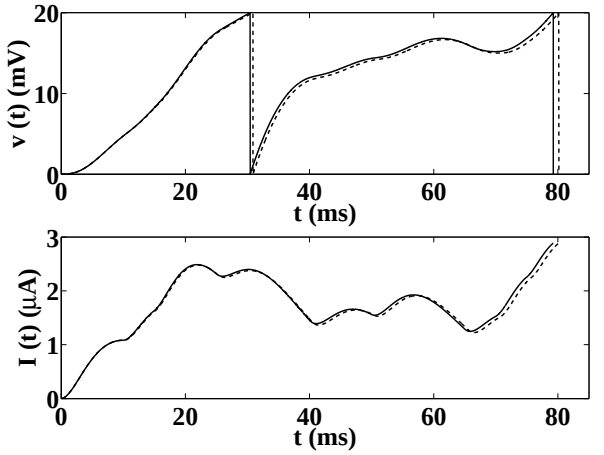
$$C\frac{\mathrm{d}v(t)}{\mathrm{d}t} = -\frac{v(t)}{R} + I(t). \tag{3.11}$$

Here, $v(t)$ is the membrane potential, $C$ is the membrane capacitance, and $R$ is the membrane resistance. $I(t)$ is a smoothly varying stimulating input current to the neuron. $v(t)$ is reset to $c$ whenever $v(t)$ exceeds a constant firing threshold $v_p$. We simulate equation (3.11) with $C = 1$ $\mu$F, $R = 10$ k$\Omega$, $c = 0$ mV, and $v_p = 20$ mV. Figure 3.2(a) shows our simulation result for the case where ISIs are discontinuous w.r.t. $I(t)$.

As shown in Figure 3.2(a), a stimulating input current $I(t)$ is designed such that the membrane potential $v(t)$ reaches the firing threshold $v_p$ with zero slope at the time of the first action potential (shown by the solid line). A small perturbation in




(a)



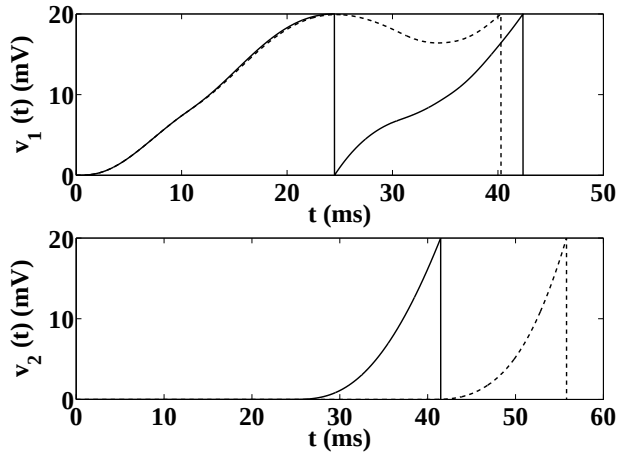

(b)

Figure 3.2: Continuity of ISIs w.r.t. a time-continuous stimulating input current $I(t)$ in single neuron models. The top plot of both (a) and (b) shows trajectories of the membrane potential $v(t)$ predicted by the Leaky Integrate-and-Fire model in response to a smoothly varying stimulating input current (solid line) and its perturbed form (dashed line). (a): The membrane potential reaches the firing threshold with zero slope at the time of the first action potential. As a result, the first ISI is discontinuous w.r.t. $I(t)$. (b): The membrane potential reaches the firing threshold with a positive slope at the time of both action potentials. As a result, the corresponding ISIs are continuous w.r.t. $I(t)$.

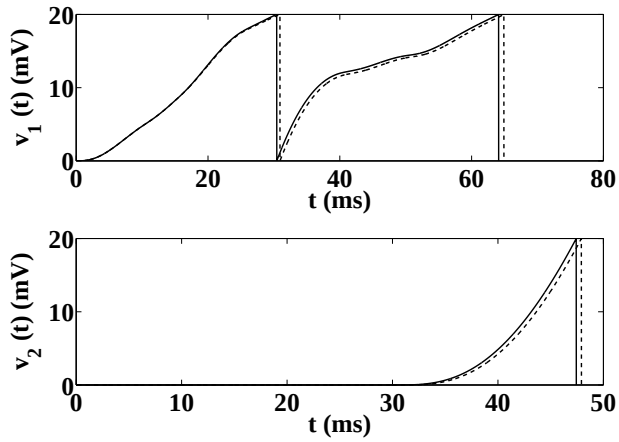
$I(t)$ shifts the occurrence of this action potential by ($\approx$) 15 ms (shown by the dashed line). This large deviation in the timing of the first action potential w.r.t. a small perturbation in $I(t)$ clearly indicates that the corresponding ISI is not continuous w.r.t. $I(t)$. Figure 3.2(b) shows the case where the membrane potential $v(t)$ reaches the firing threshold $v_p$ with a positive slope at the time of action potentials (shown by solid line). As shown in this figure, a small perturbation in $I(t)$ leads to a small change (shown by the dashed line) in the timing of the occurrence of both action potentials. Thus the corresponding ISIs are continuous w.r.t. to $I(t)$.

Next we consider a recurrent network of two synaptically connected neurons $S_1$ and $S_2$ as shown in Figure 3.1. The dynamics of both neurons are represented by equation (3.11). An external input current (same as the one used in Figure 3.2(a)) drives the network by stimulating the neuron $S_1$. Thus the total input current $I(t)$ to the neuron $S_1$ is the sum of the external input current and synaptic currents from the neuron $S_2$. The neuron $S_2$ is driven by synaptic currents induced by the action potentials of the neuron $S_1$. With this setup, we simulate the recurrent network. Figure 3.3(a) shows our simulation results for the case where ISIs of both neurons are discontinuous w.r.t. the external input current.

As shown in the top plot of Figure 3.3(a), the membrane potential $v_1(t)$ of the neuron $S_1$ reaches the firing threshold with zero slope at the time of the first action potential (shown by the solid line). A small perturbation in the designed step input current shifts the timing of the occurrence of this action potential by ($\approx$) 15 ms (shown by the dashed line) which clearly indicates that the first ISI of the neuron $S_1$ is discontinuous w.r.t. the designed input current. Next we analyze the effect of this small perturbation in the designed input current on the first ISI of the neuron $S_2$.




(a)




(b)

Figure 3.3: Continuity of ISIs w.r.t. a time-continuous stimulating input current in a recurrent network of two synaptically connected neurons. The top plot of both (a) and (b) shows trajectory of the membrane potential $v_1(t)$ of the neuron $S_1$ predicted by the Leaky Integrate-and-Fire model in response to an external input current (solid line) and its perturbed form (dashed line). The bottom plot of both (a) and (b) shows trajectories of the membrane potential $v_2(t)$ of the neuron $S_2$. (a) shows the case where ISIs of both neurons are discontinuous w.r.t. the designed input current and (b) shows the case where ISIs of both neurons are continuous w.r.t. the designed input current.

It is clear that the neuron $S_2$ is initially stimulated by the synaptic current induced by the first action potential of neuron $S_1$. We model this synaptic current as

$$I^s(t) = (q/\tau_s)(t - t_1^1)^2 \exp(-(t - t_1^1)/\tau_s)\Theta(t - t_1^1). \tag{3.12}$$

Here, $q$ is the maximum conductance transmitted by the synapse of the neuron $S_1$. $\tau_s$ is a time constant, $t_1^1$ is the time at which the first action potential occurs in neuron $S_1$, and $\Theta(\cdot)$ is the heavy-side function. For fixed $q(> 0)$ and $\tau_s(> 0)$, $I^s(t)$ is a function of $t - t_1^1$. Moreover, $I^s(t) = 0$ for $t \leq t_1^1$. Clearly, a change in $t_1^1$ will change the time at which $I^s(t)$ becomes greater than zero and thus the time at which the first action potential occurs in neuron $S_2$. This is shown in the bottom plot of Figure 3.3(a). As shown in this plot, the change in $t_1^1$ by ($\approx$) 15 ms (shown in the top plot Figure 3.3(a)) leads to same change ($\approx$ 15 ms) in the timing of the occurrence of the first action potential in the neuron $S_2$. Thus, the first ISI of neuron $S_2$ is also discontinuous w.r.t. to the designed input current even though the membrane potential $v_2(t)$ reaches the firing threshold with a positive slope.

Figure 3.3(b) shows the case where the membrane potential reaches the firing threshold with a positive slope at the time of action potentials in both neurons. As shown in this figure, a small perturbation in the designed input current (same as the one used in Figure 3.2(b)) leads to a small change in ISIs of both neurons which clearly indicates that the ISIs are continuous w.r.t. the designed input current.

## 3.6 Examples

We consider a single neuron which is stimulated by an external input current $I^E(t;\boldsymbol{\theta})$. Here, $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_r]^T$ is a vector of parameters or decision variables. The neuron fires a sequence of action potentials according to equation (3.2) for $\boldsymbol{\theta} = \boldsymbol{\theta_0}$. We assume that $I^E(t;\boldsymbol{\theta})$ is continuously differentiable w.r.t. $\boldsymbol{\theta}$ along with its continuity w.r.t. $t$ in a $\delta(>0)$ neighborhood of $\boldsymbol{\theta_0}$. We also assume that the qualitative behavior of the neuron remains same in the $\delta > 0$ neighborhood of $\boldsymbol{\theta_0}$. We represent the dynamics of the single neuron by the following known spiking single neuron models and state conditions (see Lemma 3.4.7) under which a $\delta^*(>0) \leq \delta$ neighborhood of $\boldsymbol{\theta_0}$ exists such that ISIs of the neuron are continuously differentiable function of $\boldsymbol{\theta}$ in the $\delta^*$ neighborhood of $\boldsymbol{\theta_0}$. Here we assume that parameters of these models are such that these models exhibit non-chaotic behaviors.

1. **Bidimensional models with linear $f(v,u)$:** In this class of models, we consider the Adaptive Integrate-and-Fire (aIF) model and the Adaptive Threshold Integrate-and-Fire (aTIF) model [137].

   The aIF model is given by

$$\tau \frac{\mathrm{d}v(t)}{\mathrm{d}t} = -v(t) - u(t) + RI(t), \tag{3.13a}$$

$$\tau_w \frac{\mathrm{d}u(t)}{\mathrm{d}t} = -u(t), \tag{3.13b}$$

$$\begin{aligned} &if \quad v(t) \geq 1, \quad then \\ &\quad v(t) \leftarrow 0 \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{3.13c}$$

Clearly, $f(v,u) = -v - u$ is continuously differentiable w.r.t. $v$ and $u$. Also, $g(v,u) = -u$ is continuously differentiable w.r.t. $v$ and $u$. Since $v_p(t) = 1$, the required conditions of Lemma 3.4.7 are satisfied if the slope of $v(t)$ w.r.t. $t$ at the time of action potentials is nonzero i.e. $RI^E(t_0^i; \boldsymbol{\theta_0}) - u(t_0^{i-1}) \exp(-(t_0^i - t_0^{i-1})/\tau_w) \neq 1$ for all $i \geq 1$. The same conclusion holds for the the integrate-and fire model since this model is a special case of the aIF model with $u(t) = 0$ for all $t \geq 0$. Here, $t_0^i$ is the time of the $i^{th}$ action potential for $\boldsymbol{\theta} = \boldsymbol{\theta_0}$.

The aTIF model is given by

$$\tau \frac{\mathrm{d}v(t)}{\mathrm{d}t} = -v(t) + RI(t), \tag{3.14a}$$

$$\tau_w \frac{\mathrm{d}u(t)}{\mathrm{d}t} = av(t) - u(t), \tag{3.14b}$$

$$\begin{aligned} &if \quad v(t) \geq 1 + u(t), \quad then \\ &\quad v(t) \leftarrow 0 \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{3.14c}$$

Clearly, $f(v,u) = -v$ is continuously differentiable w.r.t. $v$ and $u$. Also, $g(v,u) = av - u$ is continuously differentiable w.r.t. $v$ and $u$. Since $v_p(t) = 1 + u(t)$, the required conditions of Lemma 3.4.7 are satisfied if the partial derivative of $v(t)$ w.r.t. $t$ is not equal to the partial derivative of $u(t)$ w.r.t. $t$ at the time of action potentials.

2. **Bidimensional models with nonlinear $f(v,u)$:** In this class of models, we consider the Izhikevich model, the Adaptive Exponential (AdEx) Integrate-and-Fire (IF) model, and the Touboul model. The Izhikevich model is represented

by equation (3.2) with $f(v,u) = k(v-v_t)(v-v_r)-u$ and $g(v,u) = a(b(v-v_r)-u)$ [78]. The AdEx IF model is represented by equation (3.2) with $f(v,u) = -g_L(v-E_L)-u+g_L\Delta_T \exp((v-v_T)/\Delta_T)$ and $g(v,u) = a(v-E_L)-u$ [16]. The Touboul model is represented by equation (3.2) with $f(v,u) = v^4+2av-u$ and $g(v,u) = a(bv-u)$ [158]. In all these models, the firing threshold $v_p(t) = v_p$ (time invariant). Clearly, $f(v,u)$ is continuously differentiable w.r.t. $v$ and $u$. Also, $g(v,u)$ is continuously differentiable w.r.t. $v$ and $u$. By assumption, $I^E(t;\boldsymbol{\theta})$ is continuously differentiable w.r.t. $\boldsymbol{\theta}$ along with its continuity w.r.t. $t$ in a $\delta > 0$ neighborhood of $\boldsymbol{\theta_0}$. Because of the convexity, faster growth and regularity behavior of the nonlinear function $f(v,u)$ [158], the slope of $v(t)$ w.r.t. $t$ at $v(t) = v_p$ i.e. at the time of action potentials is always greater than 0 (required condition for the existence of the inverse of the Jacobian in order to apply the implicit function theorem). Thus, these models satisfy the required conditions of Lemma 3.4.7. From the conclusion of Lemma 3.4.7, there exists a $\delta^*(>0) \leq \delta$ neighborhood of $\boldsymbol{\theta_0}$ such that ISIs predicted by these models are continuously differentiable function of $\boldsymbol{\theta}$ in the $\delta^*$ neighborhood of $\boldsymbol{\theta_0}$. The same conclusion holds for the quadratic integrate-and-fire model since this model is a special case of the Izhikevich model with $u(t) = 0$ for all $t \geq 0$.

As an example, Figure 3.4 shows the continuity of ISIs w.r.t. $\theta$ in a small neighborhood of $\theta_0$ for the Izhikevich model. The model parameters are chosen such that the model shows regular spiking behavior with a step input current.

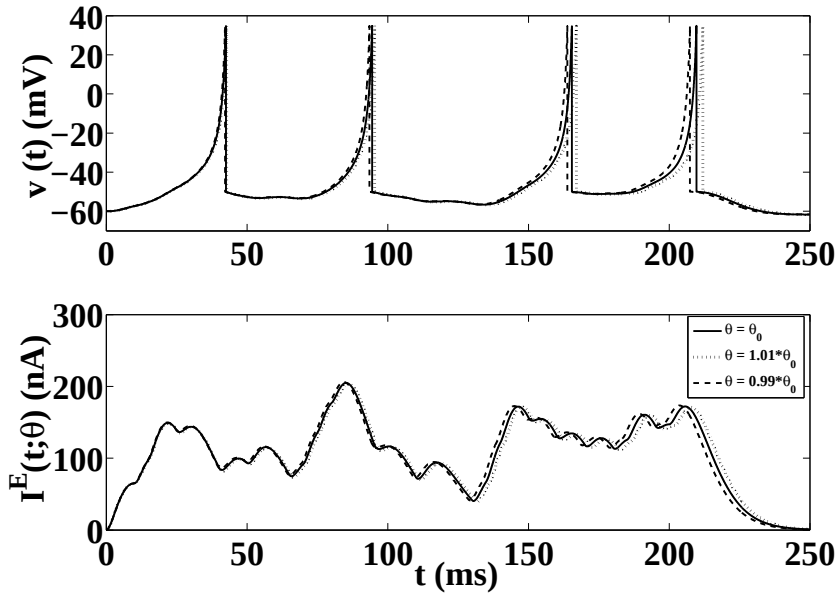



Figure 3.4: Continuity of ISIs w.r.t. $\theta$ in a small neighborhood of a $\theta_0$ in the Izhikevich model. The top plot shows the trajectories of membrane potential $v(t)$ predicted by the model in response to $I^E(t;\boldsymbol{\theta})$ (shown in the bottom plot), a time-continuous input current with its continuous differentiability w.r.t. $\boldsymbol{\theta}$.

3. **Spike Response Model:** The spike response model (SRM) is represented as

$$\begin{aligned} v_i(t) = \eta(t - \hat{t}_i) + \sum_j w_{ij} \sum_f \epsilon_{ij}(t - \hat{t}_i, t - t_j^{(f)}) \\ + \int_0^\infty \kappa(t - \hat{t}_i, s) I^E(t - s; \boldsymbol{\theta}) ds, \end{aligned} \tag{3.15}$$

where $v_i(t)$ is the membrane potential of a single neuron $i$. $\hat{t}_i$ is the firing time of the last spike of the neuron $i$ and $t_j^{(f)}$ are spikes of presynaptic neurons $j$. $w_{ij}$ is the synaptic efficacy. The function $\eta(\cdot)$ describes the form of the action potential and its spike after-potential. The function $\epsilon$ describes the time course of response to an incoming spike and $\kappa(\cdot)$ is a linear response to an input pulse. The next action potential occurs when $v_i(t)$ hits a threshold $v_{th}(t - \hat{t}_i)$ with $dv_i(t)/dt > 0$ [56].

In this model, the required conditions of Lemma 3.4.7 are satisfied if (1) $v_i(t)$ is continuously differentiable w.r.t. $t$ and $\boldsymbol{\theta}$ in a small neighborhood of $\boldsymbol{\theta_0}$, and (2) the difference between partial derivatives of $v(t)$ and $v_{th}(t - \hat{t}_i)$ w.r.t. $t$ is nonzero at the time of the next action potential. Clearly, satisfaction of these conditions require restrictions on functions $\eta$, $\epsilon$, and $\kappa$. As an example, let us assume that $v_{th}(t - \hat{t}_i)$ is time invariant and $w_{ij} = 0$. Now if $\eta$ is continuously differentiable w.r.t. $t$ for $t \geq \hat{t}_i$, $\kappa$ is continuous in $t$ for $t \geq \hat{t}_i$, and $\eta$ and $\kappa$ are continuously differentiable w.r.t. $\boldsymbol{\theta}$ at $t = \hat{t}_i$ in a small neighborhood of $\boldsymbol{\theta_0}$, then all the required conditions of Lemma 3.4.7 are satisfied by the model. As a result, there exists a small neighborhood of $\boldsymbol{\theta_0}$ such that the next ISI predicted by the model is continuously differentiable w.r.t. $\boldsymbol{\theta}$ in that neighborhood of $\boldsymbol{\theta_0}$.

## 3.7 Concluding Remarks

In this chapter, we have established conditions under which inter-spike intervals (ISIs) of individual neurons in a recurrent network of synaptically connected spiking neurons are continuously differentiable with respect to (w.r.t.) parameters (decision variables) of an external stimulating input current which drives the network. Dynamical behavior of a spiking neuron has been represented by two-state first order ordinary differential equation with reset of state variables at the occurrence of each ISI. Using existence theorems for solution of ordinary differential equations and the implicit function theorem, we have found that ISIs are continuously differentiable w.r.t. the decision variables if

1. continuously differentiable solution of spiking neurons w.r.t. time $t$ and the decision variables exists between consecutive action potentials, and

2. the partial derivative of the membrane potential of spiking neurons w.r.t. time is not equal to the partial derivative of their firing threshold w.r.t. time at the time of action potentials.

Under certain assumptions, we have shown that these conditions are fulfilled by nonlinear bidimensional spiking neuron models in the presence of a time continuous input current which is also continuously differentiable w.r.t. its parameters (see section 3.6). In case of linear bidimensional spiking neuron models, additional constraints must be imposed on the stimulating input current in order to fulfill these conditions.

Throughout our work, we have assumed that each presynaptic neuron establishes only one synapse with each of its postsynaptic neurons and all synapses within the network are identical i.e. excitatory. These assumptions are used only to simplify the

mathematical complexity of our work. Relaxation of these assumptions by including multiple synapses of both types, excitatory and inhibitory, will not change the conclusion of the reported work as long as our derived conditions are satisfied.

# Chapter 4

# Optimal Control Problems in Closed-loop Neuroprostheses: A Generalized Framework

## 4.1 Introduction

As discussed in Chapter 1, a control-theoretic system level analysis of feedback-enabled neuroprosthetic devices is highly desirable for successful transition of these devices to stable extended use in human subjects. In this direction, optimal feedback control theory [140, 94] provides an ideal theoretical framework to perform such analysis rigorously under various feedback scenarios.

In the past, the theory of optimal feedback control has been used in the context of motor control to investigate physical principles underlying the execution of skilled movements [115] and to understand optimal trajectory formation and control in multi-joint arm movements [163]. This theory has recently been applied to develop

a theoretical framework which can provide an enriched understanding of how our brain coordinates biomechanical degrees of freedom to achieve a common goal during complex reaching and grasping tasks [156, 155, 34, 147, 148, 131], a central problem in motor control. The main goal in these studies was to investigate underlying principles of motor coordination and control during complex voluntary movements using the theory of optimal feedback control to facilitate robotic applications. Therefore, these studies primarily ignored the neurophysiology of the brain involved during voluntary movements which limits the applicability of the developed framework in these studies for studying neuroprosthetic systems.

In this chapter, we propose a generalized control-theoretic framework using optimal feedback control theory for facilitating rigorous analysis of closed-loop neuroprostheses under various sensory feedback scenarios. Unlike the framework developed for investigating principles of motor control, this framework allows modulations in spiking activities of neurons by designing optimal stimulating input currents and simultaneously studying their effect on the entire system in a controlled environment. Using this framework, we formulate a minimum time optimal control problem for accomplishing voluntary single-joint movement tasks. Using the results derived in previous chapters, we perform a rigorous analysis of the closed-loop neuroprosthetic system and elucidate the importance of sensory feedback during the movement of a single-joint prosthetic arm driven by the activity of a single cortical motor neuron.

## 4.2 Closed-Loop Neuroprosthetic Control: A Generalized Framework

Figure 4.1 shows our proposed framework for rigorous control-theoretic analysis of a closed-loop neuroprosthetic system using a model-based optimal receding horizon controller (RHC) [95]. Here, "Intent" represents the direction and the goal of the

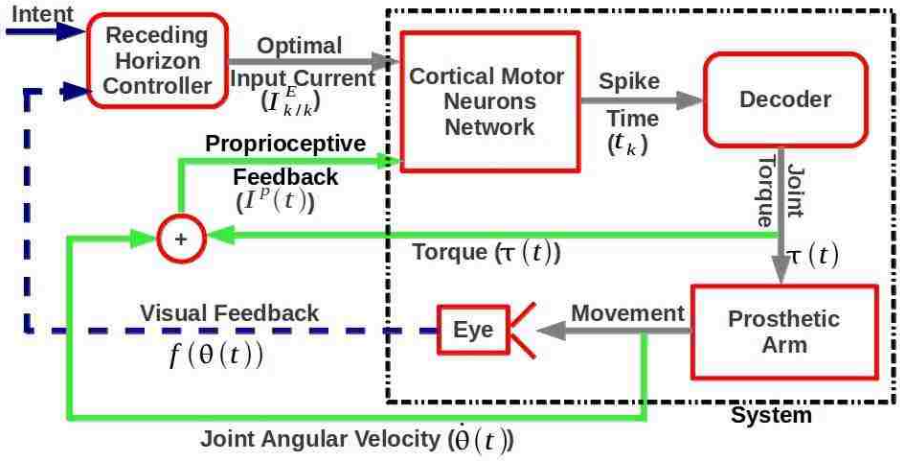



Figure 4.1: An optimal receding horizon control based closed-loop neuroprosthetic system framework.

movement. We assume that this intent information is available to the controller prior to design of the closed-loop system. "Receding Horizon Controller" represents the external controller. This controller uses a predictive model of the system and designs "optimal input current" $I^E_{k|k}$. $I^E_{k|k}$ together with $I^p(t)$, the proprioceptive feedback current, stimulates the "Cortical Motor Neurons Network". The measured output of the "Cortical Motor Neurons Network" is the time of action potentials of neurons in the network occurred between the discrete time $k$ and $k+1$ and is represented here by "Spike Time" $t_k$. Motor relevant information contained in $t_k$ is then extracted using

the "Decoder" which provides a continuous time "Joint Torque" $\tau(t)$ information. This joint torque is then used to drive the "Prosthetic Arm". The loop is closed by including "Visual Feedback" and "Proprioceptive Feedback" available from the system. Feedback information available from the "Torque" and the "Movement" of the "Prosthetic Arm" constitutes the proprioceptive feedback and represented here in the form of $I^p(t)$. The feedback information, $f(\theta(t))$ measured through the "Eye" represents the "Visual Feedback". $\theta(t)$ and $\dot{\theta}(t)$ are continuous time angular position and velocity of the joints respectively.

## 4.3 An Optimal Control Problem

Our brain plans the sequence of movements to accomplish relevant task goals. Minimum cortical effort [34] or minimum time [115] for reaching a given target using an arm represent such goals. These goals are motivated by associated rewards. A natural human tendency is to look for rewards that can easily be acquired with minimum efforts. This tendency varies with constraints on rewards. A familiar constraint is the association of the reward with the accomplishment time of the task where minimum time implies more reward. In this situation, the primary goal within the task becomes the minimum time. This goal can formally be defined as a cost function in a model-based optimal receding horizon control framework where the objective is to minimize the task relevant accomplishment time. In this section, we formulate a minimum time control problem in the framework shown in Figure 4.1 to accomplish a single joint movement task in a minimum possible time.

### 4.3.1 System Dynamics

As shown in Figure 4.1, the closed-loop neuroprosthetic system "System" is consisted of five different blocks, namely cortical motor neurons network, decoder, prosthetic arm, proprioceptive feedback, and visual feedback. For a single joint reaching task, we use the firing activity of a single cortical motor neuron to drive the single joint prosthetic arm. Therefore, we replace the "Cortical Motor Neurons Network" block in Figure 4.1 by a "Single Cortical Motor Neuron". With this, we next describe the dynamical model of individual blocks which are used in formulating the minimum time optimal control problem to accomplish a single joint movement task.

1. **Single Cortical Motor Neuron Model:** We represent the dynamical behavior of a single cortical motor neuron by the Izhikevich single neuron model [76, 77]:

$$\frac{\mathrm{d}v(t)}{\mathrm{d}t} = 0.04v^2(t) + 5v(t) + 140 - u(t) + I(t), \tag{4.1a}$$

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} = a(bv(t) - u(t)), \tag{4.1b}$$

$$\begin{aligned} &if \quad v(t) \geq 30, \quad then \\ &\quad v(t) \leftarrow c \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{4.1c}$$

Here, $v(t)$ is the membrane potential (in millivolt (mV)) at real continuous time $t$ (in millisecond (ms)). $u(t)$ is the membrane recovery variable. $I(t)$ is the total external input current to the neuron. $a, b, c, d$ are the model parameters which can be estimated from experimental single neuron data using the approach discussed in Chapter 2 of this thesis. At the initial time $t = 0$, $v(0) = c$ and $u(0) = d$. Whenever the membrane potential $v(t)$ exceeds the cut-off potential

of 30 mV, the occurrence of an action potential is assumed. At this time, the membrane potential $v(t)$ is reset to $c$ and the value of the recovery variable $u(t)$ is increased by $d$. This reset of the membrane potential and the recovery variable makes the dynamical model discontinuous in nature. Using these reset values as new initial conditions, the time of next occurrence of an action potential is determined.

We assume that the first action potential has already been occurred at time $t_1 = 0$. With this, we define $t_k$ is the time of the $k^{th}$ action potential. An inter-spike interval (ISI) is defined as the time interval between two consecutive action potentials. Mathematically, the $k^{th}$ ISI is represented as

$$\Delta t_k = \min\{t - t_k : v(t) \geq 30 | v(t_k) = c, u(t_k) = u(t_k) + d\}. \tag{4.2}$$

2. **Decoder:** As discussed in Chapter 1 of this thesis, typical decoders developed for neuroprosthetic applications such as the Weiner filter or the Kalman filter are primarily based on the average firing rate of cortical neurons which lose motor intended information because of their averaging characteristics. Also, these models are static in nature. ISI based dynamical decoder models can enhance the neuroprosthetic performances by decoding more detailed motor intended information. With this motivation, here we develop a dynamical decoder model that relates the torque generated at the joint of the prosthetic arm to the timings of action potentials of the cortical motor neuron. We express the torque induced by the $k^{th}$ action potential of the cortical motor neuron at the joint of the prosthetic arm as

$$\tau(t) = \begin{cases} \tau(t_k) + (t - t_k)\alpha & t_k \leq t \leq t(\tau^m) \\ -(t - t_k - T_c)\beta & t(\tau^m) \leq t \leq t_{k+1}, \end{cases} \tag{4.3a}$$

$$t(\tau^m) = \frac{t_k\alpha + (T_c + t_k)\beta - \tau(t_k)}{\alpha + \beta}. \tag{4.3b}$$

with $\tau(t_1) = \tau(0) = 0$. $\tau(t_k)$ is the measurement of the torque induced by the $(k-1)^{th}$ action potential at time $t = t_k$. $t(\tau^m)$ is the time at which the torque induced by the $k^{th}$ action potential reaches its maximum. $\alpha$ and $\beta$ are constant model parameters. $T_c$ assumes a constant value and defines the maximum time in which the torque induced by the $k^{th}$ action potential diminishes to zero.

3. **Single Joint Prosthetic Arm Model:** We consider the following one dimensional mechanical model to represent the dynamics of a single joint prosthetic arm which captures flexion/extension movement of the arm:

$$I_n\ddot{\theta}(t) = -k_n\dot{\theta}(t) + \tau(t). \tag{4.4}$$

Here, $\theta(t)$ is the angular position of the joint at time $t$. $\dot{\theta}(t)$ is the derivative of $\theta(t)$ with respect to time $t$ and represents the velocity of the movement. $I_n$ is the moment of inertia. $k_n\dot{\theta}(t)$ is the friction term which captures the frictional loss during the movement.

4. **Proprioceptive Feedback Model:** Proprioceptive feedback carries information such as the sense of movement, position, force and effort through natural sensory pathways which together provides the perception of the limb position.

Since these natural pathways are lost in amputees, one possibility to include proprioception in neuroprosthetics is to send information such as the joint torque and the velocity measured from the prosthetic arm to the brain in the form of external stimulating input currents. Therefore, we represent the proprioceptive feedback in the form of an external input current as

$$I_p(t) = K_\tau \tau(t) + K_\theta \dot{\theta}(t). \tag{4.5}$$

Here, $K_\tau$ and $K_\theta$ are constant gain parameters for the joint torque and the velocity respectively.

5. **Visual Feedback Model:** We include the visual feedback information as the angular difference between the final and the present position of the joint during the movement. We assume that this information is measurable at each ISI.

### 4.3.2 Problem Formulation

We formulate the overall optimal closed-loop neuroprosthetic control problem in the receding horizon control framework for all $k \geq 1$ as follows:

$$\min_{I^E_{k+j|k}, j=0,1,\cdots,N_c(k)-1} \mathbb{J}_p(k) \tag{4.6a}$$

s.t.

$$\Delta t^m \leq \Delta t_{k+j|k} \leq T_c \;\; for \;\; 0 \leq j \leq N_c(k) - 1, \tag{4.6b}$$

$$(\theta_{f|k} - \theta_{k+N_p(k)-1|k})^2 \leq \epsilon. \tag{4.6c}$$

Here, the cost function $\mathbb{J}_p(k)$ is defined as $\mathbb{J}_p(k) = T_c + \sum_{j=0}^{N_p(k)-1} \Delta t_{k+j|k}$. This cost function represents the total time required to drive the single joint prosthetic arm to the desired angular position. $k \geq 1$ represents the event of the occurrence of the $k^{th}$ action potential. $I^E_{k+j|k}, j = 0, 1, \cdots, N_c(k) - 1$ is a sequence of external stimulating input currents over the time varying control horizon $N_c(k)$. $N_p(k)$ is the time varying prediction horizon. Throughout this work, it has been assumed that $I^E_{k+j|k}$ is constant during the time interval of $\Delta t_{k+j|k}$. $\Delta t_{k+j|k}, j = 0, 1, \cdots, N_p(k) - 1$ is the sequence of predicted inter-spike intervals (ISIs) over the prediction horizon $N_p(k)$ at time $k$. These ISIs are computed using equation (4.1) with $I(t) = I^E_{k+j|k} + I^p_{k+j|k}(t)$ for $j = 0, 1, \cdots, N_c(k) - 1$ and $I(t) = I^E_{k+N_c(k)|k} + I^p_{k+j|k}(t)$ for $j = N_c(k), \cdots N_p(k) - 1$.

Equation (4.6b) defines the minimum and maximum limit on an ISI and ensures that the movement of the single joint prosthetic arm is continuous. The minimum limit $\Delta t_m$ captures the refractory period of a cortical motor neuron. This means that the cortical motor neuron cannot fire the next action potential within this time period. The maximum limit $T_c$ captures the minimum firing rate of a cortical motor neuron. This limit ensures the occurrence of the next action potential before the movement of the single joint prosthetic arm ceases. Based on our observations on experimentally recorded cortical motor neurons firing rate from a primate study, we fix the value of $T_c$ to 50 ms in this study.

Equation (4.6c) defines a target space around the final angular position to be reached during the voluntary movement of the single joint prosthetic arm. $\epsilon \geq 0$ is a user defined parameter which ensures that the control problem is feasible. $\theta_{k+N_p(k)-1|k}$ is the prediction of the angular position of the single joint prosthetic arm in the next

$N_p(k)$ ISIs predicted by the controller at time $k$. $\theta_{f|k} = \theta_f - \theta_{k|k}$ is the desired angular position of the single joint prosthetic arm at time $k$. $\theta_f$ is the desired extension / flexion of the single joint prosthetic arm assuming that the joint is initially at $0^\circ$. At time $k$, $\theta_{f|k}$ is used by the controller as visual feedback in designing the next time optimal input currents.

Out of $N_c(k)$ optimally computed input currents, only the first input current $I^E_{k|k}$ is implemented to the single cortical motor neuron for generating the $(k+1)^{th}$ action potential or the spike time. At time $k+1$, the control problem is again solved with the new measurements obtained from the system. Thus at each $k$, the overall control problem is to compute optimal $I^E_{k+j|k}, j = 0, 1, \cdots, N_c(k) - 1$ such that the cost function $\mathbb{J}_p(k)$ attains a minimum value while satisfying equations (4.6b) and (4.6c) on the system.

### 4.3.3 Feasibility

**Claim 4.3.1.** *The minimum time control problem defined by equation (4.6) is feasible for all $k > 1$ if it is feasible at $k = 1$.*

*Proof.* By the hypothesis of Claim 4.3.1, the minimum time control problem defined by equation (4.6) is feasible at $k = 1$. Thus, we can find a sequence of $I^E_{i|1}$ for $i = 1, 2, \cdots, N_c(1)$ and $I^E_{N_c(1)+j|1} = I^E_{N_c(1)|1}$ for $j = 1, 2, \cdots, N_p(1) - N_c(1)$ such that the control problem is feasible. Now at $k = 2$, we can choose a set of admissible control inputs [95] $I^E_{i|2} = I^E_{i|1}$ for $i = 2, 3, \cdots, N_c(1)$ and $I^E_{N_c(1)+j|1} = I^E_{N_c(1)|1}$ for $j = 1, 2, \cdots, N_p(1) - N_c(1)$, assuming $N_c(1)$ is fixed. This set can steer the $\theta_{N_p(1)|2}$ to the $\epsilon$ region of $\theta_{f|2}$. If we continue this way, we find that the control problem is feasible for all $k > 1$. □

## 4.4 Nonlinear Optimization Problem

For notational simplicity, we write the control problem defined by equation (4.6) as

$$\min_{\mathbf{x_k}} \mathbb{J}_p(k) \tag{4.7a}$$

s.t.

$$\mathbf{h}(\mathbf{x_k}) \geq 0. \tag{4.7b}$$

Here, $\mathbf{x_k} = [I^E_{k|k}, I^E_{k+1|k}, \cdots, I^E_{k+N_c(k)-1|k}]^T$ is a vector of decision variables. $\mathbf{h}(\mathbf{x_k}) = [\Delta t_{k|k} - \Delta t^m, \cdots, \Delta t_{k+N_c(k)-1|k} - \Delta t^m, T_c - \Delta t_{k|k}, \cdots, T_c - \Delta t_{k+N_c(k)-1|k}, \epsilon - (\theta_{f|k} - \theta_{k+N_p(k)-1|k})^2]^T$ is a vector of system constraints given by equations (4.6b) and (4.6c). $(\cdot)^T$ denotes the transpose of a vector.

We write the Lagrangian [15] of the optimization problem define by equation (4.7) as

$$L(\mathbf{x_k}, \mathbf{w}, \lambda, \mu) = \mathbb{J}_p(k) - \mu \sum_{i=1}^{2N_c(k)+1} \log(w_i) - \lambda^T(\mathbf{h}(\mathbf{x_k}) - \mathbf{w}). \tag{4.8}$$

Here, $\mu$ is a barrier parameter. $\mathbf{w} = [w_1, w_2 \cdots, w_{2N_c-1}]^T$ is a vector of slack variables and satisfies $\mathbf{h}(\mathbf{x_k}) - \mathbf{w} = 0$ for $\mathbf{w} \geq \mathbf{0}$. $\lambda$ is the non-negative vector of Lagrange multipliers with $2N_c(k)+1$ elements. $(\mathbb{J}_p(k) - \mu \sum_{i=1}^{2N_c(k)+1} \log(w_i))$ represents a logarithmic barrier function.

### 4.4.1 First-Order Optimality Conditions

The first-order optimality conditions for a minimum of equation (4.7) can be given by well known Karush-Kuhn-Tucker (KKT) conditions [15] and are here derived as

$$\nabla_{\mathbf{x_k}} L = \nabla \mathbb{J}_p(k) - \nabla \mathbf{h}(\mathbf{x_k})^T \lambda = 0, \tag{4.9a}$$

$$\nabla_{\mathbf{w}} L = -\mu \mathbf{W}^{-1} \mathbf{e} - \lambda = 0, \tag{4.9b}$$

$$\nabla_{\lambda} L = \mathbf{h}(\mathbf{x_k}) - \mathbf{w} = 0. \tag{4.9c}$$

Here, $\mathbf{W}$ is a diagonal matrix with elements of $\mathbf{w}$. $\mathbf{e}$ is a unit vector of length $2N_c(k) + 1$. $\nabla$ represents the gradient. The optimization problem now is to find optimum values of $\mathbf{x_k}$, $\mathbf{w}$ and $\lambda$ which satisfy the KKT optimality conditions i.e. equation (4.9).

### 4.4.2 Optimization Algorithm

In order to find a local solution numerically that satisfies the KKT conditions given by equation (4.9), we implement a primal-dual interior point algorithm [164]. This algorithm uses an iterative approach to search the optimality direction of decision variables. Starting from an initial guess of decision variables, a new guess of decision variables for the problem shown in equation (4.9) is determined by solving the

following Newton system:

$$\begin{bmatrix} -\mathbf{H}(\mathbf{x_k},\lambda) & \nabla\mathbf{h}(\mathbf{x_k})^T \\ \nabla\mathbf{h}(\mathbf{x_k}) & \mathbf{W}\Lambda^{-1} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x_k} \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} \nabla\mathbb{J}_p(k) - \nabla\mathbf{h}(\mathbf{xk})^T\lambda \\ \mathbf{c}(\mathbf{x_k}) + \mathbf{W}\Lambda^{-1}(\mu\mathbf{W}^{-1}\mathbf{e} - \lambda) \end{bmatrix}, \tag{4.10a}$$

$$\Delta\mathbf{w} = \mathbf{W}\Lambda^{-1}(\mu\mathbf{W}^{-1}\mathbf{e} - \lambda - \Delta\lambda). \tag{4.10b}$$

Here, $\mathbf{H}(\mathbf{x_k},\lambda)$ represents the Hessian and is defined as $\mathbf{H}(\mathbf{x_k},\lambda) = \nabla^2\mathbb{J}_p(k) - \sum_{i=1}^{2N_c(k)+1}\lambda_i\nabla^2\mathbf{h}_i(\mathbf{x_k})$. $\mathbf{c}(\mathbf{x_k}) = \mathbf{w} - \mathbf{h}(\mathbf{x_k})$. $\Lambda$ is the diagonal matrix with elements of $\lambda$. $\Delta\mathbf{x_k}$, $\Delta\lambda$ and $\Delta\mathbf{w}$ are the Newton steps in $\mathbf{x_k}$, $\lambda$ and $\mathbf{w}$ respectively. After computing step directions $\Delta\mathbf{x_k}$, $\Delta\lambda$ and $\Delta\mathbf{w}$, the implemented optimization algorithm proceeds to a new point

$$\mathbf{x_k}^N = \mathbf{x_k} + \gamma\Delta\mathbf{x_k}, \tag{4.11a}$$

$$\mathbf{w}^N = \mathbf{w} + \gamma\Delta\mathbf{w}, \tag{4.11b}$$

$$\lambda^N = \lambda + \gamma\Delta\lambda. \tag{4.11c}$$

$\gamma$ is a step control variable which is chosen such that the following conditions are satisfied:

1. $\mathbf{w}^N > \mathbf{0}$,
2. $\lambda^N > \mathbf{0}$,
3. $b_\mu(\mathbf{x_k}^N, \mathbf{w}^N) < b_\mu(\mathbf{x_k}, \mathbf{w})$,
4. $||\rho(\mathbf{x_k}^N, \mathbf{w}^N)||^2 < ||\rho(\mathbf{x_k}, \mathbf{w})||^2$.

Here, $b_\mu(\mathbf{x_k},\mathbf{w}) = \mathbb{J}_p(k) - \mu\sum_{i=1}^{2N_c(k)+1}\log(w_i)$ is a barrier function. $\mu = \nu\min((1-r)\frac{1-\zeta}{\zeta}, 2)^3\frac{\mathbf{w}^T\lambda}{m}$ is the barrier parameter. $r$ is the steplength factor which is set to

0.95. $\nu \in [0, 1)$ is a settable scale factor which is set to 0.1. $||\rho(\mathbf{x_k}, \mathbf{w})||$ measures infeasibility of the optimization problem and is defined as $\rho(\mathbf{x_k}, \mathbf{w}) = \mathbf{w} - \mathbf{h}(\mathbf{x_k})$. The last two conditions form a search strategy called the Markov filter [12]. This strategy ensures that the new point is moving towards optimality in a feasible region. Using this new point as a starting guess of decision variables, the above procedure is repeated till the following optimization stopping conditions are satisfied [164, 12]:

1. $\max(-\log_{10} \frac{|\lambda^T(\mathbf{h}(\mathbf{x_k})-\mathbf{w})|}{(|b_\mu(\mathbf{x_k},\mathbf{w}))+1|)}, 0) \geq 8$,
2. $||\rho(\mathbf{x_k}, \mathbf{w})|| \leq 10^{-6}$,
3. $||\nabla \mathbb{J}_p(k) - \nabla \mathbf{h}(\mathbf{x_k})^T \lambda|| \leq 10^{-6}$.

## 4.5 Results

In previous sections, we developed a minimum time optimal control problem (equation (4.6)) in a model-based receding horizon control framework for a cortically driven closed-loop single joint prosthetic arm. In this section, we solve the control problem numerically.

We set the Izhikevich single neuron model (equation (4.1)) parameters $a, b, c, d$ to $0.0404, 0.2497, -64.4679, 21.2777$ respectively which were obtained in Chapter 2 using an experimental data set from a primate study. $\Delta t^m$, the minimum time duration of two consecutive action potentials, was set to 2 ms. Model parameters $\alpha$, $\beta$, and $T_c$ in equation (4.3) were set to 0.5, 0.01, and 50 respectively. The moment of inertia $I_n$ and the friction parameter $k_n$ in equation (4.4) were set to 5 and 30 respectively. These choices of parameters allowed us to extend the joint of the prosthetic arm by $20^\circ$ in the time interval of $1-2$ s. Using these parameters, we studied the importance

of proprioceptive and visual feedbacks in extending the joint of the prosthetic arm by 20° in a minimum possible time.

The flexibility of a modeling framework in activating or inactivating individual feedback pathway provides an ideal platform to identify importance of these pathways in designing a closed-loop neuroprosthetic system. Therefore, here we studied four systems namely closed-loop, partial closed-loop-V, partial closed-loop-P and open-loop based on the nature of sensory feedbacks incorporated in their design. Table 4.1 shows the distinction among these systems.

| **System** | **Visual feedback** | **Proprioceptive feedback** |
|---|---|---|
| Closed-loop | Yes | Yes |
| Partial closed-loop-V | Yes | No |
| Partial closed-loop-P | No | Yes |
| Open-loop | No | No |

Table 4.1: System Design

As shown in Table 4.1, the activation and inactivation of visual and proprioceptive feedbacks in these systems allowed us to explore the fundamental importance of these feedbacks in driving a joint of the prosthetic arm. With this, we solved the minimum time control problem defined by equation (4.6) at each $k \geq 1$. We implemented the optimization algorithm described in the previous section in MATLAB. Initial values of $\mathbf{w}$ were set to $\mathbf{h}(\mathbf{I})$. Initial values of the Lagrange multipliers were set to $\lambda_i = 1/\mathbf{h}_i$ for $i = 1, \cdots, 2N_c$ and $\lambda_{2Nc+1} = 1$. We observed that this particular choice works for almost every problem we solved in our context. We analyzed the effects of visual and proprioceptive feedback in designing the closed-loop neuroprosthetic system shown in Figure 4.1.

For systems with visual feedback, initial value of $\epsilon$ was set to 0.2. $\epsilon$ was monotonically decreased during the receding horizon problem and finally fixed to 0.01. For systems without visual feedback, $\epsilon$ was set to 0.01. Thus the final target location of $\theta$ was $19.9^\circ - 20.1^\circ$. The control horizon $N_c(0)$ was set to 4. Initial guesses for $\{I_{1|1}, \{I_{2|1}, \cdots, I_{N_c(1)|1}\}$ were set to 25. The initial prediction horizon $N_p(1)$ was computed based on the feasibility of optimization problem at the initial guesses. It should be noted here that the prediction horizon $N_p(k)$ was not decreased monotonically during the receding horizon computation. Equations (4.1), (4.3) and (4.4) were computed numerically using the Euler scheme with a fixed step size of $10^{-3}$.

### 4.5.1 Importance of Visual Feedback

It has been shown in motor intended neuroprosthetics studies that visual as well as proprioceptive feedback play important role in guiding voluntary movement tasks such as reaching or grasping [152, 14]. In [14], it has been shown that the difficulty in using a prosthetic limb for grasping a virtual object increases as one or both of these feedback signals are switched off. Particularly, it may even be impossible to attain the goal in the absence of visual feedback. In this section, we show this point using the receding horizon based optimal control framework and elucidate the importance of visual feedback in guiding the single joint prosthetic arm in attaining the desired goal.

It is well known that single neurons are intrinsically noisy. To incorporate this noisy characteristic of single neurons in the Izhikevich single neuron model given by equation (4.1), we modified equation (4.1) as

$$\frac{\mathrm{d}v(t)}{\mathrm{d}t} = 0.04v^2(t) + 5v(t) + 140 - u(t) + I(t) + I_{noise}(0, \sigma^2), \tag{4.12a}$$

$$\frac{\mathrm{d}u(t)}{\mathrm{d}t} = a(bv(t) - u(t)), \tag{4.12b}$$

$$\begin{aligned} &if \quad v(t) \geq 30, \quad then \\ &\quad v(t) \leftarrow c \quad and \quad u(t) \leftarrow u(t) + d. \end{aligned} \tag{4.12c}$$

Here, $I_{noise}(0, \sigma^2)$ is a Gaussian noise with mean 0 and variance $\sigma^2$. We say that the system shown in Figure 4.1 is nominal (WN) if $\sigma = 0$ and stochastic (N) if $\sigma > 0$.

We assumed that it is possible to reach the desired extension of the joint in the open loop, partial closed-loop-V and partial closed-loop-P system under nominal system conditions i.e. with $\sigma = 0$ in equation (4.12). With this assumption, we solved the control problem (equation (4.6)) for the open-loop system and the partial closed-loop-P system and computed optimal external input currents for both systems under the nominal condition. We set proprioceptive feedback gains $K_\tau$ and $K_\theta$ to 0 for the open-loop and partial closed-loop-V system. For the partial closed-loop-P system, we set $K_\tau = 1$ and $K_\theta = 100$. It should be noted that a model of the nominal system dynamics have been used in the receding horizon controller for predicting and computing optimal external input currents. Therefore, there is no model mismatch between the system and the controller in the nominal case. This provided us the number of ISIs required to reach the desired angular position of the joint in both systems under nominal conditions. We implemented the computed optimal external input currents using a model of the nominal system on the stochastic system with $\sigma = 20$ for the open-loop and partial closed-loop-V system and obtained same number

of ISIs as in the nominal case. This provided us angular position trajectories of the prosthetic joint for both stochastic systems.

The top plot in Figure 4.2 represents these trajectories with respect to real-time. For comparing angular position trajectory of the open-loop stochastic system with the nominal system, we have included the angular position trajectory of the open-loop nominal system in this plot of which is shown by "Open-loop(WN)".

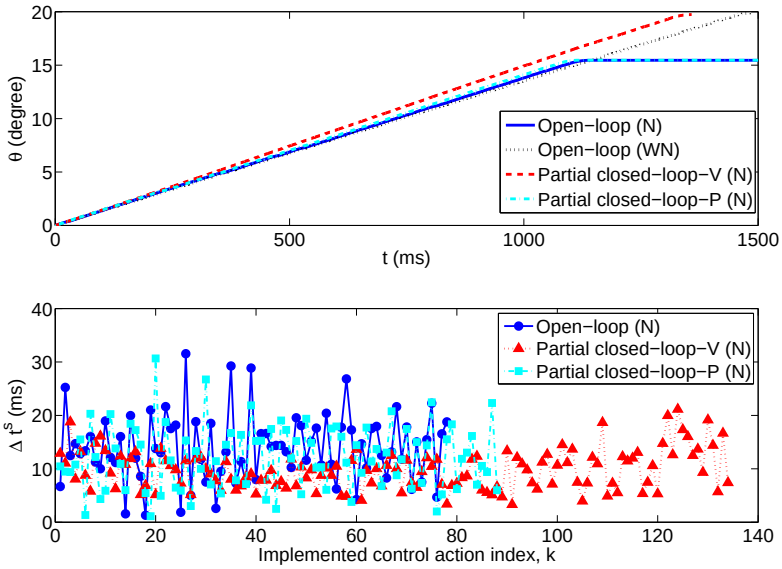



Figure 4.2: Importance of visual feedback in designing neural prosthetic system. Here, the top plot shows the angular position trajectory during the extension movement of the prosthetic joint. The y-axis represents the angular position ($\theta$ measured in degrees) and the x-axis represents the time ($t$ in milliseconds). "N" indicates the presence of noise during the design, "WN" indicates the absence of noise during the design. The bottom plot represents modulations in inter-spike intervals (ISIs) of the cortical motor neuron in the absence and the presence of visual feedback information.

In Figure 4.2, the angular trajectory of the open-loop stochastic system, "Open-loop (N)", and partial closed-loop-P stochastic system, "partial closed-loop-P (N)", clearly show that the desired angular position is not reached in both cases. Therefore

to reach the desired angular position in the presence of noise, it is necessary to modify the number of ISIs. Now we show that this modification is only possible using the available information of the prosthetic joint angular position via the visual feedback.

In reaching relevant tasks, the visual feedback carries information such as the location of the arm and the target position in the space. Naturally, this feedback is processed through eye to the visual cortex of the brain and helps the brain in modifying the planning of the movement to attain the desired goal of reaching. In a similar way, here we used the difference of the desired and measured angular position of the prosthetic joint at each time $k$ to guide the receding horizon controller for modifying the planning of the movement. Using this concept of visual feedback inclusion in the modeling framework, next we show that the prosthetic joint can be driven to the desired angular position which was impossible in the absence of the visual feedback.

We solved the control problem (equation (4.6)) using the nominal system models (equations (4.1), (4.3), and (4.4)) at time $k$. Out of $N_c(k) = 4$ computed optimal external input currents over the prediction horizon $N_p(k)$ at time $k$, we implemented the first computed external current $I^E_{k|k}$ on the partial closed-loop-V stochastic system (equations (4.12), (4.3), and (4.4)). We measured the angular position $\theta(t)$ of the prosthetic joint at time $k + 1$ and used this information as a visual feedback in designing the next input currents at time $k + 1$. The angular trajectory for this system is shown in the top plot of Figure 4.2 as "Partial closed-loop-V (N)". Clearly the trajectory for this system shows the accomplishment of the reaching task by accounting the visual feedback for corrections.

The bottom plot in Figure 4.2 shows the modulation in cortical activity of the

neuron in the presence of noise for the open-loop, the partial closed-loop-V and the partial closed-loop-P system. It indicates that the visual feedback helps the brain in rejecting the effect of external perturbations during the extension movement of the prosthetic arm by modifying the cortical activity of the neuron in the partial closed-loop-V (N) system. Next, we studied the role of proprioception in a closed-loop neural prosthetic system.

### 4.5.2 Importance of Proprioceptive Feedback

Experimental evidence indicates that there is a direct relation between the muscle force and activities of cortical motor neurons [152, 87]. Particularly in [152], it has been shown that proprioception affects firing rates of cortical motor neurons significantly and enhances performance of BMIs in on-line operation. Therefore, here we studied firing rate of the cortical motor neuron in the closed-loop and the partial closed-loop-V designs for the nominal system. By analyzing firing patterns in both systems, we investigated a role of proprioceptive feedback in designing closed-loop neural prosthetic system.

It is known that a part of afferent fibers provides the proprioceptive feedback information directly to M1 neurons by making synaptic connections with them. Therefore, we included the proprioceptive feedback current, $I_p(t) = K_\tau \tau(t) + K_\theta \dot{\theta}(t)$, directly to the cortical motor neuron. We set proprioceptive feedback gains $K_\tau$ and $K_\theta$ to 1 and 100 respectively in the closed-loop system design. Further, we assumed that it is possible to reach the desired extension of the finger in the absence and the presence of proprioceptive feedback. With this assumption, we solved the control problem (equation (4.6)) for the closed-loop system and the partial closed-loop-V system under

nominal conditions i.e. in the absence of noise and computed optimal external input currents for both systems. We implemented these optimal external input currents on the nominal closed-loop and partial closed-loop-V system and computed ISIs of the cortical motor neuron firings for the reaching task. Figure 4.3 shows firing patterns, shown here in terms of ISIs, for both systems.

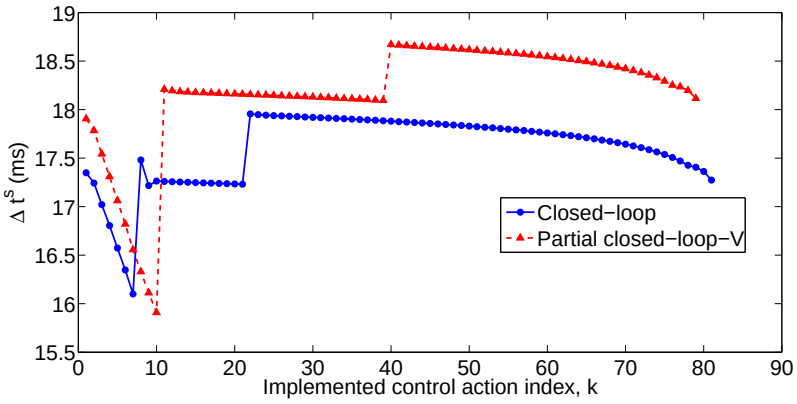



Figure 4.3: Modulations in inter-spike intervals (ISIs) of the cortical motor neuron in the presence and the absence of proprioception for the extension movement of the prosthetic joint by 20°. Here the y-axis represents ISIs variations ($\Delta t^s$ in milliseconds) and the x-axis represents the control action index ($k$).

Figure 4.3 clearly shows that ISIs of the cortical motor neuron decreased after including the proprioceptive feedback in the design (i.e. increased in the firing rate). It is also shown in Figure 4.3 that these decrements in ISIs are not significant to conclude necessity of proprioception in enhancing the performance of neuroprosthetic systems. We suspect that a possible reason for this small decrement may be the small gain value of $K_\tau$ used in the proprioceptive feedback. Therefore, next, we studied how the activity of the cortical motor neuron changes as we change the value of the gain $K_\tau$ in the proprioceptive feedback. This is equivalent to weakening or strengthening afferent fibers which carry the proprioceptive feedback back to the brain. For this,

we fixed the gain $K_\theta$ to 100 and varied the gain $K_\tau$. It should be noted that the gain $K_\tau$ carries torque information in the proprioceptive feedback current model and resembles the perception of the torque exerted by the muscle at the joint. With this, we studied variation in ISIs as a function of the gain $K_\tau$ for the closed-loop system. Figure 4.4 shows ISIs for $K_\tau = 1$, 5, and 10.

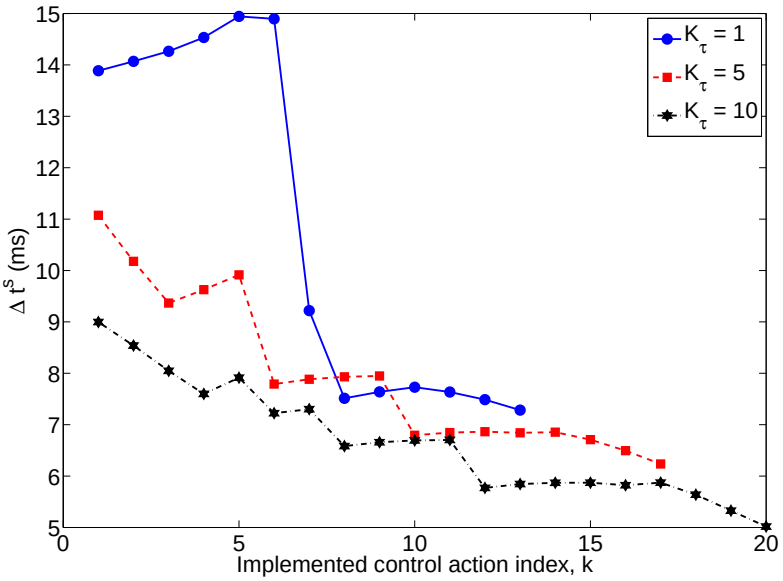



Figure 4.4: Modulations in inter-spike intervals (ISIs) of the cortical motor neuron in presence of three different proprioceptive feedbacks for the extension movement of the prosthetic joint by 10°. Here the closed-loop system includes proprioception as well visual feedbacks. Increasing values of the gain $K_\tau$ resemble more perception of the required torque for accomplishing the task. The y-axis represents ISIs variations ($\Delta t^s$ in milliseconds) and the x-axis represents the control action index ($k$).

Figure 4.4 shows that ISIs of the cortical motor neuron changed significantly when we increased the torque perceived by the proprioceptive feedback current.

## 4.6 Concluding Remarks

In this chapter, we have proposed a generalized optimal control framework using a model-based receding horizon control policy to rigorously analyze neuroprosthetic systems by including various sensory feedback pathways in the system. This is the first systematic attempt to formalize a system-level closed-loop neuroprosthesis analysis in an optimal predictive control framework. Using this framework, a minimum time control problem has been formulated to elucidate the importance of sensory feedback pathways in a single joint prosthetic arm movement controlled by the firing activity of a single cortical motor neuron. From our results we conclude that visual feedback is important in rejecting internal noises, naturally occurs in neuronal network, while reaching the desired goal of the task. Our results clearly indicate significant advantages of using an optimal control framework in studying complex biological systems such as neuroprosthetic systems which may be difficult in an experimental framework.

# Chapter 5

# Design of Closed-loop Brain-Machine Interfaces: An Optimal Control Approach

## 5.1 Introduction

In the previous chapter, we proposed a generalized optimal control-theoretic framework for a system-level rigorous analysis of closed-loop neuroprosthetic systems using a model-based receding horizon control policy [95]. The formulation of a minimum time control problem within this framework for accomplishing reaching tasks in a minimum time demonstrated the capability of the optimal controller in designing higher level motor planning. In this chapter, we show that the framework can be modified to design missing sensory feedback pathways in brain-machine interfaces (BMIs) optimally and thus to close the loop in BMIs for developing stimulus-enhanced next

generation BMIs as motivated in Chapter 1. In particular, we theoretically demonstrate the recovery of closed-loop performance of a BMI for voluntary single joint extension task by designing an optimal artificial sensory feedback in the absence of the natural proprioceptive feedback pathways.

Using synthetic data obtained through the simulation of an experimentally validated psycho-physiological cortical circuit model for voluntary single joint reaching task [19], we design a brain-machine interface (BMI). We analyze the performance of the BMI in the presence and the absence of natural proprioceptive feedback information. Through simulation, we show that the BMI performance degrades significantly in the absence of the natural proprioception. Throughout our analysis, we exclude the treatment of visual feedback as well any form of cortical learning. Finally, we design an optimal artificial sensory feedback in the receding horizon control framework to stimulate appropriate cortical sensory area neurons and thus to recover the natural performance of the reaching task during the online operation of the designed BMI.

## 5.2 A Psycho-physiological Cortical Circuit Model

Figure 5.1 shows a psycho-physiological cortical circuit model, proposed by Bullock et al. [19], for voluntary control of a single joint movement. This minimal model captures the essential cortical pathways as well as the proprioceptive feedback pathways which are relevant during voluntary extension or flexion of a single joint such as elbow. Although the model excludes the treatment of visual feedback during the movement, the model has shown its capability in a qualitative reproduction of several experimentally observed results on voluntary control of a single joint movement. The details of the model and its connection with neurophysiology of a single joint

voluntary movement can be found in [19].

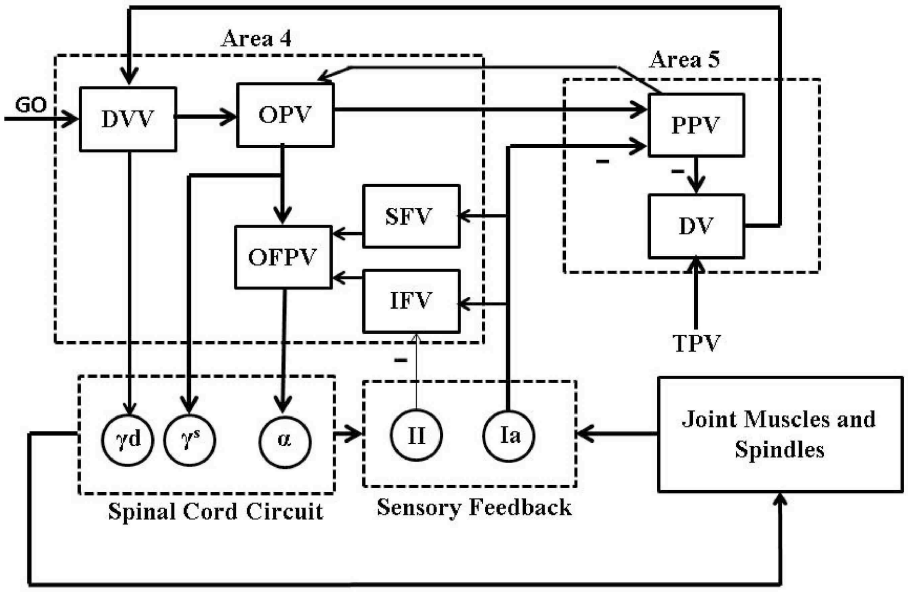



Figure 5.1: A psycho-physiological cortical circuit model for voluntary control of single joint movement: The diagram has been redrawn from Bullock et al. [19], Figure 1.1. Nomenclature (adopted from [19]): "GO" is a scalable gating signal; "DVV" is the desired velocity vector; "OPV" is the outflow position vector; "OFPV" is the outflow force and position vector; "SFV" is the static force vector; "IFV" is the inertial force vector; "PPV" is the perceived position vector; "DV" is the difference vector; "TPV" is the target position vector; "$\gamma^d$" and "$\gamma^s$" are dynamic and static gamma motoneurons respectively; "$\alpha$" is alpha motoneuron; "$Ia$" and "$II$" are type Ia and II afferent fibers; $-$ represents inhibitory feedback. The rest of the connections are excitatory.

Briefly, a population of area 5 ("DV") neurons computes the difference between the target and the perceived limb position vectors. The average firing activity of a population of these neurons is represented as

$$r_i(t) = \max\{T_i - x_i(t) + B^r, 0\}. \tag{5.1}$$

Here, $0 \leq r_i(t) \leq 1$ represents the average firing activity of a population of "DV"

neurons associated with the agonist muscle $i$ and shows a phasic behavior during the movement. Throughout the paper, we will denote the average firing activity of neurons associated with the agonist muscle $i$ by the subscript $i$ and the corresponding antagonist muscle by the subscript $j$. $T_i$ is the target position vector ("TPV") command for the target position of the agonist muscle $i$. $x_i(t)$ is the average firing activity of a population of area 5 "PPV" neurons. These neurons continuously compute the present position of the agonist muscle $i$. $B^r$ is the base firing activity of the "DV" neurons. Continuously computed difference vector information by the area 5 "DV" neurons is then scaled by a population of area 4 "DVV" neurons as

$$u_i(t) = \max\{g(t).(r_i(t) - r_j(t)) + B^u, 0\}. \tag{5.2}$$

Here, $u_i(t)$ is the average firing activity of a population of area 4 "DVV" neurons. $B^u$ is the base firing activity of the "DVV" neurons. $g(t)$ is an internal "GO" signal which is assumed to be originated from the basal ganglia. "DVV" neurons fire only during the movement and thus their average firing activity shows a phasic-movement time (MT) behavior. The dynamics of the internal "GO" signal is modeled as

$$\frac{\mathrm{dg^1(t)}}{\mathrm{dt}} = \epsilon(-g^1(t) + (C - g^1(t))g^0), \tag{5.3a}$$

$$\frac{\mathrm{dg^2(t)}}{\mathrm{dt}} = \epsilon(-g^2(t) + (C - g^2(t))g^1(t)), \tag{5.3b}$$

$$g(t) = g^0\frac{g^2(t)}{C}. \tag{5.3c}$$

Here, $\epsilon$ represents a slow integration rate and is treated as constant. $C$ is a constant

value at which the "GO" neurons saturate. The area 4 "OPV" neurons receive information from the area 4 "DVV" neurons as well as the area 5 "PPV" neurons and show tonic firing activity. The average firing activity of a population of "OPV" neurons is modeled as

$$\begin{aligned}\frac{\mathrm{dy_i(t)}}{\mathrm{dt}} &= (1 - y_i(t))(\eta x_i(t) + \max\{u_i(t) - u_j(t), 0\}) \\ &\quad - y_i(t)(\eta x_j(t) + \max\{u_j(t) - u_i(t), 0\}).\end{aligned} \tag{5.4}$$

Here, $\eta$ is a scaling factor. The average firing activity of a population of static ($\gamma_i^S(t)$) and dynamic ($\gamma_i^D(t)$) gamma motoneurons are modeled as

$$\gamma_i^S(t) = y_i(t), \tag{5.5a}$$

$$\gamma_i^D(t) = \rho \max\{u_i(t) - u_j(t), 0\}. \tag{5.5b}$$

Here, $\rho$ is a scaling parameter. The average firing activity of the primary ("Ia") and the secondary ("II") muscle spindles afferents are modeled as

$$s_i^1(t) = S(\theta \max\{\gamma_i^S(t) - p_i(t), 0\} + \phi \max\{\gamma_i^D(t) - \frac{\mathrm{dp_i(t)}}{\mathrm{dt}}, 0\}), \tag{5.6a}$$

$$s_i^2(t) = S(\theta \max\{\gamma_i^S(t) - p_i(t), 0\}). \tag{5.6b}$$

Here, $s_i^1(t)$ and $s_i^2(t)$ are the primary and the secondary spindles afferents average firing activity respectively. $p_i$ is the position of the agonist muscle $i$. $\theta$ is the sensitivity of the static nuclear bag and chain fibers. $\phi$ is the sensitivity of the dynamic nuclear

bag fibers. The saturation of spindles afferents activity is given by the function $S(\omega) = \omega/(1 + 100\omega^2)$. The average firing activity $x_i(t)$ of a population of area 5 "PPV" neurons is modeled as

$$\begin{aligned}\frac{\mathrm{dx_i(t)}}{\mathrm{dt}} &= (1 - x_i(t)) \max\{\Theta y_i(t) + s_j^1(t-\tau) - s_i^1(t-\tau), 0\} \\ &\quad - x_i(t) \max\{\Theta y_j(t) + s_i^1(t-\tau) - s_j^1(t-\tau), 0\}.\end{aligned} \tag{5.7}$$

Here, $\tau$ is the delay time of the spindles feedback. $\Theta$ is a constant gain. The average firing activity $q_i(t)$ of a population of area 4 "IFV" neurons is modeled as

$$q_i(t) = \lambda_i \max\{s_i^1(t-\tau) - s_i^2(t-\tau) - \Lambda, 0\}. \tag{5.8}$$

Here, $\Lambda$ is a constant threshold. The average firing activity $f_i(t)$ of a population of area 4 "SFV" neurons is modeled as

$$\frac{\mathrm{df_i(t)}}{\mathrm{dt}} = (1 - f_i(t))hs_i^1(t-\tau) - \psi f_i(t)(f_j(t) + s_j^1(t-\tau)). \tag{5.9}$$

Here, $h$ is a constant gain which controls the strength and speed of an external load compensation. $\psi$ is an inhibitory scaling parameter. The average firing activity $a_i(t)$ of a population of the area 4 "OFPV" neurons is modeled as

$$a_i(t) = y_i(t) + q_i(t) + f_i(t). \tag{5.10}$$

The average firing activity of these neurons shows a phasic-tonic behavior. The

average firing activity $\alpha_i(t)$ of alpha motoneurons is modeled as

$$\alpha_i(t) = a_i(t) + \delta s_i^1(t), \tag{5.11}$$

where $\delta$ is a stretch reflex gain. The limb dynamics is described by

$$\frac{\mathrm{d^2p_i(t)}}{\mathrm{dt^2}} = \frac{1}{I}(M(c_i(t) - p_i(t)) - M(c_j(t) - p_j(t)) + E_i - V\frac{\mathrm{dp_i(t)}}{\mathrm{dt}}). \tag{5.12}$$

Here $p_i(t)$ is the position of the agonist muscle $i$ within its range of origin-to-insertion distances. $p_j(t)$ is the position of the antagonist muscle such that $p_i(t) + p_j(t) = 1$. $I$ is the moment of inertia of the limb. $V$ is the joint viscosity. $E_i$ is the external force applied to the joint. $M(c_i(t), p_i(t)) = \max\{c_i(t) - p_i(t), 0\}$ represents the total force generated by the agonist muscle $i$. $c_i(t)$ is the muscle contraction activity dynamics of which is given by

$$\frac{\mathrm{dc_i(t)}}{\mathrm{dt}} = \nu(-c_i(t) + \alpha_i(t)). \tag{5.13}$$

In this chapter, we use this model in generating synthetic experimental data for voluntary control of a single joint extension task and use these data in designing a closed-loop brain-machine interface.

## 5.3 Closed-Loop Brain-Machine Interface Design

### 5.3.1 Data Generation

In a typical non-human primate experiment, a monkey is trained to accomplish a given motor task such as reaching or grasping. After the training, spiking activity of

single neurons are recorded through implanted multi-channel electrodes from various motor relevant cortical areas such as the primary motor cortex (M1), the premotor area (PMv, PMd), and the primary somatosensory area (S1). Simultaneously, kinetic and kinematic information such as joint torque, velocity and position of the real arm are measured to generate a data set.

Here, we generated a synthetic experimental data set for voluntary control of a single joint extension task by simulating the system model shown by equations (5.1) - (5.13) in MATLAB. The target position of the agonist muscle $i$ was set to the desired one at $t = 0$. The "GO" signal was turned on at $t = 50$ ms. During the initial 50 ms, the system was at the priming state. The initial condition of variables was set to 0 except $x_i(0) = x_j(0) = 0.5$, $y_i(0) = y_j(0) = 0.5$, $p_i(0) = p_j(0) = 0.5$, $u_i(0) = u_j(0) = B^u$ and $r_i(0) = r_j(0) = B^r$. For the simulation, we used the following model parameters [19]: $I = 200$, $V = 10$, $\nu = 0.15$, $B^r = 0.1$, $B^u = 0.01$, $\Theta = 0.5$, $\theta = 0.5$, $\phi = 1$, $\eta = 0.7$, $\rho = 0.04$, $\lambda_1 = 150$, $\lambda_2 = 10$, $\Lambda = 0.001$, $\delta = 0.1$, $C = 25$, $\epsilon = 0.05$, $\psi = 4$, $h = 0.01$, $T_1 = 0.7$ and $\tau = 0$.

In BMI experiments, a trial is considered successful if the trained monkey accomplishes the specified motor task in a given time duration. Thus the accomplishment time of the task in successful trials is allowed to vary. In our case, the "GO" signal controls the velocity of the joint movement and thus the accomplishment time of a given task. Therefore we assumed that there is a trial-to-trial variability in the internal "GO" signal. To introduce the trial-to-trial variability in the "GO" signal, we modeled $g^0$ as a Gaussian distributed random variable with mean 0.75 and variance 0.0025. For a given trial, $g^0$ is constant. It should be noted that the "GO" signal has no effect on the accuracy of the movement.

We simulated the model and generated synthetic experimental data for 1600 independent trials of the voluntary single joint extension task. In each of these trials, the simulation was performed for the duration of 1.45 seconds which includes a variable holding period at the target position after the accomplishment of the task. To generate a synthetic experimental data set, we measured the average firing activity of a population of area 4 "DVV", "OPV", and "OFPV" agonist and the corresponding antagonist neurons sampled at every 10 ms. Simultaneously, we measured the total force difference between the agonist and the corresponding antagonist muscle i.e. $\Delta M(k) = M(c_i(t), p_i(t)) - M(c_j(t), p_j(t))$, the agonist muscle position $p_i(k)$, and the agonist muscle velocity $v_i(k) = \frac{\mathrm{dp_i(t)}}{\mathrm{dt}}(k)$. Here, $k = 1, 2, \cdots$ is a discrete sample time at which data were recorded for a given trial. With this, we created a data set of 233600 samples by embedding the recorded data from 1600 trials.

### 5.3.2 Decoder

In Chapter 1, we discussed decoder models such as a discrete-time Weiner filter and the Kalman filter which are typical used in BMI studies to extract motor information from continuously recorded spike trains of the cortical area 4 neurons. In this section, we use both the Weiner filter and the Kalman filter based decoder models to extract the total force difference between the agonist and the corresponding antagonist muscle ($\Delta M(k)$), the agonist muscle position $p_i(k)$, and the agonist muscle velocity $v_i(k)$ from the recorded average firing activity of the area 4 "DVV", "OPV", and "OFPV" neurons. It should be noted that these neurons have direct contribution to the spinal cord circuit of the real system shown in Figure 5.1.

**Weiner Filter**

In a discrete-time adapted Weiner filter based decoder design [89] as described in Chapter 1 (see equations (1.8), (1.9) and (1.10) in Chapter 1), the relation between $\Delta M(k)$ and the average firing activity of area 4 neurons i.e. "DVV", "OPV", and "OFPV" neurons can be expressed as

$$\Delta M(k) = \mathbf{w}^T \mathbf{z}(k). \tag{5.14}$$

Here, $\mathbf{w}$ is a $(L.N) \times 1$ weight vector. $L$ is the number of delay elements. $(\cdot)^T$ is the transpose of a vector. $\mathbf{z}(k) = [z_1(k), z_1(k-1), \cdots, z_1(k-L+1), z_2(k), \cdots, z_N(k-L+1)]^T$. $z_m(k-l)$ represents the average firing activity of the population $m$ delayed by $l$ samples. For our system, $z_1 = y_i$, $z_2 = y_j$, $z_3 = u_i$, $z_4 = u_j$, $z_5 = a_i$, and $z_6 = a_j$. Thus, $N = 6$. We assume the number of delay elements $L = 10$. Thus the weight vector $\mathbf{w}$ has a dimension of $60 \times 1$. We also assume that there is no measurement noise in obtaining data i.e. $n_1(k) = 0$ in equations (1.8) and (1.9). Similarly, the relation between $[p_i(k), v_i(k)]^T$ and the average firing activity of area 4 neurons i.e. "DVV", "OPV", and "OFPV" neurons can be expressed as

$$[p_i(k), v_i(k)]^T = \mathbf{W}^T \mathbf{z}(k). \tag{5.15}$$

Here, $\mathbf{W}^T$ is a weight matrix of dimension $60 \times 2$.

For consistency with BMI experiments, in this work, we used 220000 samples of the recorded synthetic data to train the weight vector '$\mathbf{w}$' and the weight matrix '$\mathbf{W}$' of the designed decoders (equations (5.14) and (5.15)). For this, we used the following normalized least mean squares algorithm [89] (see equation (1.10) in Chapter 1):

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\eta}{\beta + ||\mathbf{z}(k)||^2}\mathbf{z}(k)e(k). \tag{5.16a}$$

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \frac{\eta}{\beta + ||\mathbf{z}(k)||^2}\mathbf{z}(k)(\mathbf{e}(k))^T. \tag{5.16b}$$

Here, $\eta$ and $\beta$ are constants. $||\cdot||$ represents the Euclidean norm. In equation (5.16a), $e(k)$ represents a scalar error between the recorded $\Delta M(k)$ and the estimated value through equation (5.14). In equation (5.16b), $\mathbf{e}(k)$ represents the error vector between the recorded $[p_i(k), v_i(k)]^T$ and the estimated value through equation (5.15). For our study, we set $\eta = 0.01$ and $\beta = 1$. After the training, we froze the weight vector '$\mathbf{w}$' and the weight matrix '$\mathbf{W}$' to the final adapted value. Then we used the rest of 13600 samples to validate the performance of both decoders. Figure 5.2 and Figure 5.3 show the offline performance of the adapted Weiner filter based decoder defined by equation (5.14) and equation (5.15) respectively on the test data for 1000 samples.

**Kalman Filter**

As described in Chapter 1 (see equations (1.13), (1.14), (1.15), (1.16), (1.17) and (1.18) in Chapter 1), the Kalman filter based decoder design is given by

$$\hat{x}(k \mid k-1) = A\hat{x}(k-1), \tag{5.17a}$$

$$\hat{P}(k \mid k-1) = A\hat{P}(k-1)A^T + R, \tag{5.17b}$$

$$\hat{x}(k) = \hat{x}(k \mid k-1) + K_k(\mathbf{z}(k) - C\hat{x}(k \mid k-1)), \tag{5.17c}$$

$$\hat{P}(k) = (I - K_k C)\hat{P}(k \mid k-1), \tag{5.17d}$$

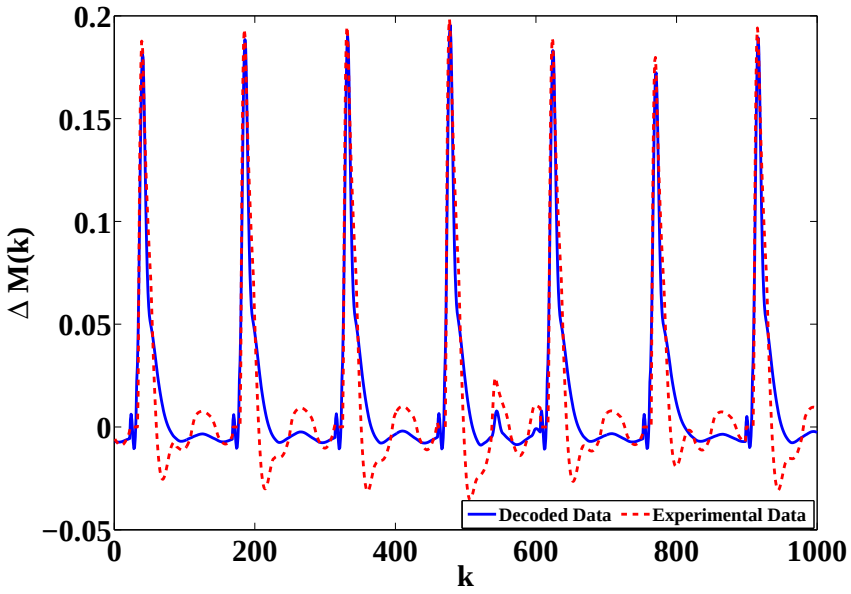
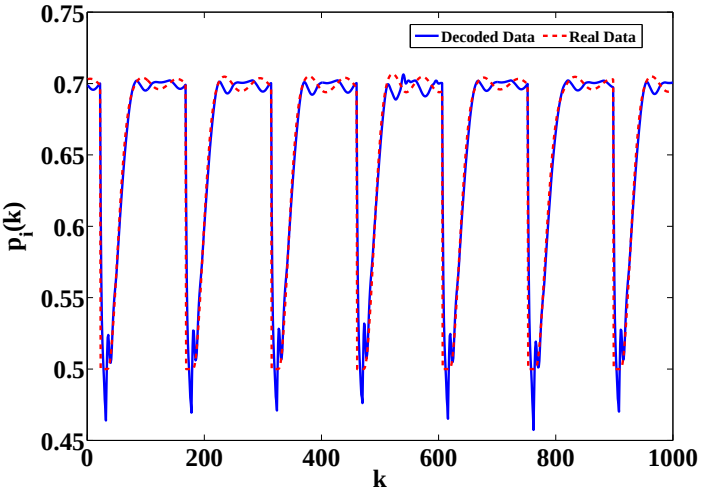
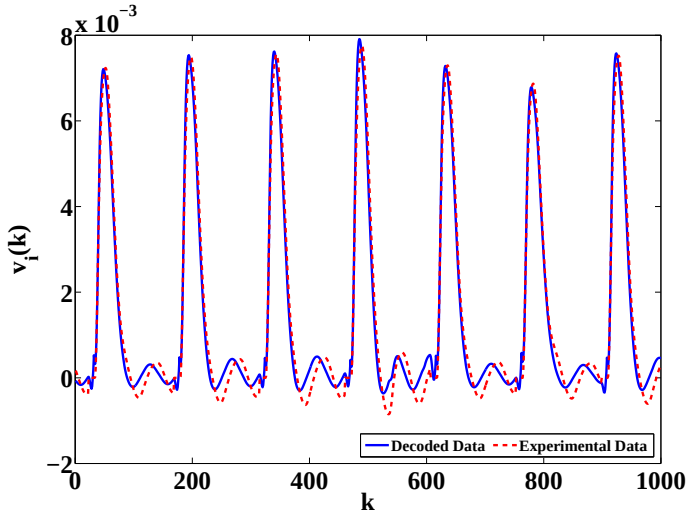



Figure 5.2: The actual (dotted line) and the estimated (solid line) force difference between the agonist and the corresponding antagonist muscle, $\Delta M(k)$, using the Weiner filter based decoder (equation (5.14)) for the single joint reaching task on a sample part of the test data.




(a)




(b)

Figure 5.3: The actual (dotted line) and the estimated (solid line) (a) position, $p_i(k)$, and (b) velocity, $v_i(k)$, of the agonist muscle using the Weiner filter based decoder (equation (5.14)) for the single joint reaching task on a sample part of the test data.

$$K_k = \hat{P}(k \mid k-1)C^T(C\hat{P}(k \mid k-1)C^T + Q)^{-1}. \tag{5.17e}$$

Here, $\hat{x}(k \mid k-1)$ and $\hat{x}(k)$ represent *a priori* and *a posteriori* estimate of the state vector $x(k)$ of dimension $p \times 1$ at time $k$ respectively. $\hat{P}(k \mid k-1)$ and $\hat{P}(k)$ are the estimate of *a priori* and *a posteriori* covariance matrix respectively. $\mathbf{z}(k)$ is the observation (firing rate) vector of dimension $r \times 1$. $K_k$ is the Kalman gain and $I$ is an identity matrix. $A \in \mathbb{R}^{p \times p}$ is the state matrix and is given by $A = X_2 X_1^T (X_1 X_1^T)^{-1}$. $C \in \mathbb{R}^{r \times p}$ represents the observation matrix and is given by $C = ZX^T(XX^T)^{-1}$. $R = \frac{1}{D-1}(X_2 - AX_1)(X_2 - AX_1)^T$ and $Q = \frac{1}{D}(Z - CX)(Z - CX)^T$ are covariance matrices of Gaussian noise sources with mean zero to the state and the observation vectors respectively. $Z$, $Y$, $X_1$, and $X_2$ are given by $Z = \begin{pmatrix} z_{1,1} & \cdots & z_{1,D} \\ \vdots & \ddots & \vdots \\ z_{r,1} & \cdots & z_{r,D} \end{pmatrix}$, $X = \begin{pmatrix} x_{1,1} & \cdots & x_{1,D} \\ \vdots & \ddots & \vdots \\ x_{p,1} & \cdots & x_{p,D} \end{pmatrix}$, $X_1 = \begin{pmatrix} x_{1,1} & \cdots & x_{1,D-1} \\ \vdots & \ddots & \vdots \\ x_{p,1} & \cdots & x_{p,D-1} \end{pmatrix}$, and $X_2 = \begin{pmatrix} x_{1,2} & \cdots & x_{1,D} \\ \vdots & \ddots & \vdots \\ x_{p,2} & \cdots & x_{p,D} \end{pmatrix}$ respectively. Here, $z_{i,j}$ represents the $j^{th}$ firing rate data of the $i^{th}$ neuron. $x_{i,j}$ represents the $j^{th}$ data of the $i^{th}$ state.

For our system, $x(k) \equiv \Delta M(k)$ $(p = 1)$ if the total force difference between the agonist and the corresponding antagonist muscle $(\Delta M(k))$ is extracted and $x(k) \equiv [p_i(k), v_i(k)]^T$ $(p = 2)$ if the position and the velocity of the agonist muscle are extracted from the average firing activity of area 4 neurons i.e. "DVV", "OPV", and "OFPV" neurons. $r = 6$ and $D = 220000$. $\mathbf{z}(k) = [y_i(k), y_j(k), u_i(k), u_j(k), a_i(k), a_j(k)]^T$.

For consistency with the Weiner filter design, we used 220000 samples of the

recorded synthetic data to compute $A$, $C$, $R$, and $Q$ for both $x(k) \equiv \Delta M(k)$ and $x(k) \equiv [p_i(k), v_i(k)]^T$. Then we used the rest of 13600 samples to validate the performance of the decoder for both cases. Figure 5.4 shows the offline performance of the Kalman filter based decoder on the test data for 1000 samples when $x(k) \equiv \Delta M(k)$ and Figure 5.5 shows the offline performance of the Kalman filter based decoder on the test data for 1000 samples when $x(k) \equiv [p_i(k), v_i(k)]^T$. Clearly, the Kalman filter based decoders performed better than the Weiner filter based decoders on the test data.

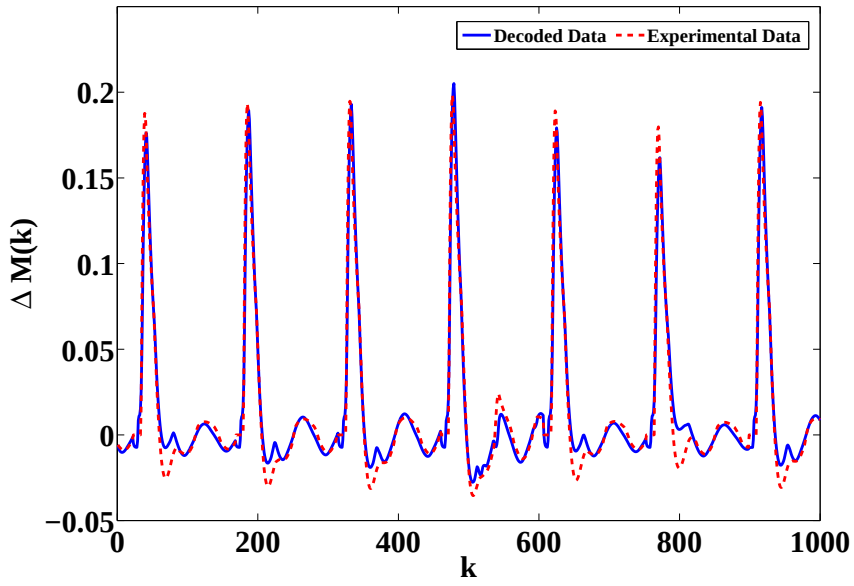
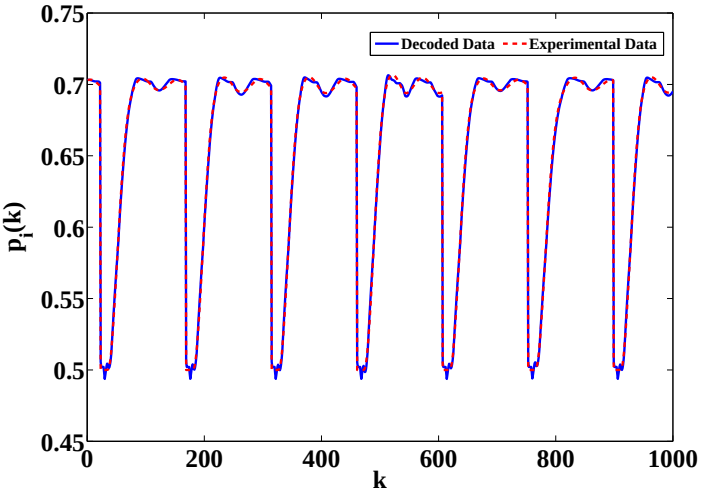
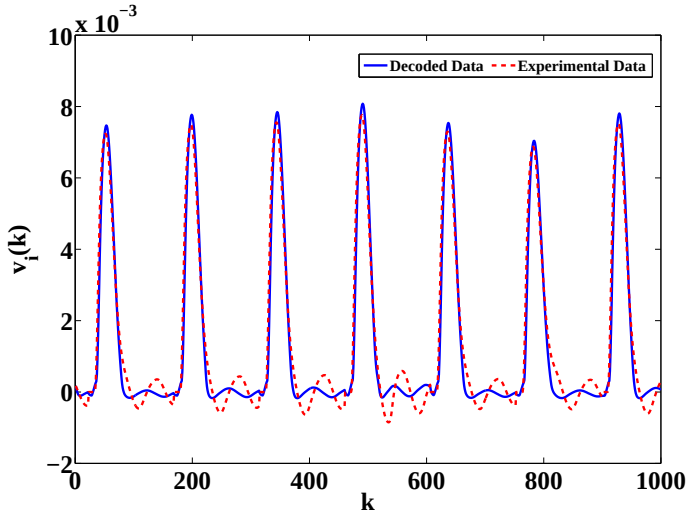



Figure 5.4: The actual (dotted line) and the estimated (solid line) force difference between the agonist and the corresponding antagonist muscle, $\Delta M(k)$, using the Kalman filter based decoder for the single joint reaching task on a sample part of the test data.




(a)




(b)

Figure 5.5: The actual (dotted line) and the estimated (solid line) (a) position, $p_i(k)$, and (b) velocity, $v_i(k)$, of the agonist muscle using the Kalman filter based decoder for the single joint reaching task on a sample part of the test data.

### 5.3.3 Need of a Closed-loop BMI

In this section, we first study the performance of the designed decoders in the previous section in an open-loop BMI system design shown in Figure 5.6. We show that the performance of these decoders degrades substantially in the absence of the natural proprioceptive feedback information. It should be noted here that we are not considering the visual feedback or any form of learning in this work for the online trajectory corrections during the movement.

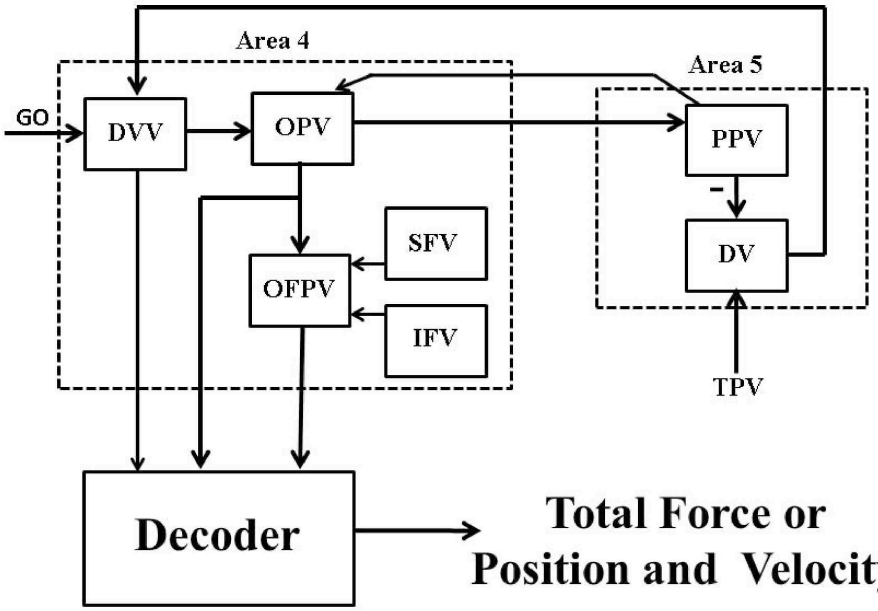



Figure 5.6: An open-loop BMI system design.

As shown in Figure 5.6, the average firing activity of area 4 neurons i.e. "DVV", "OPV", and "OFPV" neurons are used by the decoder to extract either the total force ($\Delta M(k)$) or the position ($p_i(k)$) and the velocity ($v_i(k)$) of the agonist muscle. To compare the performance of both decoders (the Weiner filter and the Kalman filter), we simulate equations (5.1) - (5.9) along with the particular decoder model.

We set $g^0 = 0.75$, $\theta = 0$ and $\rho = 0$ in equations (5.1) - (5.9). Rest of the model parameters are same as given in section 5.3.1.

Figure 5.7 compares the open-loop performance of the Weiner filter based decoder and the Kalman filter based decoder with the performance of the closed-loop real system shown in Figure 5.1 when the extracted information from both the decoders was the total force ($\Delta M(t)$) in real time $t$.

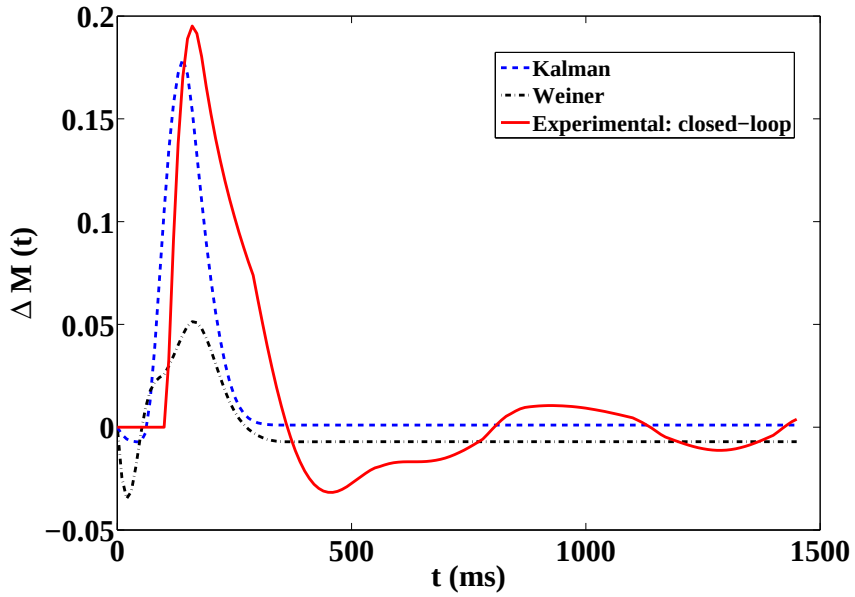



Figure 5.7: Comparison of the open-loop performance of the Weiner filter based decoder and the Kalman filter based decoder with the performance of the closed-loop real system shown in Figure 5.1 when the extracted information from both decoders was the force difference between the agonist and the corresponding antagonist muscle, $\Delta M(t)$.

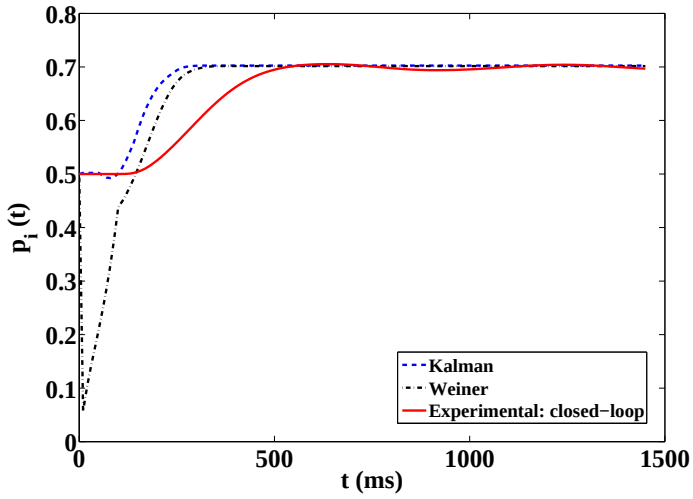
As shown in Figure 5.7, the performance of both decoders degraded substantially in the absence of the natural proprioception information. Moreover, the Kalman filter performed better than the Weiner filter based decoder. Figure 5.8 compares the open-loop performance of the Weiner filter based decoder and the Kalman filter

based decoder with the performance of the closed-loop real system shown in Figure 5.1 when the extracted information from both decoders was the position ($p_i(t)$) and the velocity ($v_i(t)$) of the agonist muscle in real time $t$. As shown clearly in this figure, the performance of both decoders degraded substantially in the absence of the natural proprioception information. Moreover, the Kalman filter showed better performance than the Weiner filter in decoding the position compared to the velocity of the agonist muscle.

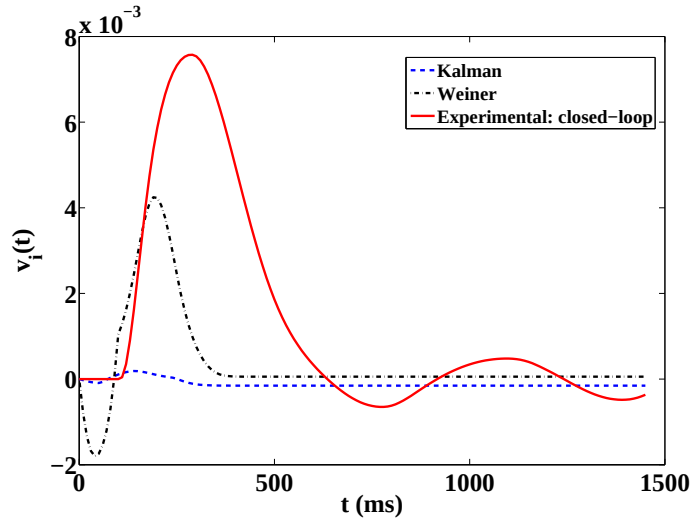
Next, we consider the Weiner filter based decoder with $\Delta M(k)$ as the extracted information and study the online performance of this decoder in the presence and the absence of the natural proprioceptive feedback information when the decoder interacts with the dynamics of the muscle as shown in Figure 5.9.

To make a realistic comparison of the performance of the decoder with the real system, we first study the performance of the real system shown in Figure 5.1 in the presence and the absence of the natural proprioceptive feedback i.e. the sensory feedback. For both cases, we simulate equations (5.1) - (5.13) with $g^0 = 0.75$. Rest of the model parameters are same as given in section 5.3.1 for both cases except $\theta = 0$ and $\rho = 0$ in the case of no proprioception. This means that the primary ("Ia") and the secondary ("II") muscle spindles afferents become inactive (see equation 5.6) in the absence of proprioception. The top plot of Figure 5.10 shows the position trajectory of the agonist muscle $i$ in the presence and the absence of proprioception.

As shown in the top plot of Figure 5.10, the desired position of the agonist muscle $i$ has been achieved in both cases for the real system. The result is consistent with a prior neurophysiological experiment where it was shown that a trained monkey (in the absence of visual feedback) can reach the desired target position in the presence




(a)




(b)

Figure 5.8: Comparison of the open-loop performance of the Weiner filter based decoder and the Kalman filter based decoder with the performance of the closed-loop real system shown in Figure 5.1 when the extracted information from both decoders was (a) the position ($p_i(t)$) of the agonist muscle, and (b) the velocity ($v_i(t)$) of the agonist muscle.

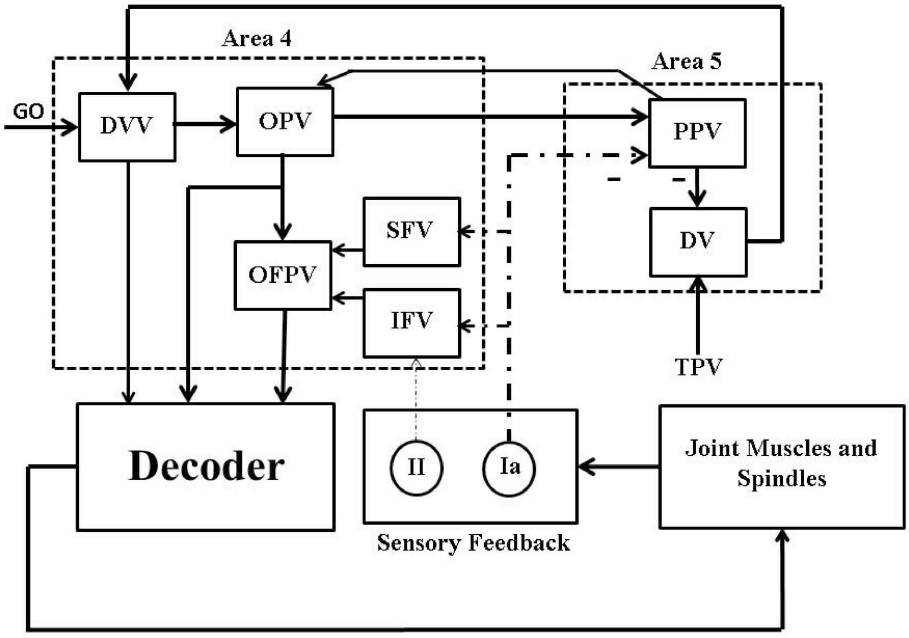



Figure 5.9: A closed-loop BMI system design using the natural proprioceptive feedback information (sensory feedback).

and the absence of proprioception [13].

Next we study the performance of the closed-loop BMI (decoder) (in the presence of the natural proprioceptive feedback information) and the open-loop BMI (decoder) (in the absence of the natural proprioceptive feedback information) shown in Figure 5.9. For this, we simulate equations (5.1) - (5.10), equation 5.12, and equation 5.14. Here we assumed that the limb dynamics is same for the real and the prosthetic system. For the open-loop BMI, we again set $\theta = 0$ and $\rho = 0$. The bottom plot of Figure 5.10 shows the position trajectory of the agonist muscle $i$ for the closed-loop and the open-loop BMI.

It is clear from the bottom plot of Figure 5.10 that the decoder performance degrades substantially when the decoder, trained with the closed-loop data, is applied on the open-loop system. Since the decoder was trained with the closed-loop firing

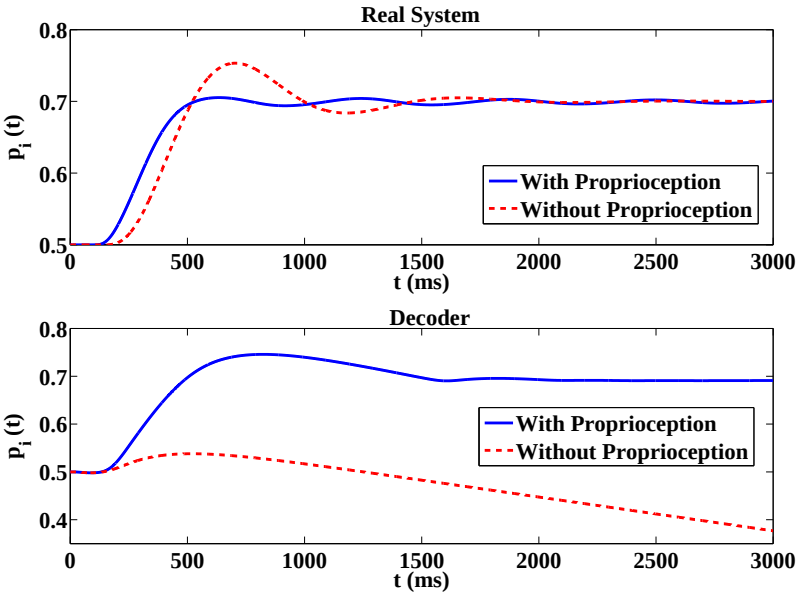



Figure 5.10: The position trajectory of the agonist muscle $i$ as a function of time in the presence (solid line) and the absence (dotted line) of the natural proprioceptive feedback information: The top plot shows the position trajectory for the real system shown in Figure 5.1. The bottom plot shows the position trajectory for the prosthetic system. The desired position target ($T_i$) for the agonist muscle $i$ is 0.7.

activity of the area 4 "OPV", "DVV" and "OFPV" neurons, the firing activity of these neurons must have changed significantly in the absence of the natural proprioception feedback information. To see this, we plot the firing activity of these neurons in the presence and the absence of the natural proprioception feedback information for the BMI design shown in Figure 5.9. Figure 5.11 shows the firing activity of the area 4 "OPV", "DVV" and "OFPV" neurons and the area 5 "PPV" neurons in the presence and the absence of the natural proprioceptive feedback information.

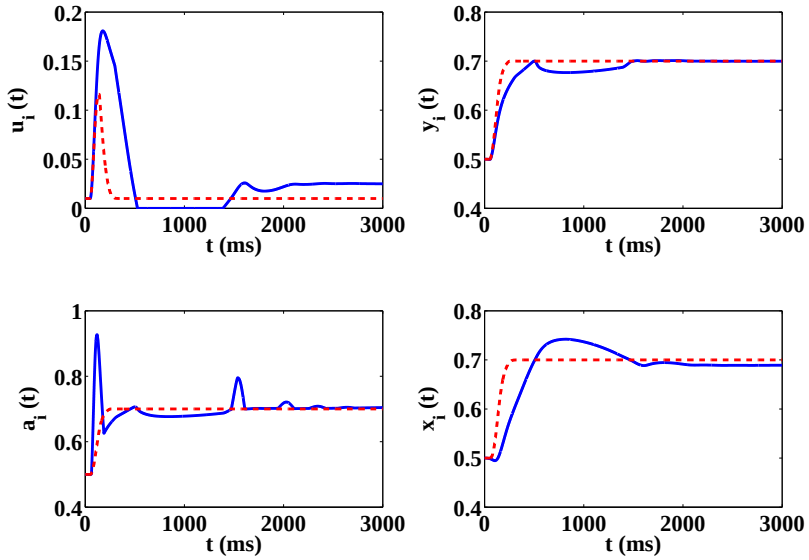



Figure 5.11: The average firing activity of a population of agonist "DVV" ($u_i(t)$), "OPV" ($y_i(t)$), "OFPV" ($a_i(t)$) and "PPV" ($x_i(t)$) neurons in the presence (solid line) and the absence (dotted line) of the natural proprioceptive feedback information.

As shown in Figure 5.11, the firing activity of cortical neurons deviates significantly from the closed-loop activity in the absence of the natural proprioceptive feedback information. Since the weights of the designed decoder were not adapted to accommodate these significant deviations in the firing activity of the area 4 neurons,

the decoder performance degrades substantially in the absence of the natural proprioceptive feedback information. These results clearly show that there is a necessity for designing an artificial proprioceptive feedback to regain the closed-loop performance of the designed decoder in the absence of the natural proprioceptive feedback.

### 5.3.4 Artificial Proprioceptive Feedback Design

As shown in Figure 5.1, the area 5 "PPV" neurons receive the position feedback information through the primary (Ia) muscle spindles afferents. These neurons then use this proprioceptive feedback information to compute the present position vector command. In the absence of the natural proprioceptive feedback pathways, this feedback information is lost. In order to compensate the lost feedback information of the area 5 "PPV" neurons, we design an artificial sensory feedback in a model-based optimal receding horizon control (RHC) framework. The goal is to recover the closed-loop performance of the decoder (the Weiner filter based decoder with $\Delta M(k)$ as the extracted information) by providing the designed optimal artificial sensory feedback to the "PPV" neurons in the absence of the natural proprioceptive feedback pathways. It should be noted that we are not designing artificial feedback to compensate the loss of sensory feedback to the area 4 "IFV" and "SFV" neurons in this study. Thus in the absence of the natural sensory feedback, these neurons remain inactive during our analysis.

In order to recover the closed-loop (natural) performance of the decoder, we formulate two control problems in this section. In the first problem, we design an optimal artificial sensory feedback to stimulate the population of area 5 "PPV" neurons such that the position trajectory of the agonist muscle $i$ matches the position trajectory

obtained in the presence of the natural proprioception during the reaching task. We call this "Problem 1". Although we are not treating the visual feedback in this work, this position trajectory tracking problem can be considered equivalent to the BMI experiments where the visual feedback is used by the subject to make online corrections in the trajectory during the reaching task in the absence of proprioception. In the second problem, the goal is to match the average firing activity of the agonist population of "PPV" neurons to its natural firing activity by designing an optimal artificial feedback to stimulate the agonist population of the area 5 "PPV" neurons. We call this "Problem 2". This firing rate trajectory tracking problem can be considered equivalent to the BMI experiments where the primary somatosensory area (S1) neurons are stimulated artificially to restore the natural proprioception information. Figure 5.12 and Figure 5.13 show the design of a closed-loop BMI operation during the reaching task for problem 1 and 2 respectively.

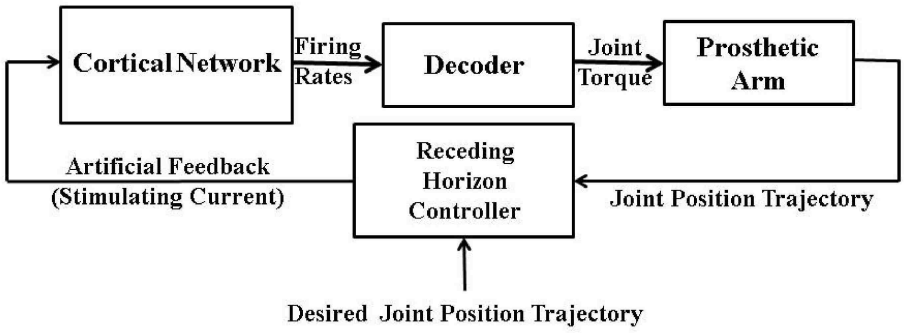



Figure 5.12: RHC based closed-loop BMI for Problem 1: Here the receding horizon controller designs the "Artificial Feedback" to stimulate "PPV" neurons such that the system output ("Single Joint Position" trajectory) mimics the "Desired Joint Position" trajectory.

As shown in Figure 5.12 and Figure 5.13, for both problems we use a model-based optimal receding horizon controller to design the optimal artificial sensory feedback.

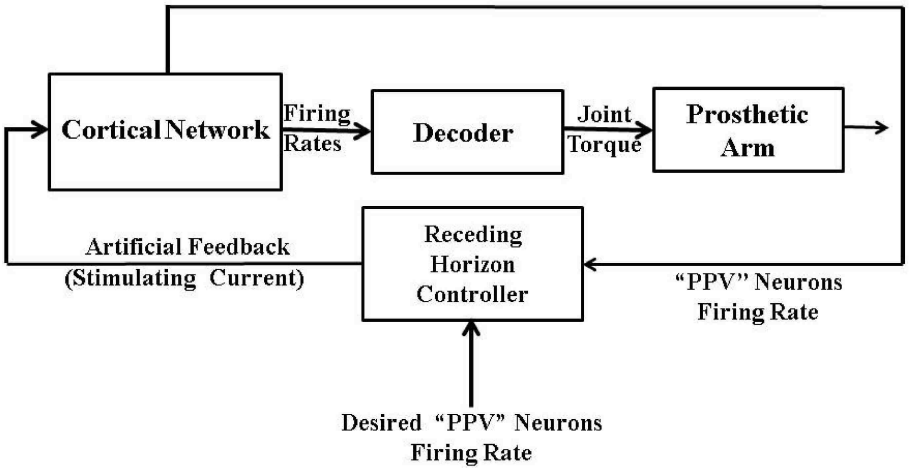



Figure 5.13: RHC based closed-loop BMI for Problem 2: Here the receding horizon controller designs the "Artificial Feedback" to stimulate "PPV" neurons such that the system output ("PPV Neurons Firing Rate" ) mimics the "Desired PPV Neurons Firing Rate".

Briefly, a model based RHC policy is an optimal control strategy that explicitly incorporates a dynamic model of the system as well as constraints in determining control actions. At each time $k \geq 1$, the system outputs are obtained and a model of the system is used to predict future outputs $O(k+m+1 \mid k)$, $m = 0, 1, 2, \cdots, N_p - 1$ as a function of current and future control moves $I(k+l \mid k)$, $l = 0, 1, 2, \cdots, N_c - 1$. How far ahead in the future the predictions are computed is called the prediction horizon $N_p$ and how far ahead the control moves are computed is called the control horizon $N_c$. Using these predictions, the $N_c$ control moves $I(k+l \mid k)$, $l = 0, 1, 2, \cdots, N_c - 1$ are optimally computed by minimizing a cost function $\mathbb{J}_p(k)$ over the prediction horizon $N_p$ subject to constraints on the control inputs as well as any other constraints on the internal states and outputs of the system. Only the first optimally computed move $I(k \mid k)$ is then used by the system to compute the outputs. At the next time $k+1$, new system measurements are obtained and the optimization problem is solved again

with the new measurements. Thus, the control and prediction horizon recede by one step as time moves ahead by one step.

For the systems shown in Figure 5.12 and Figure 5.13, we use the described model-based RHC policy and formulate the following control problem:

$$\min_{I(k|k),I(k+1|k),\cdots,I(k+N_c-1|k)} \mathbb{J}_p(k) \tag{5.18a}$$

such that

$$I(k+l \mid k) \in [-0.5, 0.5] \;\; for \;\; 0 \leq l \leq N_c - 1, \tag{5.18b}$$

$$I(k+l \mid k) = 0 \;\; for \;\; N_c \leq l \leq N_p - 1. \tag{5.18c}$$

Here, $I(k+l \mid k)$ for $l = 0, 1, \cdots, N_c - 1$ is the designed artificial sensory input. $\mathbb{J}_p(k) = \sum_{m=0}^{N_p-1}(O(k+m+1 \mid k) - R(k+m+1 \mid k))^2$ is the cost function. In case of Problem 1 (Figure 5.12), the measured output of the system $O(k \mid k)$ at a given time $k$ is the position of the agonist muscle $i$ i.e. $p_i(k \mid k)$. $R(\cdot)$ represents the desired position trajectory. In case of Problem 2 (Figure 5.13), the measured output of the system $O(k \mid k)$ at a given time $k$ is the average firing activity of the area 5 "PPV" neurons associated with the agonist muscle $i$ i.e. $x_i(k \mid k)$. $R(\cdot)$ represents the desired average firing activity of the area 5 "PPV" neurons.

To solve the control problem, first we compute the desired position and firing activity trajectory for Problem 1 and 2 respectively. For this, we simulate equations (5.1) - (5.5), equation (5.6a), equation (5.7), equation (5.10), equation (5.12), and equation (5.14). It should be noted that we have included only the natural sensory feedback (Ia) to the area 5 "PPV" neurons through equation 5.6a for computing

the desired trajectory. Thus in the absence of natural proprioception to the area 4 "IFV" and "SFV" neurons, $q_i(t) = f_i(t) = 0$ in equation (5.10). Next we compute $p_i(k+m+1 \mid k)$ and $x_i(k+m+1 \mid k)$ for $m = 0, 1, 2, \cdots, N_p - 1$ for Problem 1 and 2 respectively. For this, we use a model of the system given by equation (5.1) - (5.4), equation (5.10), equation (5.12), and equation (5.14) along with the following modified firing activity dynamics of the "PPV" neurons:

$$\begin{aligned} \frac{\mathrm{dx_i(t)}}{\mathrm{dt}} &= (1 - x_i(t)) \max\{\Theta y_i(t) - I(k+l \mid k), 0\} \\ &\quad - x_i(t) \max\{\Theta y_j(t) + I(k+l \mid k), 0\}. \end{aligned} \tag{5.19}$$

Here, $I(k+l \mid k)$ is constant during $t$ and $t + 10$ ms i.e. between the sample time. With this, we solve the optimization problem (equations (5.18a), (5.18b), and (5.18c)) numerically in MATLAB for both problems. We use the MATLAB optimization function "fmincon" with the sequential quadratic programming "sqp" algorithm. For both problems, we set $N_c = 5$ and $N_p = 30$. Figure 5.14 shows the performance of the controller in tracking the desired position trajectory of the agonist muscle $i$ for "Problem 1".

As shown in the top plot of Figure 5.14, the controller performs well in tracking the desired position trajectory. Also the stimulation of the area 5 "PPV" neurons by the designed optimal artificial sensory feedback recovers the closed-loop velocity trajectory, as shown in the bottom plot of Figure 5.14. Thus the designed optimal artificial sensory feedback is sufficient in recovering the closed-loop performance of the decoder in the absence of the natural proprioceptive feedback pathways. Next we wonder if the designed optimal artificial sensory feedback in "Problem 1" also recovers the natural average firing activity of the cortical neurons. Figure 5.15 shows

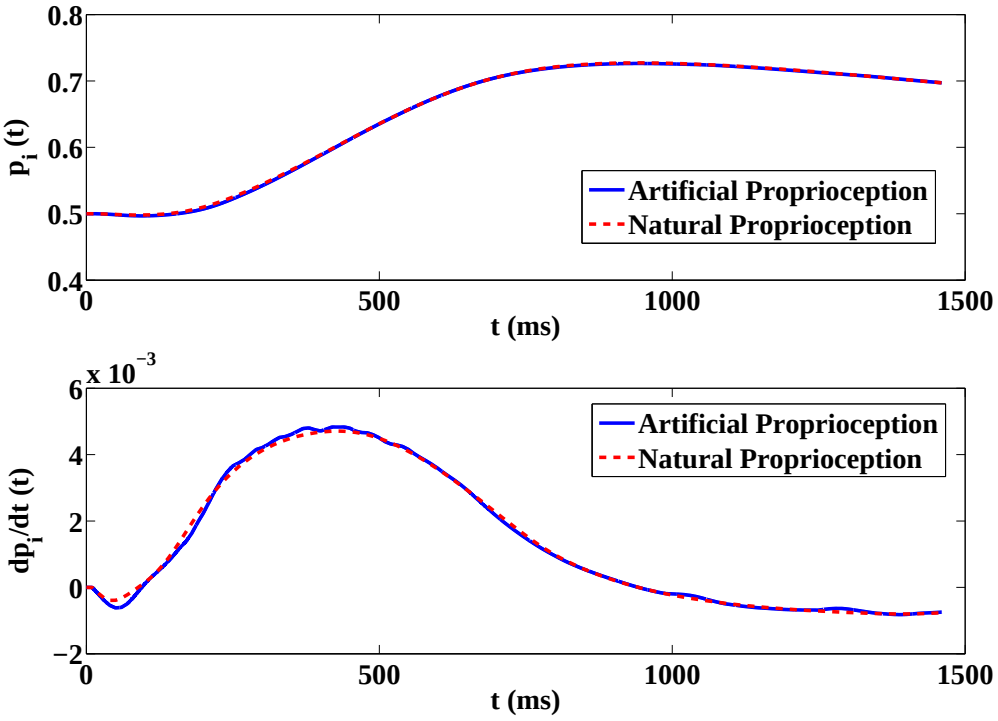



Figure 5.14: The position and velocity trajectory profile for "Problem 1" (Figure 5.12): The top plot shows the position trajectory of the agonist muscle $i$ in the presence of the designed artificial sensory feedback (solid line) and the natural sensory feedback (dotted line). The bottom plot shows the velocity trajectory of the agonist muscle $i$ in the presence of the designed artificial sensory feedback (solid line) and the natural sensory feedback (dotted line).

the average firing activity of the cortical area 4 and 5 neurons during the reaching task.

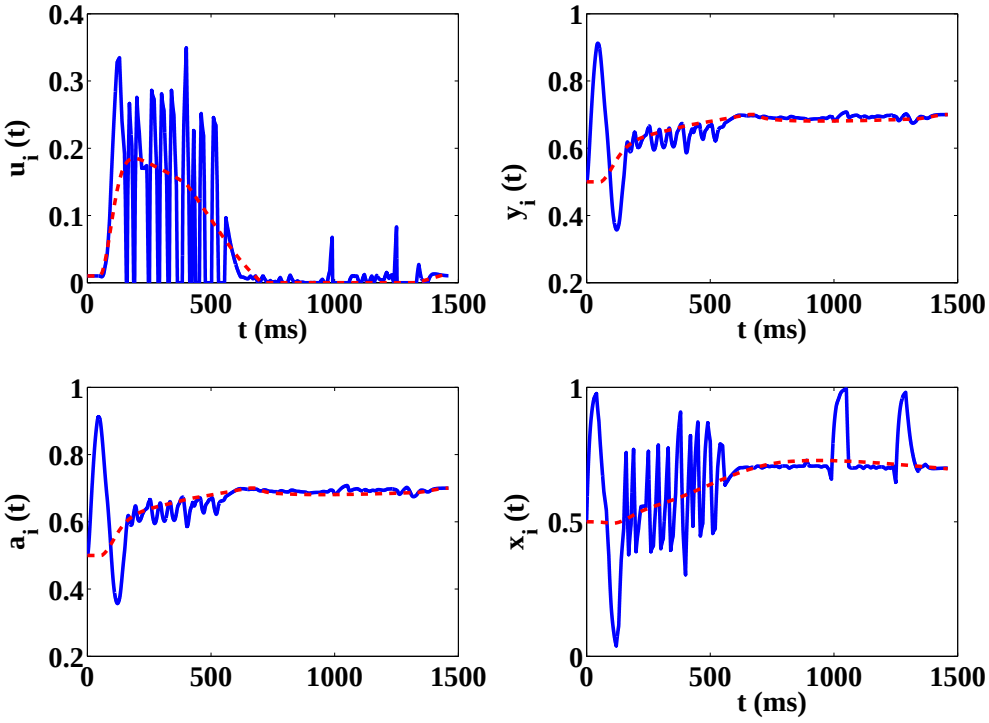



Figure 5.15: The average firing activity of a population of the cortical area 4 "DVV" neurons ($u_i(t)$), "OPV" neurons ($y_i(t)$), "OFPV" neurons ($a_i(t)$), and the cortical area 5 "PPV" neurons ($x_i(t)$) in the presence of the artificial sensory feedback (solid line) and the natural sensory feedback (dotted line) for "Problem 1" (Figure 5.12).

As shown in Figure 5.15, the average firing activity of the cortical area 4 and 5 neurons during the artificial stimulation differs significantly from the natural. This shows that although the artificial stimulation of the "PPV" neurons through the design of "Problem 1" recovers the closed-loop performance of the decoder, it fails to recover the natural firing activity of the cortical neurons.

Next we study the performance of the designed controller for "Problem 2" (Figure

5.13). Remember that the objective of the controller here is to track the natural average firing activity of the area 5 "PPV" neurons by designing optimal artificial sensory input to the "PPV" neurons. Figure 5.16 shows the average firing activity of the cortical area 4 and 5 neurons.

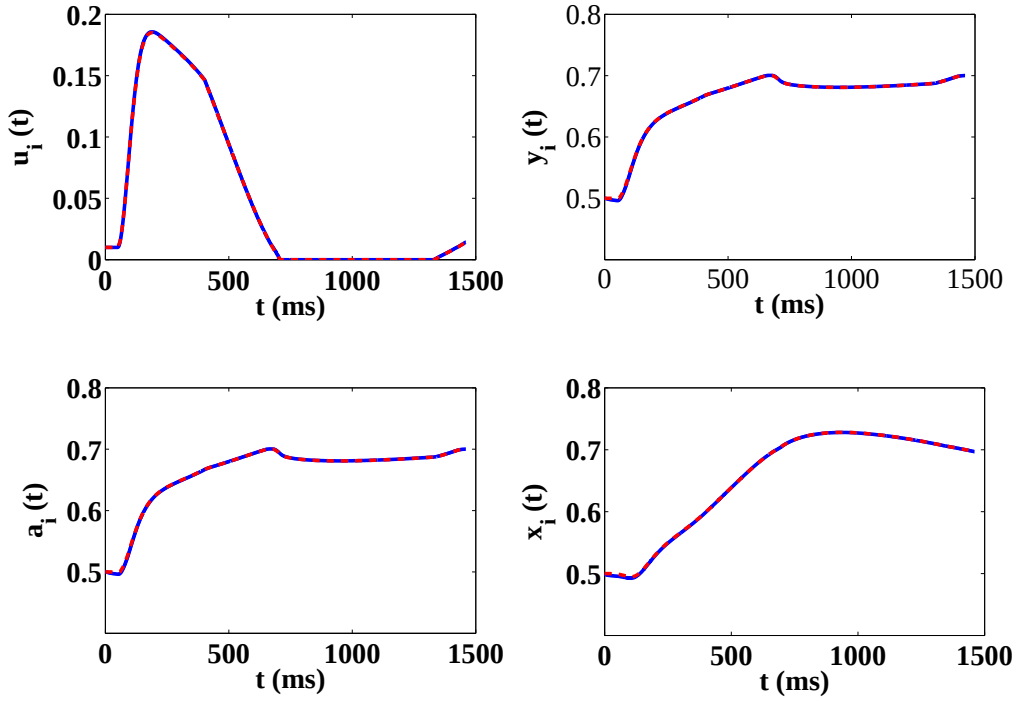



Figure 5.16: The average firing activity of the cortical area 4 "DVV" neurons ($u_i(t)$), "OPV" neurons ($y_i(t)$), "OFPV" neurons ($a_i(t)$), and the cortical area 5 "PPV" neurons ($x_i(t)$) in the presence of artificial sensory feedback (solid line) and the natural sensory feedback (dotted line) for "Problem 2" (Figure 5.13).

As shown in the bottom right plot of Figure 5.16, the designed controller performs well in tracking the natural firing activity of the area 5 "PPV" neurons. Moreover, the stimulation results in recovering the natural firing activity of the area 4 cortical neurons. Figure 5.17 shows the position and velocity trajectory of the agonist muscle

$i$ during the movement.

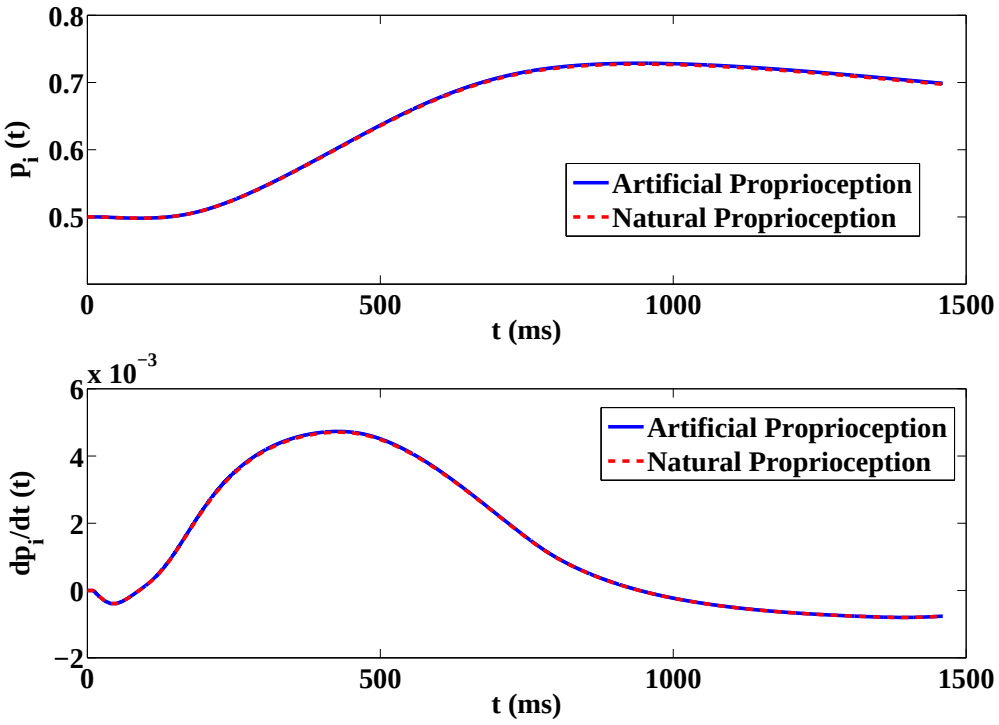



Figure 5.17: The Position and velocity trajectory for "Problem 2" (Figure 5.13): The top plot shows the position trajectory of the agonist muscle $i$ in the presence of artificial sensory feedback (solid line) and the natural sensory feedback (dotted line). The bottom plot shows the velocity trajectory of the agonist muscle $i$ in the presence of artificial sensory feedback (solid line) and the natural sensory feedback (dotted line).

As shown in Figure 5.17, the decoder recovers the closed-loop (natural) performance in this case. This shows that the designed controller in "Problem 2" not only recovers the closed-loop performance of the decoder but also recovers the natural firing activity of the cortical neurons through the optimal artificial stimulation.

## 5.4 Concluding Remarks

Brain-machine interfaces (BMIs) offer applications of control theory from their design to the real implementation. In this chapter, we have designed optimal artificial sensory feedback in a control-theoretic framework to recover the closed-loop performance of a BMI during voluntary single joint extension task. This is the first systematic attempt to incorporate artificial proprioception in BMIs towards stimulation enhanced next generation BMIs. A psycho-physiological cortical circuit model for the voluntary control of a single joint movement has been used to design the BMI. A need for designing an artificial sensory feedback has been shown by analyzing the performance of the BMI in the presence and the absence of the natural proprioceptive feedback pathways. Two control problems namely, the position trajectory tracking problem and the cortical sensory neurons average firing rate tracking problem, have been investigated towards designing an optimal artificial sensory feedback for the BMI in the receding horizon control framework. From our results, we conclude that tracking the natural firing activity of the cortical sensory neurons using an external stimulating controller is the appropriate approach towards recovering the natural performance of the motor task. Although we have excluded the treatment of visual feedback as well any form of cortical learning throughout our analysis, the designed framework allows incorporation of both the visual feedback and the cortical learning during a reaching task.

# Chapter 6

# Optimal Regulation of Multi-Agent Systems: A Centralized Approach

## 6.1 Introduction

In this chapter, we propose an optimal centralized control strategy called "Broadcast Stochastic Receding Horizon Control (BSRHC)" for the dynamical stabilization of a class of multi-agent systems such as a swarm of identical agents. The proposed control strategy is a generalization of previous broadcast feedback strategy reported in literature [162, 121, 171, 85, 25] and allows incorporation of realistic physical constraints of the dynamical system in the control problem formulation. The central idea of the proposed strategy is to design and broadcast the optimal control inputs in a predictive framework to all the agents in a swarm using the aggregate behavior of agents as the only available feedback information. Our novelty here is in integrating the broadcast concept with existing probabilistic tools and the theory of finite receding horizon based optimal control policy.

Using the developed strategy in this chapter, we theoretically demonstrate the stabilization of a swarm of stochastically behaving agents to the desired state. Each agent in the system is capable of making random transitions from one state to another among a finite number of available states. The desire to regulate a swarm of agents simultaneously using a centralized controller as well as the presence of a nonlinear constraint on control inputs make the strategy suitable for the particular problem. The dynamical behavior of individual agents is represented by the discrete time finite state Markov chain model. Probabilistic tools such as the supermartingale theory and the bounded convergence theorem are applied to guarantee the almost sure convergence of the closed-loop system behavior to the desired one. The derived stability and convergence results establish key principles applicable to stabilize general stochastic dynamical systems.

## 6.2 Mathematical Tools

In this chapter, we make use of the following known results from probability theory in deriving our system model as well as results:

1. **Bernoulli Random Variable:** Let $X$ be a Bernoulli random variable with

$$X = \begin{cases} 1 & \text{w.p.} \quad p, \\ 0 & \text{w.p.} \quad 1-p. \end{cases} \tag{6.1}$$

   Here "w.p." is the abbreviation of "with probability". The expected value and variance of $X$ are given by $\mathbb{E}X = p$ and $Var(X) = p(1-p)$.

2. **Conditional Expectation Results [41]:** Let $X$ be a random variable. $\mathcal{F}_1$ and $\mathcal{F}_2$ are two $\sigma$-fields. If $\mathcal{F}_1 \subset \mathcal{F}_2$ then

   (a) $\mathbb{E}[\mathbb{E}[X \mid \mathcal{F}_1] \mid \mathcal{F}_2] = \mathbb{E}[X \mid \mathcal{F}_1]$,

   (b) $\mathbb{E}[\mathbb{E}[X \mid \mathcal{F}_2] \mid \mathcal{F}_1] = \mathbb{E}[X \mid \mathcal{F}_1]$.

3. **Supermartingale [41]:** Let $\mathcal{F}_n$ be a filtration, i.e. an increasing sequence of $\sigma$-fields. A sequence $X_n$ is said to be adapted to $\mathcal{F}_n$ if $X_n \in \mathcal{F}_n$ for all $n$. If $X_n$ is a sequence with

   (a) $\mathbb{E}|X_n| < \infty$,

   (b) $X_n$ is adapted to $\mathcal{F}_n$,

   (c) $\mathbb{E}[X_{n+1} \mid \mathcal{F}_n] \leq X_n$ for all $n$,

   then $X_n$ is said to be a supermartingale with respect to $\mathcal{F}_n$.

4. **Supermartingale Results [41]:** If $X_n$ is a supermartingale then for $n > m$, $\mathbb{E}[X_n \mid \mathcal{F}_m] \leq X_m$.

5. **Supermartingale Convergence Theorem [41]:**

   **Theorem 6.2.1.** *[41] If $X_n$ is a supermartingale then for $n > m$, $\mathbb{E}[X_n \mid \mathcal{F}_m] \leq X_m$.*

6. **Bounded Convergence Theorem [41]:**

   **Theorem 6.2.2.** *If $X_n \to X$ a.s., $|X_n| \leq M$ for all $n$ where $M < \infty$ and is a constant, then $\mathbb{E}[X_n] \to \mathbb{E}[X]$.*

## 6.3 Stochastic Two-State Multi-Agent System

### 6.3.1 System Model

We define a stochastic two-state multi-agent system as an ensemble of identical agents, where each agent assumes a state value stochastically out of two possible states. Each state has an associated value which is defined as

$$X_{i,k} = \begin{cases} 1 & \text{when "ON"}, \\ 0 & \text{when "OFF"}. \end{cases} \tag{6.2}$$

Here, $X_{i,k}$ is the state of the $i^{th}$ agent at time $k$. We assume that all agents in this system behave independently. At time $k$, an agent at the state 'ON' can make transition to the state 'OFF' with a transition probability $p_{k|k}$. Similarly, an agent at the state 'OFF' can make transition to the state 'ON' with a transition probability $q_{k|k}$. Figure 6.1 illustrates the state transition behavior of agents in a two-state multi-agent system.

We write the conditional expectation of the state of the $i^{th}$ agent at time $k+1$ given the information till time $k$ as

$$\mathbb{E}[X_{i,k+1} \mid X_{i,k}] = X_{i,k}(1 - p_{k|k}) + (1 - X_{i,k})q_{k|k}. \tag{6.3}$$

Since we are interested in the aggregate system behavior, we define an ensemble behavior of the system at time $k$ as

$$N_k = \sum_{i=1}^{N} X_{i,k}, \tag{6.4}$$

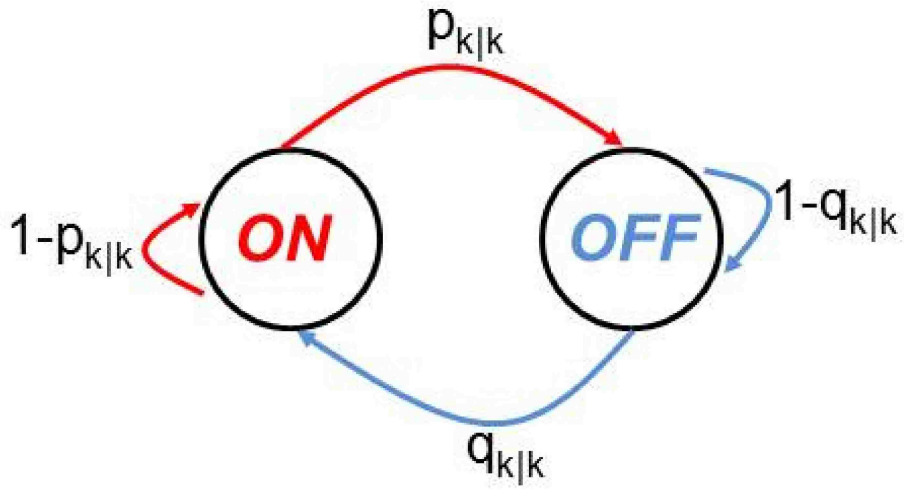



Figure 6.1: State transition behavior of agents in a two-state multi-agent system.

Here, $N_k$ is the number of agents with 'ON' state at time $k$. $N$ is the predefined number of agents present in the system. Now by applying the summation $\sum_{i=1}^{N}$ on both sides of equation (6.3) and using equation (6.4), we derive the conditional expectation of the number of agents at the 'ON' state at time $k+1$ given the information till time $k$ as

$$\mathbb{E}[N_{k+1} \mid N_k] = N_k(1 - p_{k|k}) + (N - N_k)q_{k|k}. \tag{6.5}$$

Equation (6.5) states that the conditional expectation of number of agents at the state 'ON' at time $k+1$ is equal to the sum of the expected number of agents at the state 'ON' at time $k$ which stays at the state 'ON' at time $k+1$ and the expected number of agents at the state 'OFF' at time $k$ which makes transition to the state 'ON' at time $k+1$. We define the system error at time $k+1$ as $e_{k+1} = N_r - N_{k+1}$, where $N_r$ is the time invariant desired number of agents at the state 'ON'. Now by replacing $N_{k+1}$ with $N_r - e_{k+1}$ in equation (6.5), we express the conditional expectation of the system error at time $k+1$ as

$$\begin{aligned}\mathbb{E}[e_{k+1} \mid \mathcal{F}_k] &= e_k + (N_r - e_k)p_{k|k} - (N - N_r + e_k)q_{k|k} \\ &= (1 - p_{k|k} - q_{k|k})e_k + N_r(p_{k|k} + q_{k|k}) - Nq_{k|k} \\ &= A_{k|k}e_k + F_{k|k}.\end{aligned} \tag{6.6}$$

Here, $\mathcal{F}_k$ is a $\sigma$-algebra generated by the set $\{e_0, e_1, \cdots, e_k\}$ [41]. Now using equation (6.6) and a conditional expectation property $\mathbb{E}[e_{k+j} \mid \mathcal{F}_k] = \mathbb{E}[\mathbb{E}[e_{k+j} \mid \mathcal{F}_{k+j-1}] \mid \mathcal{F}_k]$ for $j \geq 1$, we derive a predictive expression for the conditional expectation $\mathbb{E}[e_{k+2} \mid \mathcal{F}_k]$ as

$$\begin{aligned}\mathbb{E}[e_{k+2} \mid \mathcal{F}_k] &= \mathbb{E}[\mathbb{E}[e_{k+2} \mid \mathcal{F}_{k+1}] \mid \mathcal{F}_k] \\ &= \mathbb{E}[A_{k+1|k+1}e_{k+1} + F_{k+1|k+1} \mid \mathcal{F}_k] \\ &= A_{k+1|k}\mathbb{E}[e_{k+1} \mid \mathcal{F}_k] + F_{k+1|k} \\ &= A_{k+1|k}A_{k|k}e_k + A_{k+1|k}F_{k|k} + F_{k+1|k} \\ &= (\textstyle\prod_{m=0}^{1} A_{k+m|k})e_k + \sum_{m=0}^{1} F_{k+m|k}(\prod_{n=m+1}^{1} A_{k+n|k}).\end{aligned} \tag{6.7}$$

In general, we can write a predictive expression for the conditional expectation $\mathbb{E}[e_{k+j} \mid \mathcal{F}_k]$ for $j \geq 1$ as

$$\mathbb{E}[e_{k+j} \mid \mathcal{F}_k] = (\prod_{m=0}^{j-1} A_{k+m|k})e_k + \sum_{m=0}^{j-1} F_{k+m|k}(\prod_{n=m+1}^{j-1} A_{k+n|k}). \tag{6.8}$$

Here, $A_{k+m|k} = 1 - p_{k+m|k} - q_{k+m|k}$ and $F_{k+m|k} = N_r(1 - A_{k+m|k}) - Nq_{k+m|k}$ for $m \geq 0$. Next, we compute the conditional variance of the system error at time $k+1$ i.e. $Var[e_{k+1} \mid \mathcal{F}_k]$. Using the standard variance formula for a Bernoulli random variable, we first compute the conditional variance of the $i^{th}$ agent state at time $k+1$ as

$$Var[X_{i,k+1} \mid X_{i,k}] = X_{i,k}p_{k|k}(1 - p_{k|k}) + (1 - X_{i,k})q_{k|k}(1 - q_{k|k}). \tag{6.9}$$

Using $Var[\sum_{i=1}^{N} X_{i,k+1} \mid X_{i,k}, i = 1, 2, \cdots N] = \sum_{i=1}^{N} Var[X_{i,k+1} \mid X_{i,k}, i = 1, 2, \cdots N]$, $e_{k+1} = N_r - N_{k+1}$ and equation (6.9), we write the conditional variance of the system error at time $k+1$ as

$$\begin{aligned} Var[e_{k+1} \mid \mathcal{F}_k] &= (N_r - e_k)p_{k|k}(1 - p_{k|k}) + (N - N_r + e_k)q_{k|k}(1 - q_{k|k}) \\ &= C_{k|k}e_k + G_{k|k}. \end{aligned} \tag{6.10}$$

Here, $C_{k|k} = q_{k|k}(1 - q_{k|k}) - p_{k|k}(1 - p_{k|k})$ and $G_{k|k} = Nq_{k|k}(1 - q_{k|k}) - N_r C_{k|k}$. Since $\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] = Var[e_{k+1} \mid \mathcal{F}_k] + (\mathbb{E}[e_{k+1} \mid \mathcal{F}_k])^2$, we can write the conditional expectation $\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k]$ as

$$\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] = C_{k|k}e_k + G_{k|k} + (A_{k|k}e_k + F_k)^2. \tag{6.11}$$

Now using conditional expectation properties $\mathbb{E}[e_{k+j}^2 \mid \mathcal{F}_{k+j-1}] = Var(e_{k+j} \mid \mathcal{F}_{k+j-1}) + (\mathbb{E}[e_{k+j} \mid \mathcal{F}_{k+j-1}])^2$ and $\mathbb{E}[e_{k+j}^2 \mid \mathcal{F}_k] = \mathbb{E}[\mathbb{E}[e_{k+j}^2 \mid \mathcal{F}_{k+j-1}] \mid \mathcal{F}_k]$ along with equations (6.8) and (6.10), we derive an expression for the conditional expectation of $e_k^2$ at a future time $k+j$ with $j \geq 1$ as

$$\begin{aligned} \mathbb{E}[e_{k+j}^2 \mid \mathcal{F}_k] = &(\prod_{m=0}^{j-1} A_{k+m|k}^2)e_k^2 + \sum_{m=0}^{j-1}((2F_{k+m|k}A_{k+m|k} + C_{k+m|k}) \\ &(\prod_{n=0}^{m-1} A_{k+n|k})(\prod_{n=m+1}^{p-1} A_{k+n|k}^2))e_k \\ &+ \sum_{m=0}^{j-1}((2F_{k+m|k}A_{k+m|k} + C_{k+m|k})(\sum_{n=0}^{m-1} F_{k+n|k} \\ &(\prod_{l=n+1}^{m-1} A_{k+l|k})) + F_{k+m|k}^2 + G_{k+m|k})(\prod_{n=m+1}^{j-1} A_{k+n|k}^2). \end{aligned} \tag{6.12}$$

Here, $C_{k+m|k} = q_{k+m|k}(1 - q_{k+m|k}) - p_{k+m|k}(1 - p_{k+m|k})$, $G_{k+m|k} = -N_r C_{k+m|k} + Nq_{k+m|k}(1 - q_{k+m|k})$ for $m \geq 0$.

### 6.3.2 Problem Formulation

In the previous section, we derived conditional expectation based predictive dynamical models to represent the evolution of the aggregate system behavior in future time given the information till the present time. In this section, we use these models to formulate an optimal control problem in the broadcast stochastic receding horizon control (BSRHC) framework, as shown in Figure 6.2, to stabilize the aggregate system behavior.

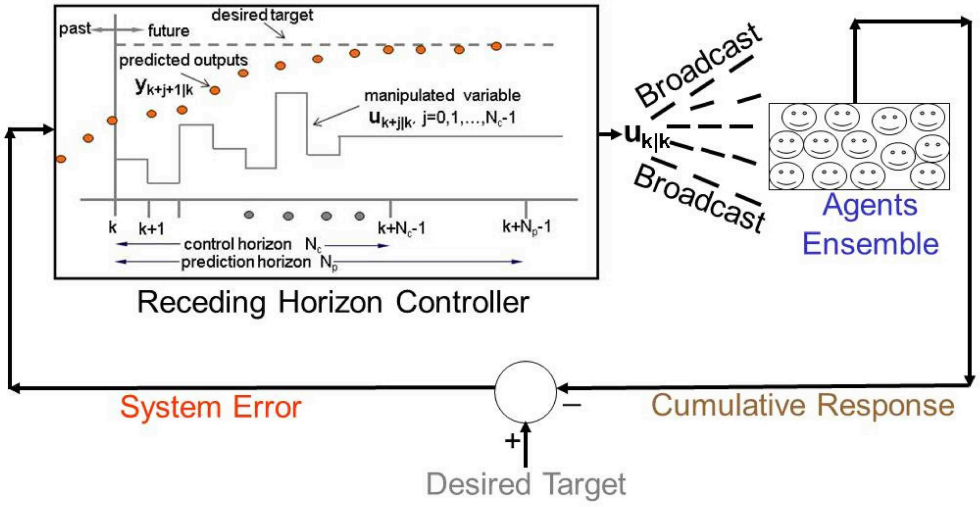



Figure 6.2: Broadcast RHC: Here at the current time step $k$, the "Receding Horizon Controller" designs control inputs, "$u_{k|k}$" and "Broadcast" them to the system ("Agents Ensemble"). Each agent in the system makes independent decision and contributes to the "Cumulative Response" of the system. The "Cumulative Response" of the system is then compared with the "Desired Target" and the "System Error" is fed back to the controller for designing new control inputs at time step $k + 1$.

In the design of the broadcast stochastic receding horizon control policy for the two-state multi-agent system, the objective is to minimize the system error by computing optimal transition probabilities. To achieve this objective, we formulate a constrained non-linear stochastic finite receding horizon control (RHC) problem as follows:

$$\min_{\substack{p_{k|k},\cdots,p_{k+N_c-1|k},\\ q_{k|k},\cdots,q_{k+N_c-1|k}}} \mathbb{J}_k \tag{6.13}$$

s.t.

$$(p_{k+m|k}, q_{k+m|k}) \in [0,1] \times [0,1] \;\; for \;\; 0 \leq m \leq N_c - 1, \tag{6.14a}$$

$$p_{k+m|k} = q_{k+m|k} = 0 \;\; for \;\; m \geq N_c, \tag{6.14b}$$

$$\mathbb{E}[e^2_{k+m+1} \mid \mathcal{F}_k] < e^2_k \;\; for \;\; e_k \neq 0, \; 0 \leq m \leq N_c - 1, \tag{6.14c}$$

$$\mathbb{E}[e^2_{k+m+1} \mid \mathcal{F}_k] = e^2_k \;\; for \;\; e_k = 0, \; 0 \leq m \leq N_c - 1. \tag{6.14d}$$

The cost function $\mathbb{J}_k$ at time $k$ is defined as

$$\mathbb{J}_k = \sum_{m=0}^{N_p-1} \mathbb{E}[e^2_{k+m+1} \mid \mathcal{F}_k] + \sum_{n=0}^{N_c-1} (p^2_{k+n|k} + q^2_{k+n|k}). \tag{6.15}$$

$N_p$ and $N_c$ are time invariant prediction and control horizon respectively. The optimization problem defined by equations (6.13), (6.14) and (6.15) is solved at each time

$k$ and the computed optimal transition probabilities are updated using the following steps:

**step 1:** Given $e_k$ at time $k \geq 0$, express equations (6.14c), (6.14d) and (6.15) in terms of $e_k$, $p_{k+m|k}$ and $q_{k+m|k}$ for $m = 0, 1, \cdots, N_c - 1$ using (6.12).

**step 2:** Solve equations (6.13), (6.14) and (6.15) and compute optimal transition probabilities $p_{k+m|k}$ and $q_{k+m|k}$ for $m = 0, 1, \cdots, N_c - 1$.

**step 3:** Broadcast $p_{k|k}$ and $q_{k|k}$ to the system and measure the aggregate system behavior at time $k + 1$ i.e. $e_{k+1}$.

**step 4:** Using $e_{k+1}$ as a feedback, repeat step 1, 2 and 3 for time $k + 1$ by receding the prediction horizon by one time step.

**Remark 6.3.1.** *Replacing equation (6.14b) in the broadcast finite RHC problem defined by equations (6.13),(6.14), and (6.15) with*

$$p_{k+m|k} = p_{k+N_c-1|k} \ \ for \ \ m \geq N_c, \tag{6.16a}$$

$$q_{k+m|k} = q_{k+N_c-1|k} \ \ for \ \ m \geq N_c, \tag{6.16b}$$

*provides an alternative design for regulating behaviors of agents in the multi-agent system. It should be noted that stability and convergence theorems, presented in the next section, are equally applicable to both designs.*

Next, we show that the control problem defined by equations (6.13), (6.14) and (6.15) is feasible. This feasibility is sufficient to conclude the stability and convergence of the constrained non-linear finite RHC control problem, as stated in Theorem 6.3.4.

### 6.3.3 Stability and Convergence

**Claim 6.3.2.** *For all $k \geq 0$, there exists a pair of transition probabilities $(p_{k|k}, q_{k|k}) \in [0,1]\times[0,1]$ for the two-state multi-agent system such that $\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] < e_k^2$ whenever $e_k \neq 0$. Moreover, if $(p_{k|k}, q_{k|k}) = (0,0)$ then $\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] = e_k^2$.*

*Proof.* We define a function $f(p_{k|k}, q_{k|k})$ as

$$f(p_{k|k}, q_{k|k}) = \mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] - e_k^2. \tag{6.17}$$

Using equation (6.12) for $j = 1$, we express $f(p_{k|k}, q_{k|k})$ as

$$\begin{aligned} f(p_{k|k}, q_{k|k}) =& N_r^2 + [(N_r - e_k)p_{k|k}(1-p_{k|k}) + (N - N_r + e_k)q_{k|k}(1-q_{k|k})] \\ &+ [(N_r - e_k)(1-p_{k|k}) + (N - N_r + e_k)q_{k|k}]^2 \\ &- 2N_r[(N_r - e_k)(1-p_{k|k}) + (N - N_r + e_k)q_{k|k}] - e_k^2. \end{aligned} \tag{6.18}$$

Clearly, $f(0,0) = 0$. Thus to show $\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] < e_k^2$ for some $(p_{k|k}, q_{k|k}) \in [0,1]\times[0,1]$, it is sufficient to show that $f(p_{k|k}, q_{k|k})$ is a decreasing function of $p_{k|k}$ and $q_{k|k}$ on $(p_{k|k}, q_{k|k}) \in [0,x]\times[0,y]$, where $(x,y) \in [0,1]\times[0,1]$. For this, we consider two cases based on the sign of $e_k$ at a given time step $k$. As a first case, we consider $e_k < 0$ and fix $q_{k|k}$ in equation (6.18). We compute partial derivative of $f(p_{k|k}, q_{k|k})$ w.r.t. $p_{k|k}$. We then impose $\frac{\partial f(p_{k|k}, q_{k|k})}{\partial p_{k|k}} < 0$ and $p_{k|k} > 0$ on the computed partial derivative. This results in

$$p_{k|k} < \frac{N - N_r + e_k}{N_r - e_k - 1} q_{k|k} - \frac{1 + 2e_k}{2(N_r - e_k - 1)}, \tag{6.19a}$$

$$q_{k|k} > \frac{1 + 2e_k}{2(N - N_r + e_k)}. \tag{6.19b}$$

Since $e_k < 0$, equation (6.19) implies that $q_{k|k} \in [0,y]$ with $y = 1$ and $p_{k|k} \in (0,x)$

with $x = \frac{N-N_r+e_k}{N_r-e_k-1}q_{k|k} + \frac{e_k+0.5}{N_r-e_k-1}$ for all $e_k < 0$. Next, we consider $e_k > 0$ and fix $p_{k|k}$ in equation (6.18). Now we compute partial derivative of $f(p_{k|k}, q_{k|k})$ w.r.t. $q_{k|k}$. We then impose $\frac{\partial f(p_{k|k},q_{k|k})}{\partial q_{k|k}} < 0$ and $q_{k|k} > 0$ on the computed partial derivative. This results in

$$q_{k|k} < \frac{N_r - e_k}{N - N_r + e_k - 1}p_{k|k} + \frac{e_k - 0.5}{N - N_r + e_k - 1}, \tag{6.20a}$$

$$p_{k|k} > \frac{0.5 - e_k}{N_r - e_k}. \tag{6.20b}$$

Since $e_k > 0$, equation (6.20) implies that $p_{k|k} \in [0, x]$ with $x = 1$ and $q_{k|k} \in (0, y)$ with $y = \frac{N_r-e_k}{N-N_r+e_k-1}p_{k|k} + \frac{e_k-0.5}{N-N_r+e_k-1}$ for all $e_k > 0$. Thus for all $k \geq 0$, there exists a pair of transition probabilities $(p_{k|k}, q_{k|k}) \in [0, 1] \times [0, 1]$ such that $\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] < e_k^2$ whenever $e_k \neq 0$. □

**Claim 6.3.3.** *The broadcast finite stochastic RHC problem defined by equations (6.13), (6.14) and (6.15) is feasible for the stochastic two-state multi-agent system.*

*Proof.* To prove Claim 6.3.3, it is sufficient to show the existence of a pair of transition probabilities $(p_{k+m|k}, q_{k+m|k})$ for $0 \leq m \leq N_c - 1$ which satisfies equations (6.14c) and (6.14d). We first analyze the feasibility of equation (6.14d). We observe that if $(p_{k+m|k}, q_{k+m|k}) = (0, 0)$ for $0 \leq m \leq N_c - 1$ and $k \geq 0$ in equation (6.12) with $j = m + 1$, equation (6.12) reduces to $\mathbb{E}[e_{k+m+1}^2 \mid \mathcal{F}_k] = e_k^2$. Thus, equation (6.14d) is feasible. Next we consider equation (6.14c). From Claim 6.3.2, we know that equation (6.14c) is feasible for $m = 0$. Now we write $\mathbb{E}[e_{k+m+1}^2 \mid \mathcal{F}_k]$ for $m = 1$ as $\mathbb{E}[e_{k+2}^2 \mid \mathcal{F}_k] = \mathbb{E}[\mathbb{E}[e_{k+2}^2 \mid \mathcal{F}_{k+1}] \mid \mathcal{F}_k]$. We know from Claim 6.3.2 that there exists $(p_{k+1|k+1}, q_{k+1|k+1}) \in [0, 1] \times [0, 1]$ such that $\mathbb{E}[e_{k+2}^2 \mid \mathcal{F}_{k+1}] < e_{k+1}^2$. Thus for this pair of transition probability, $\mathbb{E}[\mathbb{E}[e_{k+2}^2 \mid \mathcal{F}_{k+1}] \mid \mathcal{F}_k] < \mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k]$ from the monotonicity property [41] of conditional expectations. Now by choosing

$(p_{k+1|k}, q_{k+1|k}) = (p_{k+1|k+1}, q_{k+1|k+1})$ and using the results for $m = 0$, one can guarantee the existence of $(p_{k+1|k}, q_{k+1|k}) \in [0,1] \times [0,1]$ satisfying $\mathbb{E}[e_{k+2}^2 \mid \mathcal{F}_k] < e_k^2$. The existence of a pair of transition probabilities $(p_{k+m|k}, q_{k+m|k}) \in [0,1] \times [0,1]$ for $m \geq 2$ follows directly from mathematical induction. □

**Theorem 6.3.4.** *If there exists a pair of transition probabilities $(p_{k+m|k}, q_{k+m|k})$ for all $k \geq 0$ and $0 \leq m \leq N_c - 1$ such that the optimization problem defined by equations (6.13), (6.14) and (6.15) is feasible then*

$$e_k \xrightarrow{a.s.} 0.$$

*Here, a.s. means $Pr(\lim_{k\to\infty} e_k = 0) = 1$. Further,*

$$\mathbb{J}_k \xrightarrow{a.s.} 0.$$

*Proof.* To show the claim in the Theorem 6.3.4 for the two-state multi-agent system, we consider following sequential steps:

**step 1:** Results of Claim 6.3.3 guarantee the existence of $(p_{k+m|m}, q_{k+m|m}) \in [0,1] \times [0,1]$ for all $k \geq 0$ and $0 \leq m \leq N_c - 1$ such that the optimization problem defined by equations (6.13), (6.14) and (6.15) is feasible. Moreover under these transition probabilities, $e_k^2$ is a supermartingale.

**step 2:** By the supermartingale convergence theorem [41]

$$e_k^2 \xrightarrow{a.s.} X$$

and

$$\mathbb{E}X \leq \mathbb{E}e_0^2,$$

where $X$ is a limit random variable.

**step 3:** Since $\mathbb{E}e_k^2 < \infty$ for all $k \geq 0$, $|e_k^2| \leq \max\{(N - N_r)^2, N_r^2\} =$ constant for the two-state multi-agent system and $e_k^2 \xrightarrow{a.s.} X$ , by the bounded convergence theorem [41]

$$\mathbb{E}e_k^2 \overset{k\to\infty}{\longrightarrow} \mathbb{E}X \ .$$

**step 4:** Since $\mathbb{E}e_{k+1}^2 < \mathbb{E}e_k^2$ for all $e_k \neq 0$ and $k \geq 0$, $\mathbb{E}e_k^2 \downarrow 0$. Thus, $\mathbb{E}X = 0$.

**step 5:** From Chebyshev inequality, $Pr(|e_k| > \epsilon) \leq \mathbb{E}e_k^2/\epsilon^2$ for all $\epsilon > 0$. Since $\mathbb{E}e_k^2 \overset{k\to\infty}{\longrightarrow} 0$ , $\lim_{k\to\infty} Pr(|e_k| > \epsilon) \longrightarrow 0$. Thus, $e_k$ converges to 0 in probability.

**step 6:** Since $e_k$ converges to 0 in probability implies that $f(e_k)$ converges to $f(0)$ for any continuous function $f$, $e_k^2$ converges to 0 in probability. From step 2, we know that $e_k^2$ converges to $X$ a.s. This means that $e_k^2$ also converges to $X$ in probability. Therefore, $X$ must be 0.

**step 7:** Since $e_k^2 \xrightarrow{a.s.} 0$ , $Pr(|e_k| > \epsilon \ i.o.) = 0$ for all $\epsilon > 0$ (here $i.o.$ stands for infinitely often). Thus, $e_k \xrightarrow{a.s.} 0$ .

**step 8:** Since the optimization problem defined by equations (6.13), (6.14) and (6.15) is feasible and $e_k \xrightarrow{a.s.} 0$, $\mathbb{E}[e_{k+m+1}^2 \mid \mathcal{F}_k] \xrightarrow{a.s.} 0$ for $m = 0, 1, \cdots, N_c - 1$ (see equation (6.14d)). Therefore, $\mathbb{E}[e_{k+m+1} \mid \mathcal{F}_k] \xrightarrow{a.s.} 0$ $a.s.$ and $Var[e_{k+m+1} \mid \mathcal{F}_k] \xrightarrow{a.s.} 0$ for $m = 0, 1, \cdots, N_c - 1$. By using these results in equations (6.6) and

(6.10), it is easy to verify that $(p_{k|k}, q_{k|k}) \overset{k\to\infty}{\longrightarrow} (0,0)$ if $N_r \neq N$ and $N_r \neq N/2$. If $N_r = N/2$ then $(p_{k|k}, q_{k|k})$ can be $(0,0)$ or $(1,1)$ as $k \to \infty$ . If $N_r = N$ then $p_{k|k} = 0$ and $q_{k|k} \in [0,1]$ as $k \to \infty$. Since the designed controller also minimizes the total input cost, the optimality of inputs will lead to $(p_{k|k}, q_{k|k}) = (0,0)$ as $k \to \infty$. Now by using mathematical induction along with equations (6.8) and (6.12), it is not difficult to verify that $(p_{k+m|k}, q_{k+m|k}) \overset{k\to\infty}{\longrightarrow} (0,0)$ for $m = 1, 2, \cdots, N_c - 1$.

**step 9:** Since $\mathbb{J}_k = \sum_{m=0}^{N_c-1} \mathbb{E}[e_{k+m+1}^2 \mid \mathcal{F}_k] + \sum_{m=0}^{N_c-1}(p_{k+m|k}^2 + q_{k+m|k}^2) + \sum_{m=N_c}^{N_p-1} \mathbb{E}[e_{k+N_c}^2 \mid \mathcal{F}_k]$, an upper and lower bound on $\mathbb{J}_k$ is computed using $\mathbb{E}[e_{k+m+1}^2 \mid \mathcal{F}_k] \leq e_k^2 \ \ for \ \ m = 0, 1, \cdots, N_p - 1$ as

$$0 \leq \mathbb{J}_k \leq (N_p)e_k^2 + \sum_{n=0}^{N_c-1}(p_{k+m|k}^2 + q_{k+m|k}^2).$$

**step 10:** Since $e_k^2 \overset{a.s.}{\longrightarrow} 0$ and $(p_{k+m|k}, q_{k+m|k}) \overset{k\to\infty}{\longrightarrow} (0,0), \implies \mathbb{J}_k \overset{a.s.}{\longrightarrow} 0.$

□

**Remark 6.3.5.** ***An alternate proof of Theorem 6.3.4:*** *Since the optimization problem defined by equations (6.13), (6.14) and (6.15) is feasible and $Pr(e_k = 0) > 0$ for all $k > 0$, $e_k$ will hit zero in some finite time (say $k = k_1 > 0$) under the designed transition probabilities. At $k = k_1$, (6.14d) ensures that $\mathbb{E}[e_{k+m+1}^2 \mid \mathcal{F}_k] = 0$ for $m = 0, 1, \cdots, N_c - 1$. Thus the cost function $\mathbb{J}_k$ reduces to $\mathbb{J}_k = \sum_{m=0}^{N_c-1}(p_{k+m|k}^2 + q_{k+m|k}^2)$ at $k = k_1$. Since the minimum value of the cost function is zero when $p_{k+m|k} = q_{k+m|k} = 0$ for $m = 0, 1, \cdots, N_c - 1$, $\mathbb{J}_k$ also converges to zero.*

### 6.3.4 Analytical and Numerical Results

In this section, we provide the solution of the control problem defined by equations (6.13), (6.14) and (6.15) for the two-state multi-agent system. For the control horizon $N_c = 1$, we show the optimal solution analytically by solving first order Karush-Kuhn-Tucker (KKT) conditions [15] for the resultant non-linear constrained optimization problem. For $N_c > 1$, we show the optimal solution of the problem numerically using the MATLAB optimization toolbox. Obtained numerical results show consistency with our theoretical findings.

#### Analytical

In this section, we find the optimal solution of equations (6.13), (6.14) and (6.15) analytically for the two-state multi-agent system. For $N_c = 1$, we write the resultant optimization problem as

$$\min_{p_{k|k}, q_{k|k}} \ \mathbb{J}_k \tag{6.22}$$

s.t.

$$(p_{k|k}, q_{k|k}) \in [0, 1] \times [0, 1], \tag{6.23a}$$

$$p_{k+m|k} = q_{k+m|k} = 0 \ \ for \ \ m = 1, 2, \cdots, N_p, \tag{6.23b}$$

$$\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] < e_k^2 \ \ for \ \ e_k \neq 0, \tag{6.23c}$$

$$\mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] = e_k^2 \;\; for \;\; e_k = 0. \tag{6.23d}$$

The cost function $\mathbb{J}_k$ over a prediction horizon $N_p$ (see equation (6.15)) is expressed as

$$\mathbb{J}_k = N_p \mathbb{E}[e_{k+1}^2 \mid \mathcal{F}_k] + p_{k|k}^2 + q_{k|k}^2. \tag{6.24}$$

Now we write the Lagrangian [15] of the optimization problem (equations (6.22), (6.23) and (6.24)) as $L(p_{k|k}, q_{k|k}, \lambda_i, i = 1, 2, \cdots, 5) = \mathbb{J}_k(1 + \lambda_5/N_p) + \lambda_1(p_{k|k} - 1) + \lambda_2(-p_{k|k}) + \lambda_3(q_{k|k} - 1) + \lambda_4(-q_{k|k}) + (\lambda_5/N_p)(-p_{k|k}^2 - q_{k|k}^2 - N_p e_k^2)$. Here, $\lambda_i, i = 1, 2, \cdots, 5$ are non-negative Lagrange multipliers. We derive first order KKT conditions [15] for the optimization problem which is represented here in the following form:

$$\frac{\partial L}{\partial p_{k|k}} = \frac{\partial L}{\partial q_{k|k}} = 0, \tag{6.25a}$$

$$\lambda_1(p_{k|k} - 1) = 0, \tag{6.25b}$$

$$\lambda_2(-p_{k|k}) = 0, \tag{6.25c}$$

$$\lambda_3(q_{k|k} - 1) = 0, \tag{6.25d}$$

$$\lambda_4(-q_{k|k}) = 0, \tag{6.25e}$$

$$\lambda_5(\mathbb{J}_k - p_{k|k}^2 - q_{k|k}^2 - N_p e_k^2) = 0, \tag{6.25f}$$

$$\lambda_i \geq 0 \;\; for \; i = 1, 2, \cdots, 5. \tag{6.25g}$$

A closed-form of the optimal solution which satisfy equation (6.25) is presented in

Table 6.1.

| Parameter | $e_k < 0$ | $e_k > 0$ |
|---|---|---|
| $p_{k\|k}$ | $-\frac{N_p N_k(1+2e_k)}{2(1-N_k N_p+N_k^2 N_p)}$ | 0 |
| $q_{k\|k}$ | 0 | $-\frac{N_p(N-N_k)(1-2e_k)}{2(1+N_p(N-N_k)(N-(N_k-1)))}$ |
| $\lambda_1$ | 0 | 0 |
| $\lambda_2$ | 0 | $\alpha(k)(>0)$ |
| $\lambda_3$ | 0 | 0 |
| $\lambda_4$ | $\beta(k)(>0)$ | 0 |
| $\lambda_5$ | 0 | 0 |

Table 6.1: Optimal Solution: Two-State Multi-Agent System for $N_c = 1$.

In Table 6.1, $\alpha(k) = \frac{N_p N_k}{1+N_p(N-N_k)(N-N_k-1)}(1 + 2e_k + N_p(N - N_k)(2(N - N_r) - 1))$ and $\beta(k) = \frac{N-N_k)N_p}{1-N_k N_p+N_k^2 N_p}(1 - 2e_k + N_p N_k(2N_r - 1))$. Now with a bit of algebraic analysis, it is easy to verify that $p_{k|k} \in (0, 1)$ for $e_k < 0$ and $q_{k|k} \in (0, 1)$ for $e_k > 0$. We find that the solution presented in Table 6.1 is optimal for all $e_k$ except cases when $e_k = 0$, $e_k = N_r > 0$ and $e_k = (N_r - N) < 0$. For $e_k = 0$, the optimal solution is given by $p_{k|k} = q_{k|k} = 0$, $\lambda_1 = \lambda_3 = 0$, $\lambda_2 = (1 + \lambda_5/N_p)N_p N_k$, $\lambda_4 = (1 + \lambda_5/N_p)N_p(N - N_k)$ and $\lambda_5 > 0$. For $e_k = N_r > 0$, the solution is same as represented in Table 6.1 except $\lambda_2 = \alpha(k) = 0$. For $e_k = (N_r - N) < 0$, again the solution is same as represented in Table 6.1 except $\lambda_4 = \beta(k) = 0$. Because of the algebraic complexity (nonlinearity) in optimal finding closed-form solution for $N_c > 1$, next, we solve the optimization problem numerically.

**Simulation**

We consider a two-state multi-agent system consisting of $N = 1000$ independently behaving agents. Dynamical behavior of this system is defined by equations (6.8) and (6.12). Initially, at time step $k = 0$, there are $N_{k=0} = 300$ agents possessing 'ON' state

and rest of 800 agents possessing 'OFF' state. The number of desired agents with 'ON' state, $N_r$, is set to 200. To show the efficacy of the control design for large prediction and control horizons, we choose $N_p = 1000$ and $N_c = 13$. With this configuration of the system, we solve the optimization problem stated by equations (6.13), (6.14) and (6.15) numerically in the MATLAB simulation environment. For this, we use the MATLAB inbuilt optimization function for the non-linear constrained optimization problems, 'fmincon' and the binomial random variable generator function 'binorand'. At each time step $t$, the optimization problem (equations (6.13), (6.14) and (6.15)) is solved for the system using the measured system error at the previous time step. The first optimally computed transition probability pair $(p_{k|k}, q_{k|k})$ is then broadcast to all agents in the system. With these probabilities, each of the agents within the system makes a random decision about the switching of their states which follows a Bernoulli distribution and thus contribute towards the next time step system error. Fig. 6.3 shows realizations of the system error $e_k$ and optimal transition probabilities $p_{k|k}$ and $q_{k|k}$ for this particular system. For most of the systems we studied by varying number of agents, prediction horizon, control horizon, initial conditions and the desired number of agents with 'ON' state, we found that the system error converge to zero within 20 time steps. During this study, we also observed that optimal value of transition probabilities $p_{k|k} = 0$ and $q_{k|k} > 0$ whenever $e_k > 0$ and $p_{k|k} > 0$ and $q_{k|k} = 0$ whenever $e_k < 0$ which is consistent with the analytical solution for $N_c = 1$.

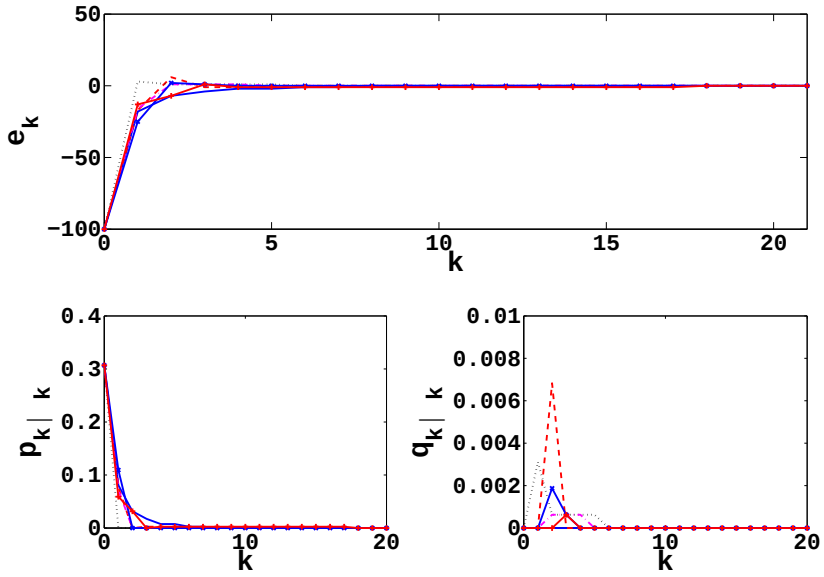



Figure 6.3: Realizations of the system error $e_k$ and optimally computed transition probabilities $p_{k|k}$ and $q_{k|k}$ within the framework of the "Broadcast RHC" for a two-state multi-agent system consists of 1000 stochastically behaving agents.

## 6.4 Stochastic Multi-State Multi-Agent System

In this section, we generalize the broadcast finite stochastic RHC design by extending the control problem from a two-state multi-agent system to a $N_s$ ($\geq 3$) state multi-agent system. In this $N_s$ state system, each agent is capable of making random transitions from one state to another among $N_s$ available states. As an example, Figure 6.4 illustrates the state transition behavior of agents in a three-state multi-agent system.

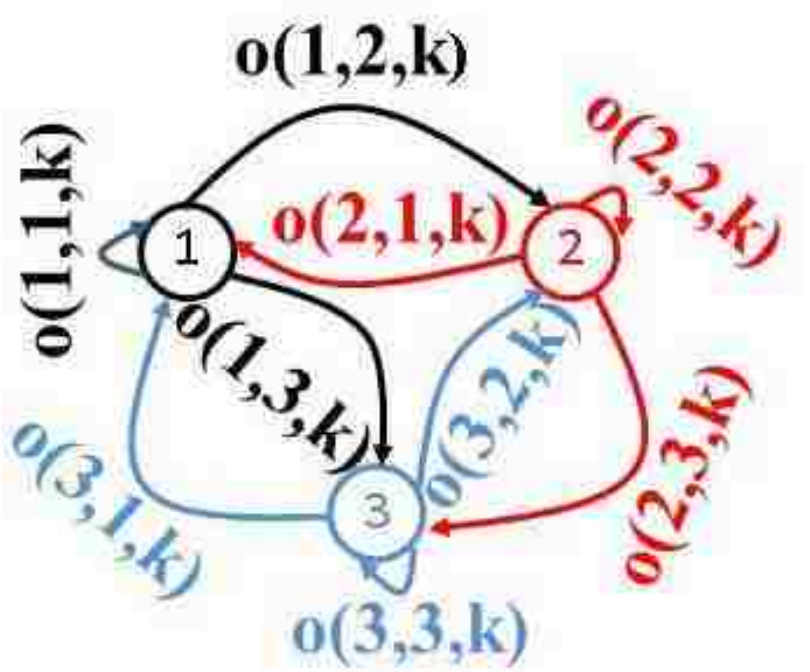



Figure 6.4: State transition behavior of agents in a three-state multi-agent system.

In the rest of this chapter, we derive the predictive dynamical model of the system and guarantee the convergence and stability of the closed-loop system under broadcast stochastic RHC.

### 6.4.1 System Model

We define $e_{k,m} = N_{r,m} - N_{k,m}$ as the system error at state $m$ at time $k$. Here $m = 1, 2, \cdots, N_s$. $N_{r,m}$ is the desired number of agents at state $m$ and $N_{k,m}$ is the total number of agents possessing the state $m$ at time $k$. $\sum_{m=1}^{N_s} N_{r,m} = \sum_{m=1}^{N_s} N_{k,m} = N$ defines the total number of agents present in the system. Clearly, $\sum_{m=1}^{N_s} e_{k,m} = 0$ for all $k \geq 0$. With this, we write the conditional expectation and variance of the error at state $m$ at time $k+1$ as

$$\mathbb{E}[e_{k+1,m} \mid \mathcal{G}_k] = N_{r,m} - \sum_{l=1}^{N_s} (N_{r,l} - e_{k,l}) o_{k|k}(l, m), \tag{6.26a}$$

$$Var[e_{k+1,m} \mid \mathcal{G}_k] = \sum_{l=1}^{N_s} (N_{r,l} - e_{k,l}) o_{k|k}(l, m)(1 - o_{k|k}(l, m)). \tag{6.26b}$$

Here $\mathcal{G}_k$ is a $\sigma$-algebra generated by $\{\{e_{i,m}\}_{i=0}^{k}\}_{m=1}^{N_s}$, as defined in [41]. $o_{k|k}(l, m)$ is the transition probability with which an agent switches the state from $l$ to $m$ at time $k$. Conservation of transition probabilities at any state $m$ satisfies

$$\sum_{l=1}^{N_s} o_{k|k}(m, l) = 1. \tag{6.27}$$

**Remark 6.4.1.** *In equation (6.26), we have assumed that the system error at each state is measurable at time $k$. The only measurable information in the broadcast framework is the aggregate emergent behavior of agents which in our case is the cumulative error $e_k = \sum_{m=1}^{N_s} e_{k,m}^2$. The formulation can be extended to incorporate an optimal observer design to estimate the system error $e_{k,m}$ for $m = 1, 2, \cdots, N_s$ using*

*the cumulative error $e_k$ and the known transition probabilities at time $k-1$. One such observer design has recently been reported for inhomogeneous population Markov chains using the approach of Kalman filter [120].*

Now using conditional expectation properties $\mathbb{E}[e^2_{k+j,m} \mid \mathcal{G}_{k+j-1}] = Var(e_{k+j,m} \mid \mathcal{G}_{k+j-1}) + (\mathbb{E}[e_{k+j} \mid \mathcal{F}_{k+j-1}])^2$ and $\mathbb{E}[e^2_{k+j,m} \mid \mathcal{G}_k] = \mathbb{E}[\mathbb{E}[e^2_{k+j,m} \mid \mathcal{G}_{k+j-1}] \mid \mathcal{G}_k]$ along with equation (6.26), it is not difficult to write the expression for the conditional expectation $\mathbb{E}[e^2_{k+j,m} \mid \mathcal{G}_k]$ at a future time $k+j$ for $j > 1$.

### 6.4.2 Problem Formulation

We formulate a constrained non-linear finite stochastic RHC problem for $N_s$ state multi-agent system as follows:

$$\min_{\substack{o_{k|k}(l,m),\cdots,o_{k+N_c-1|k}(l,m) \\ l=1,2,\cdots,N_s \\ m=1,2,\cdots,N_s \\ l\neq m}} \mathbb{J}_k \tag{6.28}$$

s.t.

$$o_{k+j|k}(l,m) \in [0,1] \ \ for \ \ 0 \leq j \leq N_c - 1, \ l \neq m, \tag{6.29a}$$

$$o_{k+j|k}(l,m) = 0 \ \ for \ \ j \geq N_c, \ l \neq m, \tag{6.29b}$$

$$\sum_{m\neq l=1}^{N_s} o_{k+j|k}(l,m) \leq 1 \ \ for \ \ 0 \leq j \leq N_c - 1, \tag{6.29c}$$

$$\mathbb{E}[e_{k+j+1} \mid \mathcal{G}_k] < e_k \ \ for \ \ e_k \neq 0, \ 0 \leq j \leq N_c - 1, \tag{6.29d}$$

$$\mathbb{E}[e_{k+j+1} \mid \mathcal{G}_k] = e_k \;\; for \;\; e_k = 0,\; 0 \leq j \leq N_c - 1. \tag{6.29e}$$

The cost function $\mathbb{J}_k$ is defined as

$$\mathbb{J}_k = \sum_{j=0}^{N_p-1} \mathbb{E}[e_{k+j+1} \mid \mathcal{G}_k] + \sum_{j=0}^{N_c-1} \sum_{l=1}^{N_s} \sum_{m \neq l=1}^{N_s} o^2_{k+j|k}(l,m) \tag{6.30}$$

where $e_k = \sum_{m=1}^{N_s} e^2_{k,m}$ and $o_{k+j|k}(l,m)$ is the transition probability with which an agent switches its state from $l$ to $m$ at a future time $k+j$ given the information till time $k$.

### 6.4.3 Stability and Convergence

In this section, we show that the control problem defined by equations (6.28), (6.29) and (6.30) is feasible. This feasibility is sufficient to conclude the stability and convergence of the control problem, as stated in Theorem 6.4.3.

**Claim 6.4.2.** *The broadcast finite stochastic RHC problem defined by equations (6.28), (6.29) and (6.30) is feasible for the stochastic $N_s$ state multi-agent system.*

*Proof.* To prove Claim 6.4.2, it is sufficient to show the existence of transition probability $o_{k+j|k}(l,m)$ for $0 \leq j \leq N_c - 1$, $l = 1, 2, \cdots, N_s$, $m = 1, 2, \cdots, N_s$ and $l \neq m$ which satisfies equation (6.29). We first consider the case when $e_k = 0$. We observe that if we set $o_{k+j|k}(l,m) = 0$ for $j = 0, 1, \cdots, N_p - 1$ whenever $l \neq m$ and $o_{k+j|k}(l,m) = 1$ for $j = 0, 1, \cdots, N_p - 1$ whenever $l = m$, then equation (6.29) satisfies for $e_k = 0$. Thus for $e_k = 0$, the optimization problem is feasible. For

$e_k \neq 0$, we choose two arbitrary state $l_1$ and $l_2$ among $N_s$ available states. Now we set $o_{k+j|k}(l,m) = 0$ for $j = 0, 1, \cdots, N_p - 1$ whenever $l \neq m$ and $o_{k+j|k}(l,m) = 1$ for $j = 0, 1, \cdots, N_p - 1$ whenever $l = m$ except for $(l,m) \in \{(l_1,l_1), (l_1,l_2), (l_2,l_1), (l_2,l_2)\}$. Clearly, $\mathbb{E}[e^2_{k+j+1,m} \mid \mathcal{G}_k] = e^2_{k,m}$ for $j = 0, 1, \cdots, N_p - 1$ and $m = 1, 2, \cdots, N_s$ except for $m = l_1$ and $m = l_2$. Now if we set $p_{k+j|k} = o_{k+j|k}(l_1, l_2)$ and $q_{k+j|k} = o_{k+j|k}(l_2, l_1)$ for $j = 0, 1, \cdots, N_p - 1$ in the two-state system shown in Figure 6.1, Claim 6.3.3 guarantees the existence of $o_{k+j|k}(l_1, l_2) \in [0, 1]$ and $o_{k+j|k}(l_2, l_1) \in [0, 1]$ such that $\mathbb{E}[e^2_{k+j+1,m} \mid \mathcal{G}_k] < e^2_{k,m}$ for $m = l_1, l_2$ and $j = 0, 1, \cdots, N_p - 1$. Since $e_{k+j+1} = \sum_{m=1}^{N_s} e^2_{k+j+1,m}$ by definition, $\mathbb{E}[e_{k+j+1} \mid \mathcal{G}_k] < e_k$ with these transition probabilities. Thus the optimization problem is also feasible for $e_k \neq 0$. □

**Theorem 6.4.3.** *If the optimization problem defined by equations (6.28), (6.29) and (6.30) is feasible for all $k \geq 0$ then*

$$e_{k,m} \xrightarrow{a.s.} 0 \qquad for \quad m = 1, \cdots, N_s.$$

*Further,*

$$\mathbb{J}_k \xrightarrow{a.s.} 0.$$

*Proof.* **step 1:** Results of Claim 6.4.2 guarantee the existence of transition probabilities such that the optimization problem defined by equations (6.28), (6.29) and (6.30) is feasible. Moreover under these transition probabilities, $\mathbb{E}[e_{k+1} \mid \mathcal{G}_k] \leq e_k$ for all $k \geq 0$. Thus $\{e_k\}_{k\geq 0}$ is supermartingale.

**step 2:** By the supermartingale convergence theorem, $e_k \overset{k\to a.s.}{\longrightarrow} X$, where $X$ is a limit random variable.

**step 3:** Clearly, $|e_{k,m}^2| \leq \max\{N_{r,m}^2, (N - N_{r,m})^2\}$ for all $k \geq 0$ and $m = 1, 2, \cdots, N_s$. Also, $\mathbb{E}e_{k,m}^2 < \infty$ and $e_k \overset{k\to a.s.}{\longrightarrow} X$ Thus by the bounded convergence theorem, $\mathbb{E}e_k \overset{k\to a.s.}{\longrightarrow} \mathbb{E}X$.

**step 4:** Since $\mathbb{E}e_{k+1} < \mathbb{E}e_k$ for all $e_k \neq 0$ and $k \geq 0$, $\mathbb{E}e_k \downarrow 0$. Thus, $\mathbb{E}X = 0$.

**step 5:** Thus $e_k \to 0$ in probability by Chebyshev inequality [41] (If $z > 0$ and $\mathbb{E}|X_k|^z \to 0$ then $X_k \to 0$ in probability).

**step 6:** From step 2, we know that $e_k$ converges to $X$ a.s. This means that $e_k$ also converges to $X$ in probability. From step 5, $e_k$ converges to 0 in probability. Therefore, $X$ must be 0. Thus $e_k \overset{k\to a.s.}{\longrightarrow} 0$.

**step 7:** Since $e_k = \sum_{m=1}^{N_s} e_{k,m}^2$ by definition, $e_{k,m}^2 \overset{k\to a.s.}{\longrightarrow} 0$ for $m = 1, 2, \cdots, N_s$.

**step 8:** Since $e_{k,m}^2 \overset{k\to\infty}{\longrightarrow} 0$ $a.s.$, $Pr(|e_{k,m}| > \epsilon\ i.o.) = 0$ for all $\epsilon > 0$ (here $i.o.$ stands for infinitely often). Thus, $e_{k,m} \overset{k\to a.s.}{\longrightarrow} 0$ for $m = 1, 2, \cdots, N_s$.

**step 8:** Since the optimization problem defined by equations (6.28), (6.29) and (6.30) is feasible and $e_k \overset{k\to a.s.}{\longrightarrow} 0$, $\mathbb{E}[e_{k+j+1} \mid \mathcal{G}_k] \overset{k\to a.s.}{\longrightarrow} 0$ for $j = 0, 1, \cdots, N_c - 1$ (see equation (6.29e)). Therefore, $\mathbb{E}[e_{k+j+1,m} \mid \mathcal{G}_k] \overset{k\to a.s.}{\longrightarrow} 0$ and $Var[e_{k+j+1,m} \mid \mathcal{G}_k] \overset{k\to a.s.}{\longrightarrow} 0$ for $j = 0, 1, \cdots, N_c - 1$ and $m = 1, 2, \cdots, N_s$. By using these results in equations (6.26) and (6.27) and the argument of optimality of inputs presented in the proof of Theorem 6.3.4, it is easy to verify that $o_{k+j|k}(m, l) \overset{k\to\infty}{\longrightarrow} 0$ for $m = 1, 2, \cdots, N_s$, $l = 1, 2, \cdots, N_s$, $m \neq l$ and $j = 0, 1, \cdots, N_c - 1$.

**step 9:** Since $\sum_{j=0}^{N_p-1} \mathbb{E}[e_{k+j+1} \mid \mathcal{G}_k] \leq N_p e_k$, $e_k \overset{k\to a.s.}{\longrightarrow} 0$ together with step 8 conclude $\mathbb{J}_k \overset{k\to a.s.}{\longrightarrow} 0$.

□

## 6.5 Concluding Remarks

In this chapter, we have proposed a centralized optimal control strategy called "Broadcast Stochastic Receding Horizon Control (BSRHC)" by incorporating the concept of broadcast in a probabilistic framework of receding horizon control. The applicability of this strategy is shown by achieving a desired aggregate system behavior in a swarm of stochastically behaving identical agents when the available feedback information is limited by the system. Almost sure (with probability 1) convergence of the system behavior to the desired one has been shown by utilizing supermartingale theory. Stability and convergence results have been presented in generalized form applicable to various similar systems. The efficacy of the controller design has been illustrated by presenting analytical and simulation results for a two-state multi-agent system.

# Chapter 7

# Investigation on Optimal Trapping of Brownian Ensemble

## 7.1 Introduction

As soon as a drop of ink is placed on the surface of water, it diffuses and eventually covers the entire volume of the water in a container. This observation leads to visualization of an important phenomenon, namely, Brownian motion. This motion, also known as diffusion, is the cumulative effect of multitudinous interactions of ink particles with water molecules. The question is whether this diffusive phenomenon can be controlled, even in an idealized theoretical sense, such that the volume occupied by ink particles in the container can be minimized. More importantly, if we place several well separated drops of ink on the surface of water, is it possible to suppress the diffusion of these droplets in a manner such that they remain well separated forever? If so, what would be the best control strategy? In this chapter, we attempt to answer this question theoretically by developing an optimal control framework.

In the previous chapter, we developed an optimal centralized control strategy called "Broadcast Stochastic Receding Horizon Control" (BSRHC) and applied this strategy to stabilize the aggregate system behavior in a vast number of stochastically behaving agents. Each agent in the ensemble was capable of making random transitions from one state to another among a finite number of available states. Assuming that a agent possesses only two states, the transition from one state to another can also be viewed as the transition in the direction of movement of a particle undergoing Brownian motion in one dimensional space. Thus in this sense, a particle undergoing Brownian motion in one dimensional space can be represented by an agent having two states. This representation suggests a potential application of BSRHC strategy in addressing the problem of optimal trapping of Brownian ensemble in a smallest possible trapping region for an extended period of time and thus the optimal suppression of Brownian fluctuations in an ensemble of particles undergoing Brownian motion.

In this chapter, we use the BSRHC framework developed in the previous chapter with relevant modifications to demonstrate the trapping of an ensemble of Brownian particles in the smallest trapping region in one, two and three dimensional homogeneous medium. The modified BSRHC strategy uses the measured distance of a particle outside the trapping region that is closest to the origin, as the only available feedback information and designs the optimal control inputs in a predictive framework. By a Brownian particle, we mean that the dynamical behavior of the particle is represented by the standard Brownian motion [41] which is approximated here by a discrete time random walk model. We assume that particles behave independently

within the ensemble, i.e. we neglect inter-particle interactions as well as hydrodynamic forces induced by the medium. We also assume that a uniform external force field is acting on all the particles in the Brownian ensemble.

## 7.2 System Model

We consider a system consisting of $n$ particles. Each particle in the system undergoes independent Brownian motion in a homogeneous medium which is approximated here with a discrete time random walk model. To define the motion of the $j^{th}$ particle in one dimensional space, we assume that there are $N$ independent and identically distributed (i.i.d.) replications of the $j^{th}$ particle indexed as $\{j_1, j_2, \cdots, j_N\}$. The displacement $\Delta X_{j_i,k}$ of the $j_i^{th}$ replication between the discrete time $k$ and $k+1$ is approximated as

$$\Delta X_{j_i,k} = \begin{cases} \delta_x & \text{w.p.} \quad p_{j,k}^x = 1 - q_{j,k}^x, \\ -\delta_x & \text{w.p.} \quad q_{j,k}^x. \end{cases} \tag{7.1}$$

Here "w.p." is the abbreviation of "with probability". $\delta_x > 0$ is the displacement of the $j_i^{th}$ replication of the $j^{th}$ particle on the x-axis. $q_{j,k}^x$ is the probability with which the $j_i^{th}$ replication of the $j^{th}$ particle moves towards the origin on the x-axis between the discrete time step $k$ and $k+1$. Equation (7.1) holds if the $j^{th}$ particle is located at the origin or on the positive side of the x-axis at time $k$. If the $j^{th}$ particle is located on the negative side of the x-axis at time $k$, the displacement $\Delta X_{j_i,k}$ of the

$j_i^{th}$ replication between the discrete time $k$ and $k+1$ is approximated as

$$\Delta X_{j_i,k} = \begin{cases} -\delta_x & \text{w.p.} \qquad p_{j,k}^x = 1 - q_{j,k}^x, \\ \delta_x & \text{w.p.} \qquad q_{j,k}^x. \end{cases} \tag{7.2}$$

The net displacement of the $j^{th}$ particle between the discrete time step $k$ and $k+1$ along x-axis is defined as $\xi_{j,k}^x = \frac{1}{N}\sum_{i=1}^{N} \Delta X_{j_i,k}$, which is the average displacement over $N$ replications. We define the position of the $j^{th}$ particle at the time step $k$ as $S_{j,k} = \sum_{l=1}^{k-1} \xi_{j,l}^x$ with $S_{j,1} = 0$. Now by considering $q_{j,k}^x = 0.5$ for all $k \geq 1$, we obtain the discrete time representation of the one dimensional unbiased Brownian motion of the $j^{th}$ particle as follows:

We define a sequence of random variable $\bar{\xi}_{j,k}^x$ by normalizing the random variable $\xi_{j,k}^x$ as $\bar{\xi}_{j,k}^x = \frac{\sqrt{N}}{\delta_x}\xi_{j,k}^x$ for $k = 1, 2, \cdots, m$. Clearly with $q_{j,k}^x = 0.5$, $\bar{\xi}_{j,1}^x, \bar{\xi}_{j,2}^x, \cdots$ are i.i.d. random variables with mean 0 and variance 1. Now writing $\bar{S}_{j,m} = \bar{\xi}_{j,1}^x + \cdots + \bar{\xi}_{j,m}^x$ as the $m^{th}$ partial sum, a continuous trajectory $B_{j,t}^m$ on $0 \leq t \leq 1$ is defined as

$$B_{j,t}^m = \begin{cases} \bar{S}_{j,k}/\sqrt{k} & \text{if} \qquad t = k/m, \\ \bar{S}_{j,k}/\sqrt{k} + \bar{B}_{j,t}^m & \text{if} \qquad t \in (k/m, (k+1)/m). \end{cases} \tag{7.3}$$

Here $\bar{B}_{j,t}^m = (mt - [mt])\bar{\xi}_{j,[mt+1]}^x/\sqrt{k}$. $[\cdot]$ is the greatest integer of the argument. With this, the following well known Donsker's theorem [41] tells us that $B_{j,t}^m$ converges to the standard one dimensional Brownian motion $B_t$ in distribution as $m \longrightarrow \infty$.

**Theorem 7.2.1.** ***(Donsker's Theorem ([41]))*** *As $m \longrightarrow \infty$, $B_{j,t}^m \implies B_t$ i.e. the associated measures on $\mathcal{C}[0,1]$ converges weakly. Here $B_t$ is the standard Brownian motion.*

We extend the discrete time representation of Brownian motion in two and three dimensional space by defining the net displacement of the $j^{th}$ particle between the discrete time step $k$ and $k+1$ as $\xi^y_{j,k} = \frac{1}{N}\sum_{i=1}^{N} \Delta Y_{j_i,k}$ and $\xi^z_{j,k} = \frac{1}{N}\sum_{i=1}^{N} \Delta Z_{j_i,k}$ in the $y$ and $z$ direction respectively. $\Delta Y_{j_i,k}$ and $\Delta Z_{j_i,k}$ are defined by equations (7.1) and (7.2) where we replace $\Delta X_{j_i,k}$ by $\Delta Y_{j_i,k}$ and $\Delta Z_{j_i,k}$ respectively in the left hand side of equations (7.1) and (7.2), and $x$ by $y$ and $z$ respectively in the right hand side of equations (7.1) and (7.2). With this, we define the position of the $j^{th}$ particle in two and three dimensional space at the time step $k$ as $S_{j,k} = \sum_{l=1}^{k-1}(\xi^x_{j,l}\hat{e}_x + \xi^y_{j,l}\hat{e}_y)$ and $S_{j,k} = \sum_{l=1}^{k-1}(\xi^x_{j,l}\hat{e}_x + \xi^y_{j,l}\hat{e}_y + \xi^z_{j,l}\hat{e}_z)$ respectively. Here we assume that $S_{j,1} = 0$. $\hat{e}_x$, $\hat{e}_y$ and $\hat{e}_z$ are unit vectors in $x$, $y$ and $z$ directions respectively.

## 7.3 Problem Formulation

### 7.3.1 Region of Trapping

To explore the possibility of trapping a particle in spatial dimension, it is necessary to quantify how far the particle can travel from the origin in a fixed number of discrete time steps for a given system. Traditionally in both experimental and theoretical designs, the standard notion for measuring this spreading is the mean square distance. Before we compute this quantitatively, we define a convenient notion for measuring the distance of a particle from the origin at time $k$. At time $k$, we define the minimum distance of the $j^{th}$ particle from the origin in the $i^{th}$ direction as $d^i_{j,k} = |d^i_{j,k-1} + \xi^i_{j,k-1}|$ where $i \in \{x, y, z\}$ with $d^i_{j,1} = 0$. Here, $\xi^i_{j,k-1}$ is given by equation (7.1). $|\cdot|$ represents the absolute value of the underlying argument.

**Remark 7.3.1.** *Since* $|\xi^i_{j,k-1}| \leq \delta_x$, $d^i_{j,k-1} + \xi^i_{j,k-1}$ *can be negative. The modulus*

*operation in the recursive expression i.e.* $d^i_{j,k} = |d^i_{j,k-1} + \xi^i_{j,k-1}|$ *ensures that the computed distance of the* $j^{th}$ *particle at time* $k$ *from the origin is always positive. As an example, lets consider the diffusion of the* $j^{th}$ *particle in one dimensional space. If the distance of the particle at time* $k-1$ *is* $\delta_x/2$ *i.e.* $d^x_{j,k-1} = \delta_x/2$ *and the particle moves towards the origin by a distance* $\delta_x$ *between time* $k-1$ *and* $k$ *i.e.* $\xi^x_{j,k-1} = -\delta_x$ *(see equation (7.1)) then the position of the particle at time* $k$ *will be* $-\delta_x/2$ *and the distance of the particle from the origin will be* $\delta_x/2$*. Now if we compute* $d^x_{j,k}$ *using our recursive definition, it is easy to verify that* $d^x_{j,k} = |-\delta_x/2| = \delta_x/2$*. Moreover, if* $d^x_{j,k} \geq \delta_x$ *then* $d^x_{j,k} = d^x_{j,k-1} + \xi^x_{j,k-1}$*.*

*The position* $S_{j,k}$ *of the particle at time* $k$ *can be computed using* $\xi^x_{j,k-1}$ *and* $S_{j,k-1}$*. If* $S_{j,k-1} \geq 0$ *then* $S_{j,k} = S_{j,k-1} + \xi^x_{j,k-1}$ *and if* $S_{j,k-1} < 0$ *then* $S_{j,k} = S_{j,k-1} - \xi^x_{j,k-1}$ *(clear from equations (7.1) and (7.2)). Here* $\xi^x_{j,k-1}$ *is computed using equation (7.1). It is not difficult to verify that the distance* $d^x_{j,k}$ *is essentially the same as the modulus of the position* $S_{j,k}$ *i.e.* $d^x_{j,k} = |S_{j,k}|$*. Table 7.1 shows computations of* $d^x_{j,k}$ *and* $S_{j,k}$ *for a particular case.*

| $k$ | $\xi^x_{j,k}$ | $d^x_{j,k+1}$ with $d^x_{j,1} = \delta_x/2$ | $S_{j,k+1}$ with $S_{j,1} = \delta_x/2$ |
|---|---|---|---|
| 1 | $-\delta_x$ (towards origin) | $\delta_x/2$ | $-\delta_x/2$ |
| 2 | $\delta_x$ (away from origin) | $3\delta_x/2$ | $-3\delta_x/2$ |
| 3 | $-\delta_x$ (towards origin) | $\delta_x/2$ | $-\delta_x/2$ |
| 4 | $-\delta_x$ (towards origin) | $\delta_x/2$ | $\delta_x/2$ |
| 5 | $\delta_x$ (away from origin) | $3\delta_x/2$ | $3\delta_x/2$ |
| 6 | $-\delta_x/2$ (towards origin) | $\delta_x$ | $\delta_x$ |

Table 7.1: Relation between $d^x_{j,k}$ and $S_{j,k}$

With these new notations, next we quantify the spreading of the particle trajectory in $k$ time steps by computing the mean square distance. In one dimensional space,

the conditional expectation of $(d^x_{j,k})^2$ for $k \geq 2$ is given as

$$\begin{aligned} \mathbb{E}[(d^x_{j,k})^2 \mid d^x_{j,k-1}] &= \mathbb{E}[(|d^x_{j,k-1} + \xi^x_{j,k-1}|)^2 \mid d^x_{j,k-1}] \\ &= (d^x_{j,k-1})^2 + 2d^x_{j,k-1}\mathbb{E}[\xi^x_{j,k-1}] + \mathbb{E}[(\xi^x_{j,k-1})^2]. \end{aligned} \tag{7.4}$$

Here $\mathbb{E}[\cdot]$ is the expectation of the underlying argument. Substituting $\mathbb{E}[\xi^x_{j,k-1}] = \delta_x(1 - 2q^x_{j,k-1})$ and $\mathbb{E}[(\xi^x_{j,k-1})^2] = \frac{\delta_x^2}{N}(1 + (N-1)(1 - 2q^x_{j,k-1})^2)$ in equation (7.4) using equation (7.1), we obtain

$$\begin{aligned} \mathbb{E}[(d^x_{j,k})^2 \mid d^x_{j,k-1}] = (d^x_{j,k-1})^2 &+ 2d^x_{j,k-1}\delta_x(1 - 2q^x_{j,k-1}) \\ &+ \frac{\delta_x^2}{N}(1 + (N-1)(1 - 2q^x_{j,k-1})^2). \end{aligned} \tag{7.5}$$

The mean square distance traveled by the $j^{th}$ particle till the time step $k$ is computed recursively by taking the expectation on both sides of equation (7.5) at $q^x_{j,k-1} = 0.5$:

$$\begin{aligned} \mathbb{E}[(d^x_{j,k})^2] &= \mathbb{E}[\mathbb{E}[(d^x_{j,k})^2 \mid d^x_{j,k-1}]] \\ &= (k-1)\delta_x^2/N. \end{aligned} \tag{7.6}$$

The computed expression for the mean square distance clearly shows that the spreading of the particle trajectory grows proportionally with $k$. In particular, as $k$ tends to infinity, the mean square distance reaches infinity. Next by substituting $q^x_{j,k-1} = 0.5$ in equation (7.5) and using the well known Chebyshev's inequality, we compute an upper bound of the conditional probability $Pr\{d^x_{j,k} \geq d^x_{j,k-1} \mid d^x_{j,k-1}\}$ as

$$\begin{aligned} Pr\{d^x_{j,k} \geq d^x_{j,k-1} \mid d^x_{j,k-1}\} &\leq \frac{\mathbb{E}[(d^x_{j,k})^2 \mid d^x_{j,k-1}]}{(d^x_{j,k-1})^2} \\ &= 1 + \frac{\delta_x^2}{N(d^x_{j,k-1})^2}. \end{aligned} \tag{7.7}$$

Equation (7.7) shows that with $q^x_{j,k-1} = 0.5$, the particle can travel further away from the origin in the next time step with a probability, which is bounded above by a quantity greater than one. By reducing the upper bound on this conditional probability, it is possible to suppress the spreading of the trajectory of the particle.

Traditionally, the continuous manipulation of $q^x_{j,k-1}$ using a feedback controller results in the suppression of Brownian trajectories of particles and thus the trapping of particles in a desired spatial region of the medium. The minimum spatial region in which particles can be trapped depends on the maximum limit on the applied external field, properties of particles, the nature of the medium in which particles are diffusing and an efficient feedback control architecture for designing external forces. These conditions together provide an upper bound on $q^x_{j,k-1}$ and defines the minimum spatial region in which it is possible to trap particles over an extended period of time. We assume the existence of such an upper bound on $q^x_{j,k-1}$, $j = 1, 2, \cdots, n$ for our model and design the minimum possible region of the trapping of an ensemble of particles using the minimum possible feedback information measured from the system.

We begin with the one dimensional system consisting of $n$ independently behaving particles. We assume that the probability $q^x_{j,k-1}$ for $j = 1, 2, \cdots, n$ belongs to the interval $[0, q^x_m]$ where $q^x_m \leq 1$. With this, we find conditions on $q^x_{j,k-1}$ and $d^x_{j,k-1}$ by enforcing $\mathbb{E}[(d^x_{j,k})^2 \mid d^x_{j,k-1}] < (d^x_{j,k-1})^2$ which makes the right hand side of equation (7.7) strictly less than one.

It should be noted that the right side of equation (7.5) is always less than $(d^x_{j,k-1})^2 + 2d^x_{j,k-1}\delta_x(1-2q^x_{j,k-1}) + \delta_x^2$. Thus by making this quantity less than $(d^x_{j,k-1})^2$, we obtain $q^x_{j,k-1} > 0.5 + \delta_x/(4d^x_{j,k-1})$. Since $q^x_{j,k-1} \leq q^x_m$, $0.5 + \delta_x/(4d^x_{j,k-1})$ must be less than $q^x_m$ for the existence of $q^x_{j,k-1}$. This results in $d^x_{j,k-1} > \delta_x/(4q^x_m - 2)$. Also, $q^x_{j,k-1} >$

0.5 implies that $q_m^x > 0.5$. Under these conditions, $\mathbb{E}[(d_{j,k}^x)^2 \mid d_{j,k-1}^x] < (d_{j,k-1}^x)^2$. In addition, depending on the maximum strength of the applied force field on a physical system i.e. $q_m^x \in (0.5, 1]$, there exists a minimum region defined by the interval $G^1 := [-\delta_x/(4q_m^x - 2), \delta_x/(4q_m^x - 2)]$ outside which it is possible to reduce the likelihood of further spreading of the trajectory of the $j^{th}$ particle in subsequent time steps. As an example by choosing $q_m^x = 1$, the minimum possible region is given by $G^1 := [-\delta_x/2, \delta_x/2]$.

Recall that our objective is to trap all the particles in an ensemble for an extended period of time by broadcasting the same control inputs, designed using the minimum feedback information measured from the system, to all the particles in the ensemble. We fulfill this objective by first defining the minimum trapping region in which it is possible to trap all the particles forever. Since the maximum distance traveled by a particle in the ensemble in one time step is $\delta_x$, the designed trapping region is $H^1 := [-\gamma_x\delta_x, \gamma_x\delta_x]$. Here $\gamma_x \geq 1$. Next, we define a manipulated transition probability which can be designed by the controller as $q_{k-1}^x \in \bigcap_{j=1}^n (0.5 + \delta_x/(4d_{j,k-1}^x), 1]$. Here we assume that the maximum bound on the transition probability is 1. $\bigcap_{j=1}^n(\cdot)$ is the common interval of the manipulated transition probability among all particles. We define $d_{k-1}^x = \max\{\min_{1\leq j\leq n}\{d_{j,k-1}^x : d_{j,k-1}^x > \gamma_x\delta_x\}, 0\}$ as the minimum measured feedback information which can be used by the controller in designing optimal $q_{k-1}^x$. Thus, $q_{k-1}^x \in (0.5 + \delta_x/(4d_{k-1}^x), 1]$. Now by broadcasting the optimally designed transition probability $q_{k-1}^x$ using $d_{k-1}^x$ as the only measured feedback information to all the particles in the ensemble, the trapping of all particles in $H^1$ can be guaranteed.

We design the minimum region in two and three dimensional spaces in which it would be possible to trap an ensemble of Brownian particles. For this, we first

compute the conditional expectation $\mathbb{E}[(d^i_{j,k})^2 \mid d^i_{j,k-1}] < (d^i_{j,k-1})^2$ for $i \in \{x, y\}$ in two dimensional space and for $i \in \{x, y, z\}$ in three dimensional space:

$$\begin{aligned}\mathbb{E}[(d^x_{j,k})^2 + (d^y_{j,k})^2 \mid d^x_{j,k-1}, d^y_{j,k-1}] = (d^x_{j,k-1})^2 + (d^y_{j,k-1})^2 \\ + \frac{\delta_x^2}{N}(1 + (N-1)(1-2q^x_{j,k-1})^2) \\ + \frac{\delta_y^2}{N}(1 + (N-1)(1-2q^y_{j,k-1})^2) \\ + \alpha_x + \alpha_y.\end{aligned} \tag{7.8a}$$

$$\begin{aligned}\mathbb{E}[(d^x_{j,k})2 + (d^y_{j,k})^2 + (d^z_{j,k})^2 \mid d^x_{j,k-1}, d^y_{j,k-1}, d^z_{j,k-1}] = (d^x_{j,k-1})^2 + (d^y_{j,k-1})^2 + (d^z_{j,k-1})^2 \\ + \frac{\delta_x^2}{N}(1 + (N-1)(1-2q^x_{j,k-1})^2) \\ + \frac{\delta_y^2}{N}(1 + (N-1)(1-2q^y_{j,k-1})^2) \\ + \frac{\delta_z^2}{N}(1 + (N-1)(1-2q^z_{j,k-1})^2) \\ + \alpha_x + \alpha_y + \alpha_z.\end{aligned} \tag{7.8b}$$

Here $\alpha_i = 2d^i_{j,k-1}\delta_i(1 - 2q^i_{j,k-1})$ for $i \in \{x, y, z\}$. The minimum trapping region in two and three dimensional spaces is now computed by forcing $\mathbb{E}[(d^x_{j,k})^2 + (d^y_{j,k})^2 \mid d^x_{j,k-1}, d^y_{j,k-1}] < (d^x_{j,k-1})^2 + (d^y_{j,k-1})^2$ and $\mathbb{E}[(d^x_{j,k})2 + (d^y_{j,k})^2 + (d^z_{j,k})^2 \mid d^x_{j,k-1}, d^y_{j,k-1}, d^z_{j,k-1}] < (d^x_{j,k-1})2 + (d^y_{j,k-1})^2 + (d^z_{j,k-1})^2$ respectively. Using equation (7.8) and the fact that $\frac{\delta_i^2}{N}(1 + (N-1)(1-2q^i_{j,k-1})^2) \leq \delta_i^2$ for $i \in \{x, y, z\}$, we obtain

$$\alpha_x + \alpha_y + \delta_x^2 + \delta_y^2 < 0, \tag{7.9a}$$

$$\alpha_x + \alpha_y + \alpha_z + \delta_x^2 + \delta_y^2 + \delta_z^2 < 0. \tag{7.9b}$$

It should be noted that both expressions in equation (7.8) are satisfied if $\alpha_i + \delta_i^2 < 0$ for $i \in \{x, y, z\}$ i.e. $2d^i_{j,k-1}\delta_i(1 - 2q^i_{j,k-1}) + \delta_i^2 < 0$ for $i \in \{x, y, z\}$. This results in $q^i_{j,k-1} > 0.5 + \delta_i/(4d^i_{j,k-1})$ and $d^i_{j,k-1} > \delta_i/(4q^i_m - 2)$ for $i \in \{x, y, z\}$ where $q^i_m \in (0.5, 1]$ is the maximum transition probability in the $i^{th}$ co-ordinate. With this, we obtain the minimum region in two dimensional space as a rectangle with co-ordinates $(\pm\beta_x, \pm\beta_y)$. Here $\beta_i = \delta_i/(4q^i_m - 2)$ outside which it is possible to reduce the likelihood of further spreading of the trajectory of the $j^{th}$ particle in subsequent time steps. Similarly, we obtain the minimum region in three dimensional space as a cuboid with co-ordinates $(\pm\beta_x, \pm\beta_y, \pm\beta_z)$.

Since the maximum distance traveled by a particle in the ensemble in the $i^{th}$ direction in one time step is $\delta_i$, the designed trapping region in two and three dimensional space is a rectangle with coordinates $(\pm\gamma_x\delta_x, \pm\gamma_y\delta_y)$ and a cuboid with coordinates $(\pm\gamma_x\delta_x, \pm\gamma_y\delta_y, \pm\gamma_z\delta_z)$. Here, $\gamma_i \geq 1$ for $i \in \{x, y, z\}$. For both two and three dimensional systems, we define $d^i_{k-1} = \max\{\min_{1\leq j\leq n}\{d^i_{j,k-1} : d^i_{j,k-1} > \gamma_i\delta_i\}, 0\}$ as the minimum measured feedback information in the $i^{th}$ direction which can be used by the controller in designing optimal $q^i_{k-1}$. Here $i \in \{x, y, z\}$.

### 7.3.2 Need for Optimality

In this section, we ask questions about the need of an optimal design for suppressing Brownian ensemble. Do we really need an optimal control design for such systems? If not, then can we replace this optimal controller with a classical control design such as a constant gain proportional controller and obtain a reasonable performance of the

system compared to the optimal controller? To answer these questions, we design a proportional controller for one dimensional system as

$$q_k^x = \begin{cases} K_p d_k^x & \text{if} \quad 0 \leq d_k^x \leq q_m^x/K_p, \\ q_m^x & \text{if} \quad d_k^x > q_m^x/K_p. \end{cases} \tag{7.10}$$

Here, $K_p$ is a constant gain. $d_k^x$ is the distance of a particle outside the specified trapping region at the time step $k$ which is closest to the origin among all the particles in the ensemble. $q_m^x \leq 1$ is the maximum limit on the control output $q_k^x$. Now we find $q_k^x$ for which $\mathbb{E}[(d_{k+1}^x)^2 \mid d_k^x] < (d_k^x)^2$ is satisfied. By inserting $q_k^x = K_p d_k^x$ from equation (7.10) in the expression of $\mathbb{E}[(d_{k+1}^x)^2 \mid d_k^x] < (d_k^x)^2$ and using the fact that $(\delta_x^2/N)(1+(N-1)(1-2q_{j,k-1}^x)^2) \leq \delta_x^2$ (see equation (7.5)), we obtain the final control design as

$$q_k^x = \begin{cases} 1/2 & \text{if} \quad d_k^x \leq f(K_p, \delta_x), \\ K_p d_k^x & \text{if} \quad f(K_p, \delta_x) < d_k^x \leq q_m^x/K_p, \\ q_m^x & \text{if} \quad d_k^x > q_m^x/K_p. \end{cases} \tag{7.11}$$

Here, $K_p$ satisfies $0 < K_p < ((4q_m^x - 1)^2 - 1)/(4\delta_x)$ and $f(K_p, \delta_x)$ is given by $f(K_p, \delta_x) = (1 + \sqrt{1 + 4K_p\delta_x})/(4K_p)$.

Equation (7.11) clearly shows that the designed proportional controller is an adaptive gain based controller without any optimality which sets its output at the saturation limits in most of the one dimensional space. As an example, consider $f(K_p, \delta_x) = \delta_x$ with $q_m^x = 1$ in equation (7.11). In this case, it is easy to verify that $K_p = 3/(4\delta_x)$ and $q_k^x = K_p d_k^x$ for $d_k^x \in (\delta_x, 4\delta_x/3]$.

Forcing the controller output to its saturation limits for most of the time clearly

shows the limitation of the designed proportional controller in handling non-linear input constraints such as the supermartingale condition. Also, the saturation of the controller output to its maximum limit in this design induces an unnecessary increase in the total input cost which can certainly be minimized by choosing $q_k^x < q_m^x$, as shown in the previous section. The presence of physical constraints in experimental designs imposed by limitations of the external force fields greatly limit the utility of such a controller design, since proportional controllers inherently cannot handle constraints. Therefore, a natural question arises in designing an optimal control strategy which can overcome these limitations while retaining, at a minimum, similar closed-loop performance. An optimal control design provides a framework to incorporate realistic constraints as well as system dynamics explicitly in the problem formulation. Specifically, a model-based receding horizon control design provides a suitable framework for handling realistic constraints as well as for enhancing the overall performance of the system when the problem is formulated as a trajectory tracking problem. Many problems such as tracking of a predefined trajectory by the average motion of particles while constraining all the particles in a fixed radial distance from the trajectory (a relevant problem in drug delivery) and replacing random walk based models with more realistic Langevin equations in the present context would require the optimal framework. With this motivation, next we develop "Broadcast Stochastic Receding Horizon Control (BSRHC)" framework to trap an ensemble of particles driven by Brownian motion in the minimum trapping region. The framework provides a unified predictive framework to incorporate realistic constraints and system dynamics explicitly in the problem formulation, leading to optimal regulation of Brownian noise-driven dynamical systems.

### 7.3.3 Optimal Control Problem

In this section, we formulate an optimal control problem in the broadcast stochastic receding horizon control (BSRHC) framework, as shown in Figure 7.1, to trap an ensemble of Brownian particles in the minimum possible trapping region.

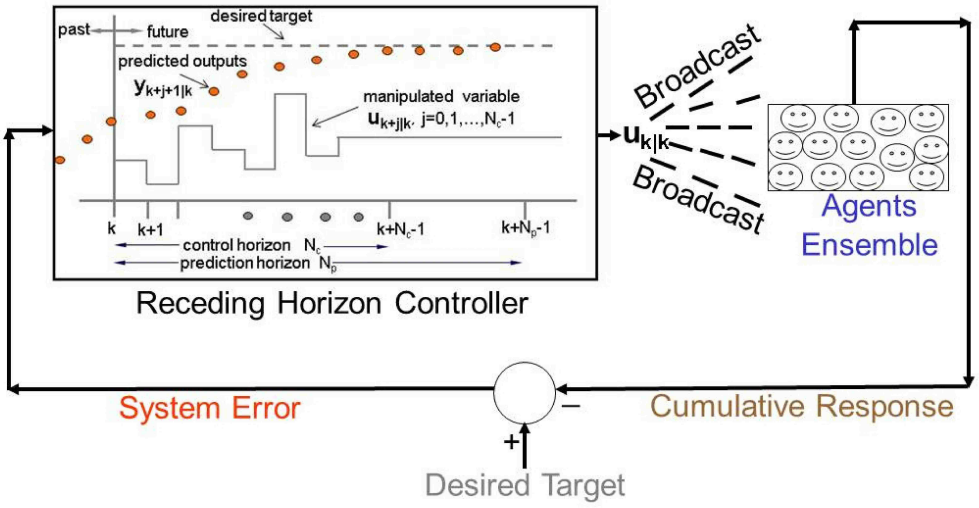



Figure 7.1: Broadcast Stochastic Receding Horizon Control (BSRHC): At the current discrete time $k$, the "Receding Horizon Controller" designs control inputs, "$u_{k|k}$" and "Broadcasts" them to the system ("Particle Ensemble"). Each particle in the ensemble possesses independent Brownian motion. The "Measured Feedback" is the distance of the particle that is closest to the origin. This feedback information is used by the "Receding Horizon Controller" to design new control inputs at the next time $k+1$.

As shown in Figure 7.1, the strategy differs from the one introduced in Chapter 6 in terms of the measured feedback information. In Chapter 6, the ensemble behavior of a swarm of agents was used as the measured feedback information in designing optimal transition probabilities by the controller. Here, the controller uses the distance of a particle outside the trapping region in a given direction which is closest to the origin as the only available feedback information in designing optimal directional probability. To clarify which particle is chosen to obtain feedback information, we consider a

distribution of particles in a two dimensional space as shown in Figure 7.2.

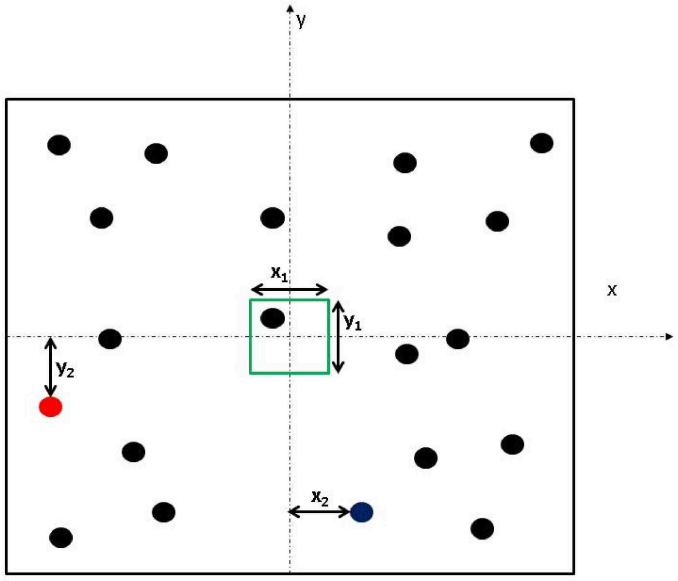



Figure 7.2: Two dimensional distribution of particles undergoing Brownian motion

As shown in Figure 7.2, the green region is the specified trapping region. At a given time $k$, we measure the distance of the particle outside the trapping region which is closest to the $x$-axis and is located at a distance greater than $y_1/2$ among all the particles in the ensemble (red particle located at a distance $y_2 > y_1/2$ from the $x$-axis). This information is then used by the controller as feedback in designing the optimal bias (probability) in $y$-direction towards the origin. Similarly, we measure the distance of the particle outside the trapping region which is closest to the $y$-axis and is located at a distance greater than $x_1/2$ among all the particles in the ensemble (blue particle located at a distance $x_2 > x_1/2$ from the $y$-axis). This information is then used by the controller as feedback in designing the optimal bias (probability) in $x$-direction towards the origin.

In the design of the broadcast stochastic receding horizon control policy for trapping an ensemble of Brownian particles, the objective is to minimize the system error by computing optimal probability with which a particle is moving towards the origin. To achieve this objective, we formulate a constrained non-linear stochastic finite receding horizon control (RHC) problem as follows:

$$\min_{q^i_{k|k},\cdots,q^i_{k+N_c-1|k}} \mathbb{J}^i_k \tag{7.12}$$

s.t.

$$q^i_{k+l|k} \in [0,1] \;\; for \;\; 0 \leq l \leq N_c - 1, \tag{7.13a}$$

$$q^i_{k+l|k} = 0.5 \;\; for \;\; l \geq N_c, \tag{7.13b}$$

$$\mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2 \;\; for \;\; 0 \leq l \leq N_c - 1. \tag{7.13c}$$

Here, $\mathcal{F}^i_k$ is a $\sigma$-algebra generated by the sequence $d^i_1, \cdots, d^i_k$ where $i \in \{x, y, z\}$. $N_p$ and $N_c$ are time invariant prediction and control horizon respectively. The cost function $\mathbb{J}^i_k$ is given by

$$\mathbb{J}^i_k = \sum_{l=0}^{N_p-1} \mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k] + \sum_{l=0}^{N_c-1} \delta_i^2((q^i_{k+l|k})^2 + (1 - q^i_{k+l|k})^2). \tag{7.14}$$

The control problem stated in equation (7.12), (7.13) and (7.14) is solved for each $k \geq 1$ till $d^i_k$ becomes zero for the first time i.e. all the particles are inside the designed trapping region for the first time. At this time, we set the transition probability $q^i_{k|k}$

to 1. This ensures that all the particles are inside the trapping region after they entered in the trapping region for the first time as we show in the next section.

**Remark 7.3.2.** *For practical implementation of the controller, we switched "OFF" the controller once all the particles are inside the trapping region for the first time. This allow all the particles to make unbiased Brownian motion as long as they are inside the designed trapping region. Whenever a particle crosses the boundary of the trapping region, the controller becomes active. The controller remains active until all particles are again brought back inside the trapping region. We discuss this design in detail in the simulation results section.*

## 7.4 Results

### 7.4.1 Feasibility

**Claim 7.4.1.** *The broadcast RHC problem stated in equations (7.12), (7.13) and (7.14) is feasible.*

*Proof.* The Claim 7.4.1 states that the control problem defined by equations (7.12), (7.13) and (7.14) is feasible i.e. there exists $q^i_{k+l|k} \in [0,1]$ for $l = 0, 1, \cdots N_c - 1$ and $i \in \{x, y, z\}$ such that $\mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$ is true. To show this claim, we first consider $l = 1$ and show that there exists $q^i_{k+1|k} \in [0,1]$ such that $\mathbb{E}[(d^i_{k+2})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$ satisfies. For this, we derive the expression for $\mathbb{E}[(d^i_{k+2})^2 \mid \mathcal{F}^i_k]$ using the system dynamics as follows:

$$\begin{aligned}\mathbb{E}[(d^i_{k+2})^2 \mid \mathcal{F}^i_k] = (d^i_k)^2 + 2d^i_k\delta_i(1-2q^i_{k|k}) \\ + 2\delta_i\mathbb{E}[d^i_{k+1} \mid \mathcal{F}^i_k](1-2q^i_{k+1|k}) \\ + \sum_{j=0}^{1}\frac{\delta_i^2}{N}(1+(N-1)(1-2q^i_{k+j|k})^2).\end{aligned} \tag{7.15}$$

Now by noticing the fact that the right hand side of equation (7.15) is always less than or equal to $(d^i_k)^2 + 2d^i_k\delta_i(1-2q^i_{k|k}) + 2\delta_i\mathbb{E}[d^i_{k+1} \mid \mathcal{F}^i_k](1-2q^i_{k+1|k}) + 2\delta_i^2$ and $2d^i_k\delta_i(1-2q^i_{k|k}) + \delta_i^2 < 0$, it would be sufficient to show that there exists $q^i_{k+1|k} \in [0,1]$ such that $2\delta_i\mathbb{E}[d^i_{k+1} \mid \mathcal{F}^i_k](1-2q^i_{k+1|k}) + \delta_i^2 < 0$. Since $\mathbb{E}[d^i_{k+1} \mid \mathcal{F}^i_k] \leq d^i_k + \delta_i$, we obtain

$$q^i_{k+1|k} > \frac{2d^i_k + 3\delta_i}{4(d^i_k + \delta_i)}. \tag{7.16}$$

It should be noted that $d^i_k > 0$ by definition. Using this, it is easy to verify that $\frac{2d^i_k+3\delta_i}{4(d^i_k+\delta_i)} \in (0,1)$. Thus there exists $q^i_{k|k}, q^i_{k+1|k} \in [0,1]$ such that $\mathbb{E}[(d^i_{k+2})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$. Next we assume that this is true for any $l > 1$, i.e. there exists $q^i_{k|k}, q^i_{k+1|k}, \cdots, q^i_{k+l-1|k}$ such that $\mathbb{E}[(d^i_{k+l})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$ for any $l > 1$. Next we show that there exists $q^i_{k|k}, q^i_{k+1|k}, \cdots, q^i_{k+l|k}$ such that $\mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$. By writing the expression for $\mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k]$ as

$$\begin{aligned}\mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k] = \mathbb{E}[(d^i_{k+l})^2 \mid \mathcal{F}^i_k] + 2\delta_i\mathbb{E}[d^i_{k+l+1} \mid \mathcal{F}^i_k] \\ (1-2q^i_{k+l|k}) + \frac{\delta_i^2}{N}(1+(N-1) \\ (1-2q^i_{k+l|k})^2)\end{aligned} \tag{7.17}$$

and using the fact that $\mathbb{E}[(d^i_{k+l})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$, it would be sufficient to show that $2\delta_i\mathbb{E}[d^i_{k+l+1} \mid \mathcal{F}^i_k](1-2q^i_{k+l|k}) + \delta_i^2 < 0$ for some $q^i_{k+l|k} \in [0,1]$. Since $\mathbb{E}[d^i_{k+l+1} \mid \mathcal{F}^i_k] > \delta_i/2$, there exists a $q^i_{k+l|k} \in [0,1]$ such that $\mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$. Thus by

mathematical induction, it follows that there exists $q^i_{k|k}, q^i_{k+1|k}, \cdots, q^i_{k+l|k}$ such that $\mathbb{E}[(d^i_{k+l+1})^2 \mid \mathcal{F}^i_k] < (d^i_k)^2$ for $l = 0, 1, \cdots, N_c - 1$ is true.

□

**Remark 7.4.2.** *Throughout the work, we have neglected the measurement uncertainties in obtaining the feedback information. The proposed framework allows the incorporation of measurement uncertainties as long as the optimization problem (see equations (7.12), (7.13) and (7.14)) is feasible. As an example, consider the case for $N_p = N_c = 1$. The measurement noise can be added to the computed distance in the $i^{th}$ direction as $d^i_{k+1} = |d^i_k + \xi^i_{k+1}| + \mathcal{N}(0, \sigma_{k+1})$. Here, $\mathcal{N}(0, \sigma_{k+1})$ is the Gaussian noise with mean 0 and variance $\sigma^2_{k+1}$. For the feasibility of the optimization problem, one requires $\mathbb{E}[(d^i_{k+1})^2 | \mathcal{F}^i_k] < (d^i_k)^2$. Using (7.5), it is easy to verify that this condition satisfies if $q^i_{k|k} > 1/2 + \delta_i/(4d^i_k) + \sigma^2_{k+1}/(4d^i_k\delta_i)$. Since $q^i_{k|k} \in [0, q^i_m]$, $\sigma_{k+1} < \sqrt{\delta_i((4q^i_k - 2)d^i_k - \delta_i}$.*

Finally, we show that the controller is capable of driving all the particles inside the trapping region with probability 1.

**Claim 7.4.3.** *There exists a finite time in which all the particles are inside the trapping region for the first time under the designed control actions. Moreover, particles are trapped inside the trapping region $H^1$ forever.*

*Proof.* Let us define $\hat{d^i_k} = \max_{j=1,2,\cdots,n} d^i_{j,k} \ \{d^i_{j,k} \geq \gamma_i\delta_i\}$. Here $\{d^i_{j,k} \geq \gamma_i\delta_i\}$ is the indicator function defined as

$$\{d^i_{j,k} \geq \gamma_i\delta_i\} = \begin{cases} 1 & \text{if} \quad d^i_{j,k} \geq \gamma_i\delta_i, \\ 0 & \text{otherwise.} \end{cases} \tag{7.18}$$

Here $\gamma_i \geq 1$. Since the particles are allowed to visit all the space in the $i^{th}$ coordinate and the movement of particles are biased towards the origin, $Pr\{\hat{d}_k^i = 0\} > 0$. Therefore, there will a finite time (say $k_0 < \infty$) when all the particles are inside the designed trapping region for the first time. Once all the particles are inside the trapping region $H^1$, the controller output i.e. $q_{k|k}^i$ is set to 1. Since the maximum distance traveled by a particle in one time step is $\delta_i$ towards the origin and the minimum trapping region with $\gamma_i = 1$ is of total length of $2\delta_i$, all the particles are trapped inside the trapping region for all future time. □

### 7.4.2 Analytical Results

In this section, we derive the closed-form solution of the control problem (equations (7.12), (7.13) and (7.14)) for the one dimensional system. For the one dimensional system with $N_c = N_p = 1$, the optimization problem results in

$$\min_{q_{k|k}^x} \mathbb{J}_k \tag{7.19}$$

s.t.

$$q_{k|k}^x \in [0, 1], \tag{7.20a}$$

$$\mathbb{E}[(d_{k+1}^x)^2 \mid \mathcal{F}_k^x] < (d_k^x)^2. \tag{7.20b}$$

The cost function $\mathbb{J}_k$ is defined as $\mathbb{J}_k^x = \mathbb{E}[(d_{k+1}^x)^2 \mid \mathcal{F}^x] + \delta_x^2((q_{k|k}^x)^2 + (1 - q_{k|k}^x)^2)$. From Claim 7.4.1, we know that the optimization problem (equations (7.19) and (7.20)) is feasible. Next, we find the optimal $q_{k|k}^x$ by solving equations (7.19) and

(7.20) analytically. For this, we define the Lagrangian function [15] $L(q^x_{k|k}, \lambda_1, \lambda_2, \lambda_3)$ as follows:

$$\begin{aligned} L(q^x_{k|k}, \lambda_1, \lambda_2, \lambda_3) = \mathbb{E}[(d^x_{k+1})^2 \mid \mathcal{F}^x_k] + \delta_x^2((q^x_{k|k})^2 + (1 - q^x_{k|k})^2) \\ + \lambda_1(q^x_{k|k} - 1) + \lambda_2(-q^x_{k|k}) + \lambda_3(2d^x_k\delta_x(1 - 2q^x_{k|k}) + \delta_x^2). \end{aligned} \tag{7.21}$$

Here $\lambda_i$, $i = 1, 2, 3$ are non-negative Lagrange multipliers. The first order Karush-Kuhn-Tucker (KKT) conditions [15] for the optimization problem are derived as follows:

$$\frac{\partial L}{\partial q^x_{k|k}} = 0, \tag{7.22a}$$

$$\lambda_1(q^x_{k|k} - 1) = 0, \tag{7.22b}$$

$$\lambda_2(-q^x_{k|k}) = 0, \tag{7.22c}$$

$$\lambda_3(2d^x_k\delta_x(1 - 2q^x_{k|k}) + \delta_x^2) = 0, \tag{7.22d}$$

$$\lambda_i \geq 0 \ \ for\ i = 1, 2, 3. \tag{7.22e}$$

By solving (7.22), we obtain the following closed form solution of the optimization problem: For $d^x_k \in (\sqrt{0.5(3/2 - 1/N)}\delta_x, (3/2 - 1/N)\delta_x)$, $\lambda_1 = \lambda_2 = \lambda_3 = 0$ and $q^x_{k|k} = \frac{1}{2} + \frac{d^x_k}{\delta_x(3-2/N)}$. For $d^x_k \geq (3/2 - 1/N)\delta_x$, $\lambda_1 = \lambda_2 = \lambda_3 = 0$ and $q^x_{k|k} \in (1/2 + \delta_x/(4d^x_k), 1)$. For $d^x_k \geq (3/2 - 1/N)\delta_x$, if $q^x_{k|k} = 1$, $\lambda_1 > 0$, $\lambda_2 = \lambda_3 = 0$. For $d^x_k \in (\delta_x/2, \sqrt{(3/4 - 1/(2N))}\delta_x]$, $\lambda_1 = \lambda_2 = \lambda_3 = 0$ and $q^x_{k|k} \in (1/2 + \delta_x/(4d^x_k), 1)$. For $d^x_k \in (\delta_x/2, \sqrt{(3/4 - 1/(2N))}\delta_x]$, if $q^x_{k|k} = 1$, $\lambda_1 > 0$, $\lambda_2 = \lambda_3 = 0$.

### 7.4.3 Simulation Results

In this section, we use simulations to demonstrate the efficacy of our control design in trapping an ensemble of Brownian particles in a predefined region. We simulate trajectories of 100 independently behaving Brownian particles in one, two and three dimensional homogeneous medium with the implemented "BSRHC" strategy. For simulation, the prediction and control horizon are set to 1 in designing the receding horizon controller i.e. $N_p = N_c = 1$. A constraint on the control input is imposed by setting its upper bound at 0.9.

Figure 7.3 shows a realization of uncontrolled and controlled trajectories of particles in one dimensional space with the implemented control architecture. The "Uncontrolled" region of Figure 7.3 shows the trajectories of particles undergoing independent unbiased Brownian motion in one dimension in the absence of the controller. We simulated these trajectories up to time $k = 500000$ which allowed particles in the ensemble to acquire random positions in one dimensional space prior to implementation of our designed controller. Next, we switch the controller "ON" at the next time step $k = 500001$. The trajectories of the particles are now controlled according to the designed "BSRHC" strategy and shown in the "Controlled" region of Figure 7.3. The designed controller drives all the particles into the trapping region for the first time within 100 time steps.

The basic principles of the designed controller in driving all the particles inside the trapping region can be explained as follows: At the time step $k = 500001$, the controller designs an optimal control input using the measured distance of a particle outside the trapping region that is closest to the origin at time $k = 500000$, as the only available feedback information. The controller then broadcasts this designed

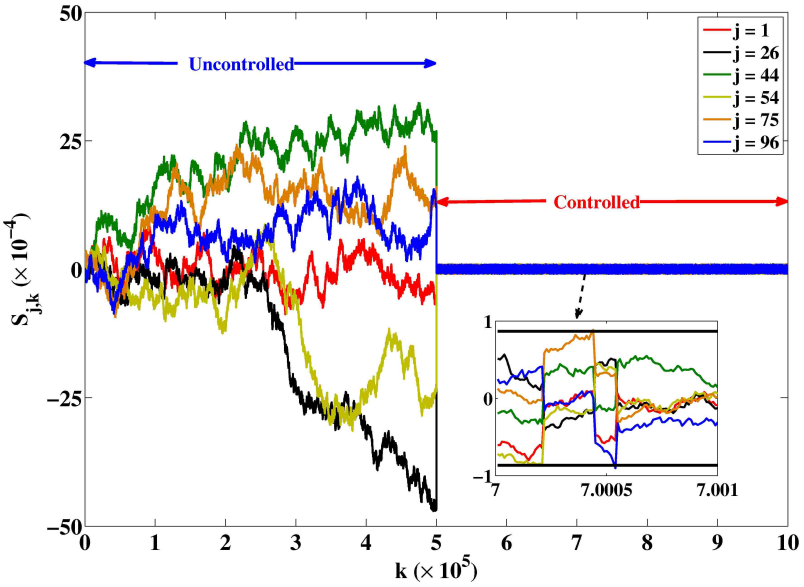



Figure 7.3: Trapping of 100 particles in one dimensional space: The horizontal axis represents discrete time steps. The vertical axis shows the position of particles, measured from the origin on a scale of $10^{-4}$, as a function of discrete time steps. $S_{j,k}$ is the position of the $j^{th}$ particle at the time step $k$. For clarity, trajectories of 6 particles are shown here. The "Uncontrolled" region shows the unbiased independent Brownian trajectories of particles till the time step $k = 500000$. The "Controlled" region shows the trajectories of particles under the designed control action. Within 100 time steps, the designed control action brings all particles into the trapping region which is defined as the region inside the interval $[-0.87\delta_x, 0.87\delta_x]$. Here, $\delta_x = 10^{-4}$. The inset plot shows the trajectories of particles inside the trapping region between the time step $k = 700001$ and $k = 700100$.

control input to all the particles and thus drives the particles towards the trapping region. This process continues until the closest particle enters the trapping region. Subsequently, the controller designs the control input using the measured distance of the next nearest particle outside the trapping region that is closest to the origin and broadcasts this control input to all the particles to drive the rest of the particles towards the trapping region. This process of designing the control input is repeated at each time step until all the particles are inside the trapping region, at which point the controller is switched "OFF". The particles now undergo unbiased Brownian motion as long as they are inside the trapping region. Whenever a particle crosses the boundary of the trapping region, the controller becomes active. The controller remains active until all particles are again brought back inside the trapping region. Thus the controller switches between "ON" and "OFF" for all the time steps once all the particles are inside the trapping region for the first time. The reason for this switching strategy has been provided in the theoretical section where we have shown that the designed controller cannot enhance the overall performance of the system further once all the particles are inside the trapping region.

The "inset plot" of Figure 7.3 shows the trajectories of 6 particles for 100 time steps, starting from $k = 700001$. Particles 54, 75 and 96 cross the trapping region boundary at three distinct time steps. At all these time steps, the controller becomes active and drives the particle into the trapping region within a few time steps. Once the particle is inside the trapping region, the controller becomes inactive and all the particles in the ensemble follow unbiased independent Brownian trajectories. This demonstrates that the designed controller is effective in keeping all particles inside the trapping region for a long period of time. Figure 7.4 shows the implemented

control action in trapping 100 particles in one dimensional space.

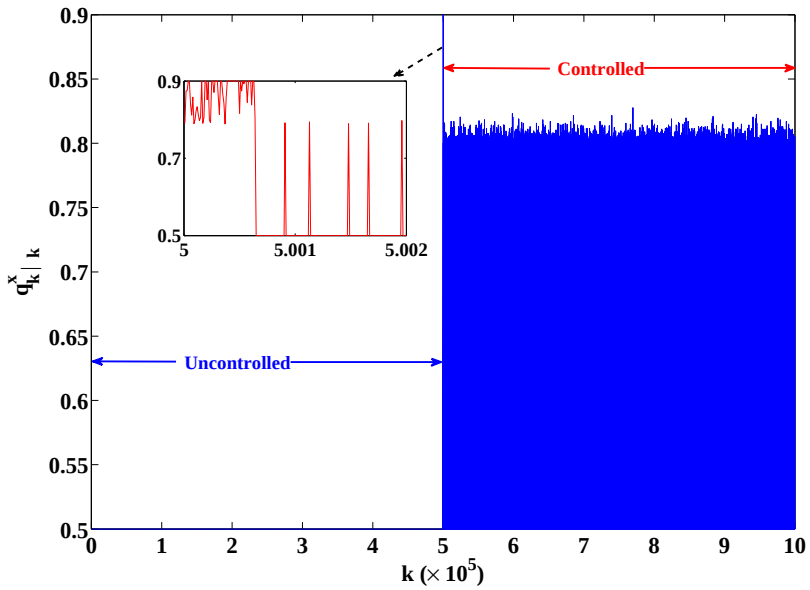



Figure 7.4: Optimally designed control inputs for trapping 100 particles in one dimensional space: The horizontal axis represents discrete time steps. The vertical axis shows the implemented control action designed using "BSRHC" strategy.

Next, we show the trapping of an ensemble of particles in two and three dimensional homogeneous medium by extending the design of the one dimensional controller to two and three dimensional systems. For the two dimensional system, two independent control inputs are designed to control the trajectories of particles in the $x-y$ plane. For the three dimensional system, three independent control inputs are designed to control the trajectories of particles in the $x-y-z$ space. We note that the number of designed control inputs depends only on the dimensionality of the system. Figures 7.5 and 7.6 show the snapshots of the positions of 100 particles at 8 distinct time steps in two and three dimensional space respectively.

As shown in Figures 7.5 and 7.6, the top 3 plots demonstrate the uncontrolled

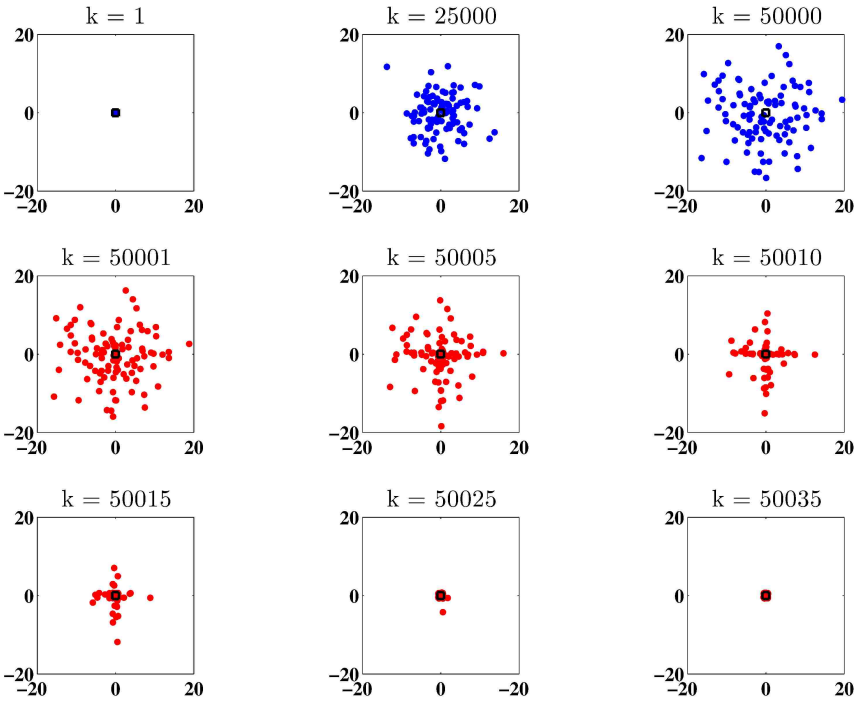



Figure 7.5: Trapping of 100 particles in two dimensional space: In each plot, the horizontal and vertical axes respectively represent the $x$ and $y$ co-ordinates of particles on a scale of $10^{-4}$. The top left plot shows that all particles are placed at the origin at time step $k = 1$. The next two plots show the diffusion of particles in two dimensional space by following the unbiased independent Brownian trajectories. The controller is "ON" at time step $k = 50001$ which is shown in the middle left plot. The next five plots show the efficacy of the controller in driving all particles into the trapping region within 35 time steps.

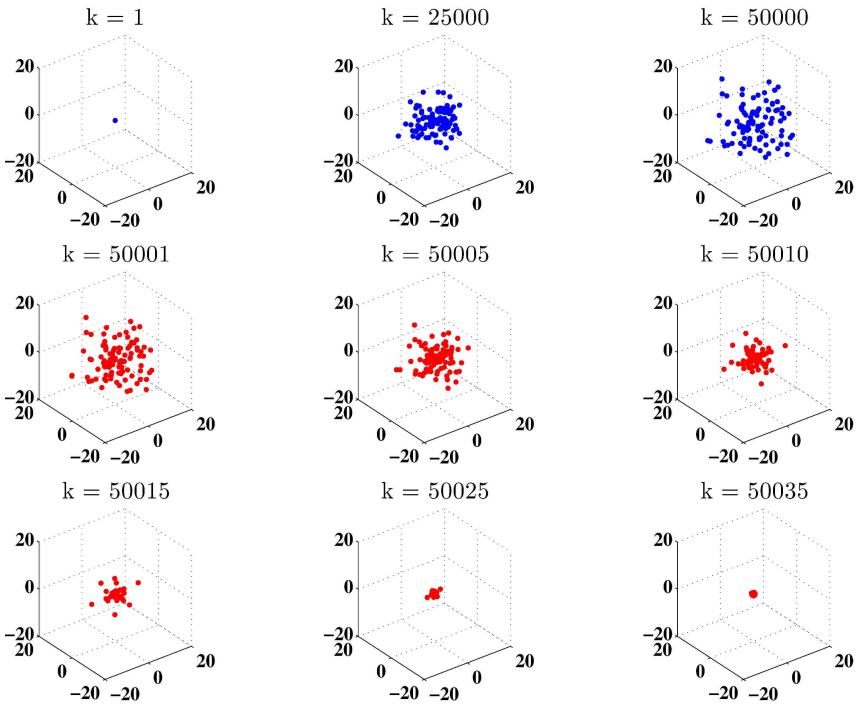



Figure 7.6: Trapping of 100 particles in three dimensional space: In each plot, the horizontal plane represents the $x-y$ co-ordinates and the vertical axis represents the $z$-co-ordinate of particles on a scale of $10^{-4}$. The top left plot shows that all particles are placed at the origin at the time step $k = 1$. The next two plots show the diffusion of particles in three dimensional space by following unbiased independent Brownian trajectories. The controller is "ON" at time step $k = 50001$ which is shown in the middle left plot. The next five plots show the efficacy of the controller in driving all particles into the trapping region within 35 time steps.

diffusion of 100 particles till the time step $k = 50000$ in two and three dimensional space respectively. The next 6 plots in both figures illustrate the efficacy of the designed controller in driving all the particles into the defined trapping region within 35 time steps. The principles on which the controller works in driving all the particles into the trapping region in two and three dimensional space are the same as those described in the one dimensional case. Figures 7.7 and 7.8 demonstrate that the controller is effective in trapping all the particles inside the trapping region for an extended period of time, beyond the initial trapping time.

Figures 7.9 and 7.10 show the implemented control action in trapping 100 particles in two dimensional space.

## 7.5 Concluding Remarks

Advances in experimental techniques for manipulating micro and nano scale particles in the last decade have stimulated considerable interest in regulating Brownian ensemble dynamics. In this chapter, we have proposed a novel "Broadcast Stochastic Receding Horizon Control (BSRHC)" strategy as a unified optimal feedback control framework for regulating small length scale Brownian dynamical systems. The framework assumes the existence of physical / biological mechanisms which can manipulate the transition probabilities of particles, undergoing predominantly Brownian motion, in a broadcast fashion. Additionally, realistic models describing the effect of experimentally feasible external force fields on the transition probabilities can be readily incorporated in this framework. Using the measurement of the position of a single particle in the ensemble, the technique eliminates the traditionally required tracking of individual particles within the ensemble in regulating their dynamical behaviors.

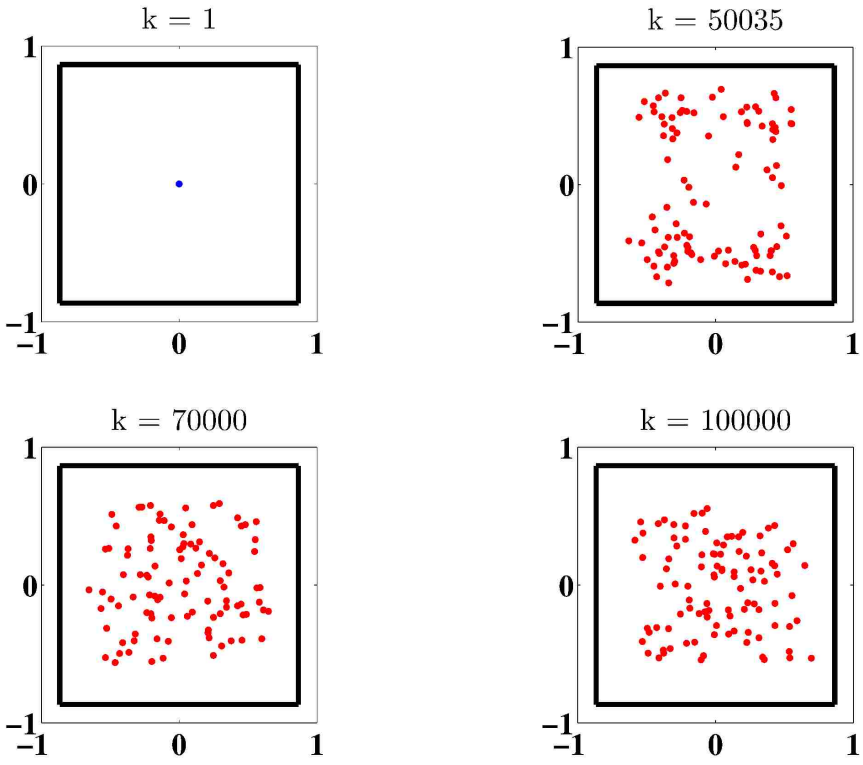



Figure 7.7: The performance of the controller in two dimensional space over an extended period of time. The $x$ and $y$ co-ordinates of particles are on a scale of $10^{-4}$. The top left plot shows the placement of particles at the origin at time step $k = 1$. The trapping region is shown by a square with coordinates $(\pm\alpha, \pm\alpha)$. Here, $\alpha = 0.87\delta_x$ and $\delta_x = 10^{-4}$ inside the plot. The next three plots show the efficacy of the controller in trapping all particles inside the square over an extended period of time.

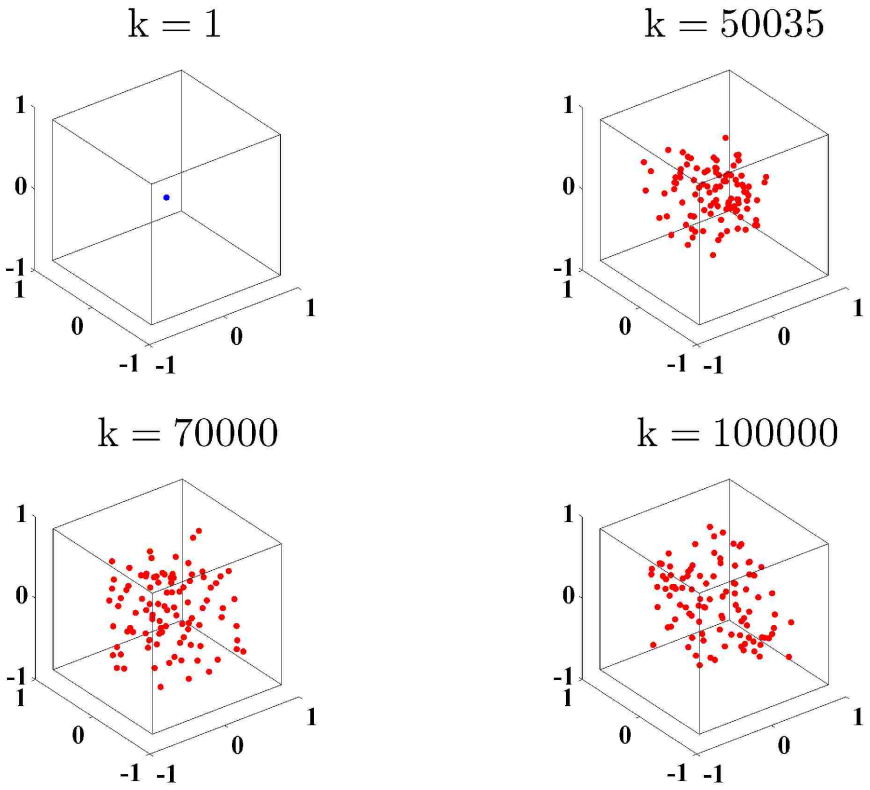



Figure 7.8: The performance of the controller in three dimensional space over an extended period of time. The $x - y - z$ co-ordinates are shown on a scale of $10^{-4}$. The top left plot shows the placement of particles at the origin at the time step $k = 1$. The trapping region is shown by a cube with coordinates $(\pm\alpha, \pm\alpha, \pm\alpha)$. Here, $\alpha = 0.87\delta_x$ and $\delta_x = 10^{-4}$. The next three plots show the efficacy of the controller in trapping all particles inside the cube over an extended period of time.

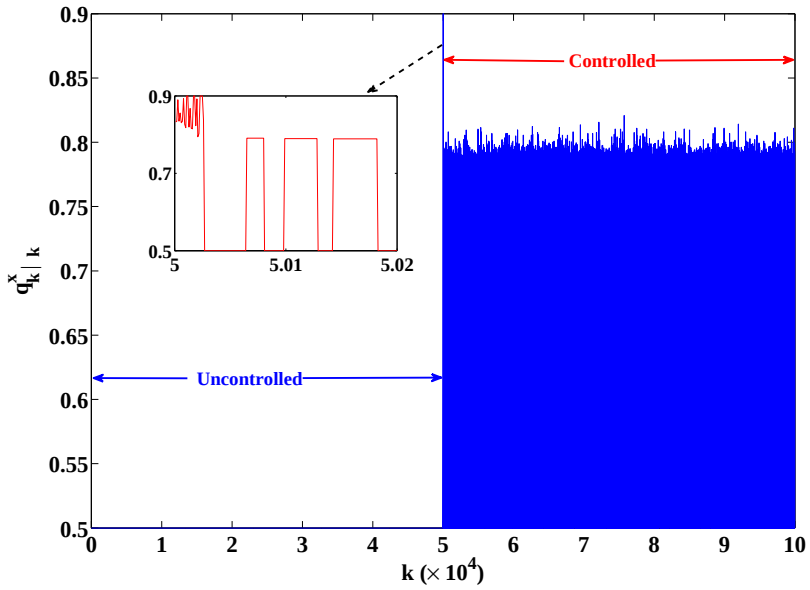



Figure 7.9: Optimally designed control inputs in the $x$-direction for trapping 100 particles in two dimensional space: The horizontal axis represents discrete time steps. The vertical axis shows the implemented control action in the $x$-direction designed using "BSRHC" strategy.

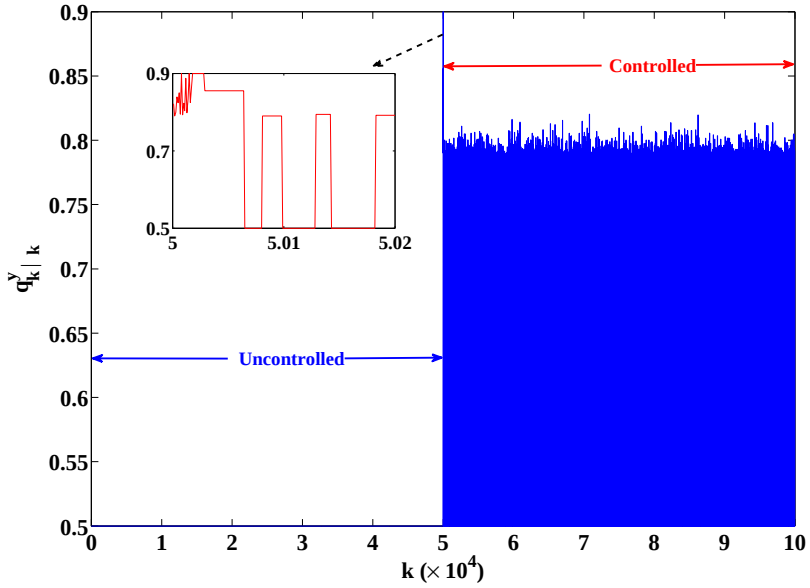



Figure 7.10: Optimally designed control inputs in the $y$-direction for trapping 100 particles in two dimensional space: The horizontal axis represents discrete time steps. The vertical axis shows the implemented control action in the $y$-direction designed using "BSRHC" strategy.

Similarly, by broadcasting a common optimal transition probability to the ensemble, the technique eliminates the need to have distributed actuators. The presented feasibility and convergence results of the controller guarantee the trapping of Brownian ensemble independent of the choice of external force fields typically used to manipulate transition probabilities of particles in experimental designs. Our results assume a uniform force field acting on all the particles in the ensemble. However, the generality of the control problem formulation allows incorporation of spatially dependent transition probability models for non-uniform force fields in the "BSRHC" framework. As long as a certain conditional expectation condition is satisfied, the results readily apply to these non-uniform field situations.

# Chapter 8

# Summary and Directions for Future Work

## 8.1 Summary

The approach of stochastic modeling and control of dynamical systems has recently become essential for facilitating many emerging applications in biology and physical sciences. In this dissertation, our focus was brain-machine interfaces (BMIs) and emerging applications in small length scale dynamical systems. In Chapters $2-5$, our main focus was to develop a generalized control-theoretic framework using a model-based optimal receding horizon controller to facilitate rigorous analysis of closed-loop BMIs under various feedback scenarios. In this direction, in Chapter 2, we investigated the quantitative predictive capability of the Izhikevich single neuron model in explaining experimental data. We used a part of experimental data to first estimate the unknown parameters of the model and then analyzed the capability of the model in predicting the remaining data. In particular, we explored two approaches namely,

the coincidence factor approach and the Fokker-Planck based maximum likelihood approach, for estimating parameters of both the deterministic and the stochastic form of the model using a part of experimental inter-spike interval (ISI) data. Our results showed that both forms of the model can explain approximately 38% of the experimental data using the parameters estimated by the approach of maximizing the average coincidence factor. Moreover, the stochastic Izhikevich model with estimated parameters using the maximum likelihood approach showed poor predictive capability in explaining experimental data in comparison to the approach of maximizing the average coincidence factor. In conclusion, the Izhikevich single neuron model showed poor predictive capability in explaining experimental data in comparison to other similar computationally efficient single neuron models.

In Chapter 3, we studied a recurrent network of synaptically connected spiking neurons and established conditions under which inter-spike intervals (ISIs) of individual neurons in the network are continuously differentiable with respect to (w.r.t.) parameters (decision variables) of an external stimulating input current which drives the network. The dynamical behavior of individual neurons was represented by a class of discontinuous single neuron models. We found that ISIs of neurons in the network are continuously differentiable w.r.t. decision variables if (1) continuously differentiable trajectory of the membrane potential exists between consecutive action potentials w.r.t. time and decision variables, and (2) the partial derivative of the membrane potential of spiking neurons w.r.t. time is not equal to the partial derivative of their firing threshold w.r.t. time at the time of action potentials.

In Chapter 4, we developed a generalized optimal control framework using a

model-based receding horizon control policy for a rigorous control-theoretic analysis of closed-loop neuroprosthetic systems under various feedback scenarios. Using this framework, we formulated a minimum time control problem for accomplishing reaching tasks in a minimum time and demonstrated the capability of the optimal controller in designing higher level motor planning. In particular, we studied the importance of sensory feedback pathways in a single joint prosthetic arm movement controlled by the firing activity of a single cortical motor neuron. From our results, we concluded that visual feedback is important in rejecting internal noises, naturally occurs in neuronal network, while reaching the desired goal of the task.

Finally in Chapter 5, we designed optimal artificial sensory feedback in an optimal control-theoretic framework to recover the closed-loop performance of a brain-machine interface (BMI) during voluntary single joint extension task. Using a psychophysiological cortical circuit model for the voluntary control of a single joint movement, we developed a BMI and analyzed its performance in the presence and the absence of the natural proprioceptive feedback pathways. We explicitly showed that the performance of the BMI degrades substantially in the absence of natural proprioception. Two control problems namely, the position trajectory tracking problem and the cortical sensory neurons average firing rate tracking problem, were investigated towards designing an optimal artificial sensory feedback for the BMI in the receding horizon control framework. From our results, we concluded that tracking the natural firing activity of the cortical sensory neurons using an external stimulating controller is the appropriate approach towards recovering the natural performance of the motor task.

In Chapters $6-7$, our focus was to develop stochastic optimal control strategies

in a receding horizon framework for facilitating emerging applications in small length scale dynamical systems. In this direction, we proposed a novel "Broadcast Stochastic Receding Horizon Control (BSRHC)" strategy in Chapter 6 for stabilizing the aggregate system behavior of a class of multi-agent systems. In particular, we studied a stabilization problem in a multi-state multi-agent system consisted of a vast number of stochastically behaving agents. Conditional expectation based time-horizon predictive dynamical models were developed to represent the time evolution of the collective behavior of agents. Probabilistic tools such as the supermartingale theory and the bounded convergence theorem were applied to guarantee the almost sure convergence of the closed-loop system behavior to the desired one. The derived stability and convergence results established key principles applicable to stabilize general stochastic dynamical systems.

In Chapter 7, we used the BSRHC strategy developed in Chapter 6 with relevant modifications to show the optimal trapping of an ensemble of particles driven by Brownian motion in the smallest trapping region in one, two and three dimensional homogeneous medium. Using the measurement of the position of a single particle in the ensemble, the developed technique eliminated the traditionally required tracking of individual particles within the ensemble in regulating their dynamical behaviors. Similarly, by broadcasting a common optimal transition probability to the ensemble, the technique eliminated the need to have distributed actuators. Our derived stability and convergence results guaranteed the trapping of Brownian ensemble independent of the choice of external force fields typically used to manipulate transition probabilities of particles in experimental designs.

## 8.2 Future Work

### 8.2.1 Artificial Feedback Design in BMIs using Currents in a Biphasic Waveform

In Chapter 5, we designed optimal artificial sensory feedback in a control-theoretic framework to recover the closed-loop performance of a BMI during voluntary single joint extension task. In this study, an average firing activity based cortical circuit model (see Figure 5.1 in Chapter 5) was used by the receding horizon controller in designing the artificial sensory feedback to stimulate the area 5 "PPV" neurons. Since the stimulating input to individual neurons in this model was based on average firing activity, the designed stimulating input by the controller in the form of an average firing activity was appropriate for the system considered in this chapter.

However in experimental BMIs studies, currents in a charge balanced biphasic waveform such as shown in Figure 8.1 are typically used to stimulate cortical sensory neurons externally and thus to provide artificial sensory feedback during closed-loop operation of BMIs [38, 168, 63]. One possible way to allow the use of such form of currents in the present firing activity based framework is to modify the cortical circuit model (See Figure 5.1) as shown in Figure 8.2.

As shown in Figure 8.2, we have included a network of spiking neurons which are driven by the firing activity of the primary spindle fibers "Ia". Each neuron in the network receives input in the form of both the external currents and synaptic currents. For simplicity, we have not included inputs to "SFV" neurons and "IFV" neurons from the secondary spindle fibers "II". As we will show later in this section, this modification in the cortical circuit model will allow us to design optimal artificial

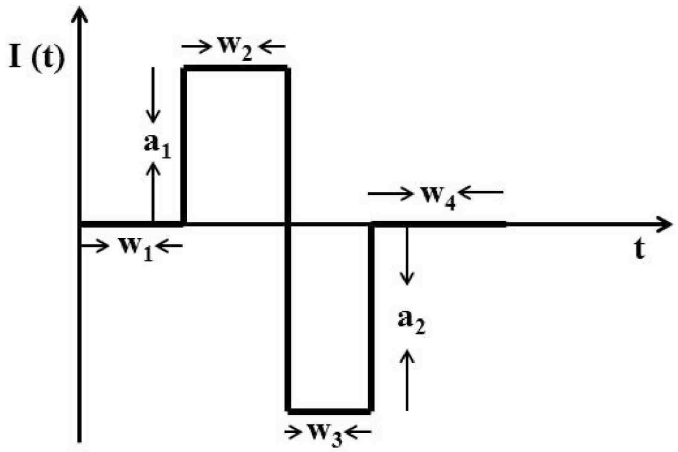



Figure 8.1: A sketch of a typical charge balanced intra-cortical micro-stimulation (ICMS) current in a biphasic waveform. Here the net current $\int_0^T I(t)\mathrm{d}t$ is zero for $T = \sum_{i=1}^{4} w_i$.

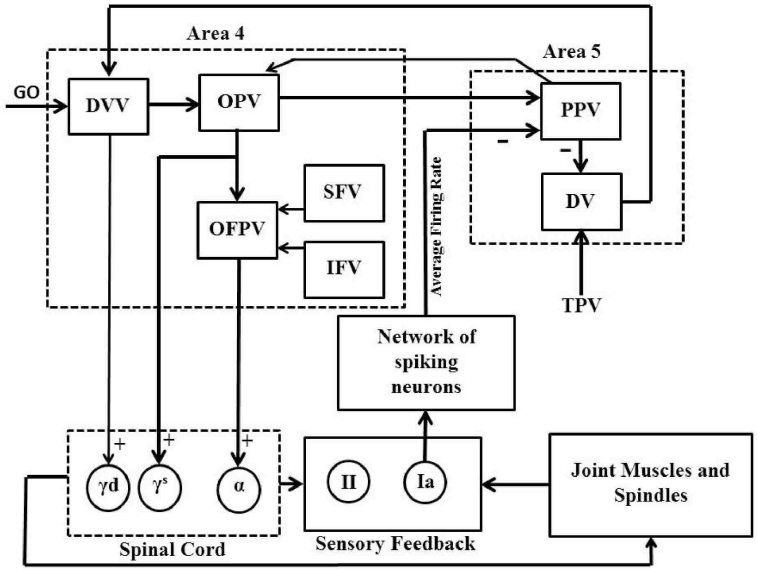



Figure 8.2: An extension of the psychophysiological cortical circuit model for voluntary control of single joint movement shown in Figure 5.1. Here, we have included a network of spiking neurons which are driven by the firing activity of the primary spindle fibers "Ia".

sensory feedback by constraining the stimulating input current to bi-phasic waveforms in closed-loop BMIs. Moreover, this extension will provide a generalized stimulation based closed-loop BMI framework which can be implemented on real systems.

The dynamical behavior of synaptically connected neurons in the network of spiking neurons can be given either by the network model described in section 3.3 or by the following population based firing rate model for the Integrate-and-Fire neurons [124, 56]:

$$\frac{\partial P(t,v)}{\partial t} = -\frac{\partial J(t,v)}{\partial v}, \tag{8.1a}$$

Initial Condition:

$$P(t,v) \mid_{t=0} = \delta(v-0), \tag{8.1b}$$

Boundary Conditions:

$$P(t,v) \mid_{v=1} = 0, \tag{8.1c}$$

$$J(t,v) \mid_{v=0} = J(t,v) \mid_{v=1} . \tag{8.1d}$$

Here, $P(t,v)$ is the membrane potential density defined by [56]

$$\lim_{N\to\infty}(\frac{\text{neurons with } v_0 < v_i(t) \leq v_0+\Delta v}{N}) = \int_{v_0}^{v_0+\Delta v} P(t,v)\mathrm{d}v. \tag{8.2}$$

$v_i(t)$ is the membrane potential of the $i^{th}$ neuron at time $t$ whose dynamics is given by the following Integrate-and-Fire single neuron model:

$$\tau_m \frac{\mathrm{d}v_i(t)}{\mathrm{d}t} = -v_i(t) + RI_i(t). \tag{8.3}$$

Here $\tau_m$ is the membrane time constant, $R$ is the input resistance, and $I_i(t)$ is the total input current to the neuron (the external input current $I^E(t)$ and the synaptic input). At $v_i(t) = 1$, the membrane potential is reset to $v_i(t) = 0$. $P(v,t)$ satisfies $\int_{-\infty}^{1} P(t,v)\mathrm{d}v = 1$ for all $t \geq 0$. $J(t,v)$ is the probability flux defined by

$$J(t,v) = \frac{1}{\tau_m}[-v + RI^E(t)]P(t,v) + \sum_k \nu_k(t) \int_{v-w_k}^{v} P(t,v')\mathrm{d}v'. \tag{8.4}$$

Here $w_k$ is the jump in the membrane potential caused by an input spike at a synapse of type $k$. $\nu_k(t)$ is the effective spike arrival rate. The obvious benefit of using this firing rate model over the network model described in section 3.3 is the small number of unknown parameters present in this model which can easily be trained to match the output of the model i.e. firing rate with the activity of "Ia" fiber. With this modification in the cortical circuit model, Figure 8.3 and Figure 8.4 show designs of biphasic stimulation based closed-loop brain-machine interfaces (BMIs) similar to the one shown in Figure (5.12) and Figure (5.13) respectively.

For the systems shown in Figure 8.3 and Figure 8.4, the following control problem can be formulated in the receding horizon control framework to design optimal artificial feedback currents in biphasic bipolar waveform:

$$\min_{\substack{a_1(k+l|k),a_2(k+l|k),w_1(k+l|k),w_2(k+l|k),w_4(k+l|k),n_t(k+l|k) \\ l=0,1,\cdots,N_c-1}} \mathbb{J}_p(k) \tag{8.5a}$$

such that

$$\begin{aligned} T_s = n_t(k+l \mid k)[w_1(k+l \mid k) + w_2(k+l \mid k)(1 + a_1(k+l \mid k)/a_2(k+l \mid k)) \\ + w_4(k+l \mid k)] \ \ for \ \ 0 \leq l \leq N_c - 1, \end{aligned} \tag{8.5b}$$

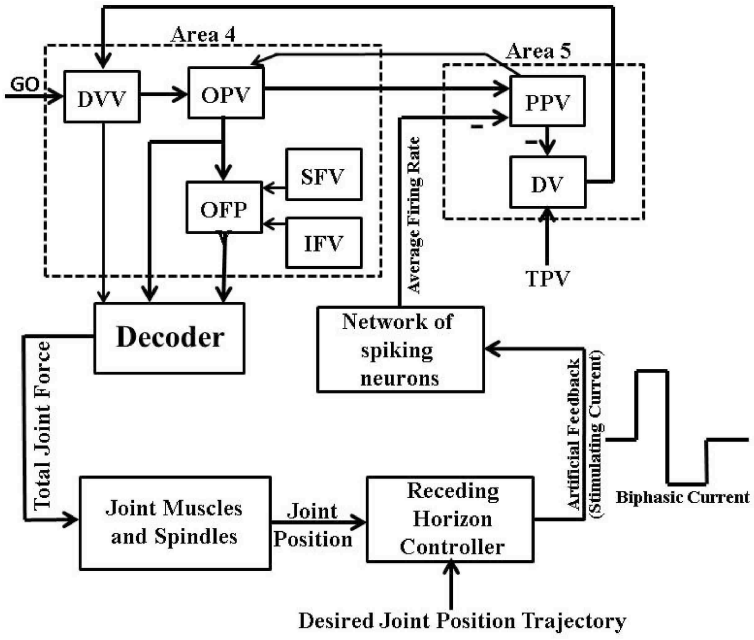



Figure 8.3: Receding horizon controller based closed-loop BMI design I: Here the receding horizon controller designs the "Artificial Feedback" stimulating current in a charge balanced bi-phasic waveform to stimulate "Network of Spiking Neurons" such that the system output ("Single Joint Position" trajectory) mimics the "Desired Joint Position" trajectory.

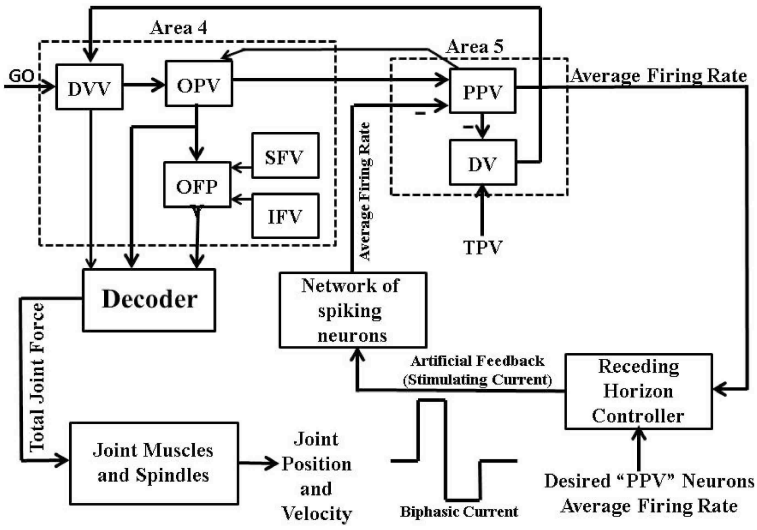



Figure 8.4: Receding horizon controller based closed-loop BMI design II: Here the receding horizon controller designs the "Artificial Feedback" stimulating current in a charge balanced bi-phasic waveform to stimulate "Network of Spiking Neurons" such that the system output ("PPV Neurons Average Firing Rate") mimics the "Desired PPV Neurons Average Firing Rate".

$$a_1(k+l \mid k) \in [0, A_{max}] \ \ for \ \ 0 \leq l \leq N_c - 1, \tag{8.5c}$$

$$a_2(k+l \mid k) \in [A_{min}, 0] \ \ for \ \ 0 \leq l \leq N_c - 1, \tag{8.5d}$$

$$w_i(k+l \mid k) \geq 0 \ \ for \ \ 0 \leq l \leq N_c - 1, \ \ i \in \{1, 2, 4\}, \tag{8.5e}$$

$$n_t(k+l \mid k) \in \{1, 2, 3, \cdots\} \ \ for \ \ 0 \leq l \leq N_c - 1. \tag{8.5f}$$

Here, $\mathbb{J}_p(k) = \sum_{m=0}^{N_p-1}(O(k+m+1 \mid k) - R(k+m+1 \mid k))^2$ is the cost function. $N_c$ and $N_p$ are the control and the prediction horizon respectively. $a_1(k+l \mid k)$, $a_2(k+l \mid k)$, $w_1(k+l \mid k)$, $w_2(k+l \mid k)$, and $w_4(k+l \mid k)$ for $l = 0, 1, \cdots, N_c - 1$ characterize a single biphasic pulse of current $I(t)$ shown in Figure 8.1. $w_3(k+l \mid k) = w_2(k+l \mid k)(a_1(k+l \mid k)/a_2(k+l \mid k))$ for $l = 0, 1, \cdots, N_c - 1$ is given by

the total charge balance in a single biphasic pulse of current $I(t)$. $T_s$ is the decoder sample time. $n_t(k+l \mid k)$ for $l = 0, 1, \cdots, N_c - 1$ is the number of biphasic pulse in the sample time $T_s$.

In Figure 8.3, the measured output of the system "Joint Position" $O(k \mid k)$ at a given time $k$ is the position of the agonist muscle $i$ i.e. $p_i(k \mid k)$. $R(\cdot)$ represents the desired position trajectory. In Figure 8.4, the measured output of the system "Average Firing Rate" $O(k \mid k)$ at a given time $k$ is the average firing activity of the area 5 "PPV" neurons associated with the agonist muscle $i$ i.e. $x_i(k \mid k)$. $R(\cdot)$ represents the desired average firing activity of the area 5 "PPV" neurons. The control problem defined by equation (8.5) can be solved using the approach described in Chapter 5.

### 8.2.2 Continuous Differentiability of First Passage Time in Stochastic Spiking Neuron Models and Networks

In Chapter 3, we derived conditions to ensure continuous differentiability of inter-spike intervals (ISIs) with respect to (w.r.t.) decision variables in a network of non-chaotic deterministic spiking neuron models. It is well known that single neurons are intrinsically noisy [45, 57]. Typically, this noisy characteristic of single neurons is captured in deterministic dynamical models of spiking neurons by including an additive or multiplicative noise term in form of an external input current to the model. In this case, the dynamical model takes the form of stochastic differential equations and the resultant sequence of ISIs (also known as the first passage times) becomes a stochastic process [135, 159, 160, 150] (also see Chapter 1). From optimization point of view, relevant questions here would be to find conditions under which the expectation or the

conditional expectation of first passage time is continuously differentiable w.r.t. to decision variables. Deriving such conditions would not only allow more rigorous analysis of closed-loop neuroprosthetic systems under various feedback scenarios studied in Chapter 4 by formulating stochastic control problems but would also facilitate the development of realistic learning algorithms to train a network of spiking neurons.

### 8.2.3 Minimum Time Control Problem in Closed-Loop Neuroprostheses: An Extension

In Chapter 4, we proposed a generalized control-theoretic framework using optimal receding horizon control theory for facilitating rigorous analysis of closed-loop neuroprostheses under various sensory feedback scenarios. Using this framework, we formulated a minimum time control problem to drive a single joint prosthetic arm using the activity of a single cortical motor neuron. It is well known in neuroscience that various functional areas of the brain contribute in controlling even a single joint movement tasks. As an example, we have shown in Chapter 5 that a minimal cortical circuit for single joint movement tasks involves various types of neurons even from a single cortical functional area (see Figure 5.1). One possible way to make our study in Chapter 4 more realistic is to develop a functionally connected network of cortical neurons such as shown in Figure 8.5 and use the spike train of the primary motor cortex neurons to drive a single joint prosthetic arm.

As shown in Figure 8.1, the neurons of PFC area, believed to be involved in the long term planning of movement, are stimulated using input currents designed by the external controller. The evoked firing activity of neurons in the PFC area eventually stimulate the network by transferring information through synaptic connectivities in

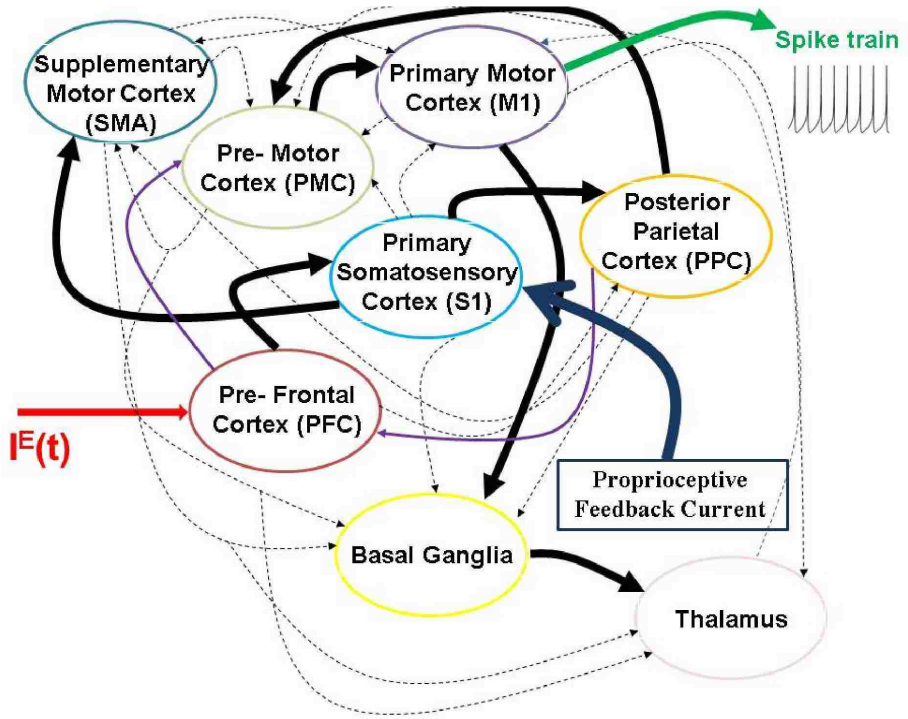



Figure 8.5: A simplified function cortical network model of spiking neurons involved in motor tasks. Here solid lines are showing strong connectivity and dashed lines are representing weak connectivity among cortical areas. Connectivity among various areas, shown here, are based on the existing experimental evidences [87].

the network. In terms of input-output model, the manipulating input to the network is the external current $I^E(t)$ and the measured output is the spike train of M1 neurons.

At the simplest level, each cortical area in the functional cortical circuit model shown in Figure 8.1 can be represented by a single neuron whose dynamics can be given by equations (1.3), (1.4), (1.5) and (1.7). The dynamical decoder dynamics, the single joint prosthetic arm dynamics and the proprioceptive feedback current can be given by equations (4.3), (4.4) and (4.5) respectively. The inter-spike intervals (ISIs) of the primary motor cortex (M1) neuron can be used by the decoder to drive the single joint prosthetic arm. With this, the minimum time control problem in the receding horizon framework can be defined by equation (4.6). To solve the minimum time control problem numerically, one would require at least first order continuous differentiability of ISIs of the primary motor cortex neuron w.r.t. $I^E_{k+j|k}$ for $j = 0, 1, \cdots, N_c(k)-1$ and $k \geq 1$ (see equation (4.6) in Chapter 4 for detailed description).

In Chapter 3, we derived sufficient conditions to ensure continuous differentiability of ISIs w.r.t. decision variables (in this case $I^E_{k+j|k}$) in a network of synaptically connected spiking neurons. By applying these results on the closed-loop neuroprosthetic system model defined by equations (1.3), (1.4), (1.5), (1.7), (4.3), (4.4) and (4.5), it is easy to verify that the ISIs of the primary motor cortex neuron are not continuously differentiable w.r.t. $I^E_{k+j|k}$ for $j = 0, 1, \cdots, N_c(k) - 1$ and $k \geq 1$. One way to ensure continuously differentiable of these ISIs w.r.t. $I^E_{k+j|k}$ is to replace $K(t - t^f_j)$ model given by equation (1.7) by

$$K(t - t^f_j) = \frac{q_j}{\tau_s}(t - t^f_j)^2 \exp(-(t - t^f_j)/\tau_s)\Theta(t - t^f_j), \tag{8.6}$$

and the decoder (joint torque) model given by equation (4.4) by

$$\tau(t) = \tau(t_k) + (2\alpha/\beta)(t - t_k)^2 \exp(-(t - t_k)/\beta) \;\; for \;\; t \in (t_k, t_{k+1}] \qquad (8.7)$$

with $\tau(t_1) = \tau(0) = 0$. Here, $\alpha$ and $\beta$ are fixed parameters. These modifications in the system model ensure that the inter-spike intervals (ISIs) of neurons in the functional network are continuously differentiable w.r.t. $I^E_{k+j|k}$ for $j = 0, 1, \cdots, N_c(k)-1$ and $k \geq 1$. With the new system model, the performance of the closed-loop neuroprosthetic system can be analyzed rigorously under various feedback scenarios by solving the minimum time control problem using a local gradient based optimization algorithm. Further extension of the neuroprosthetic system model by including more than one neurons in individual cortical areas, a dynamical model of cerebellum, and multi-joint prosthetic arm model would allow one to investigate principles of motor coordination during complex reaching and grasping tasks and to study the effect of learning on the closed-loop performance of neuroprosthetic systems.

### 8.2.4 Stochastic Receding Horizon Control of Constrained Non-linear Stochastic Dynamical Systems

In Chapter 6, we proposed an optimal broadcast stochastic receding horizon control strategy to stabilize the aggregate system behavior in a vast number of stochastically behaving agents. In Chapter 7, we extended this strategy to trap an ensemble of particles driven by Brownian motion optimally for an extended period of time. In both systems, we have ignored interactions among entities and used discrete time simplified models to represent the stochastic dynamical behavior of a single entity.

These simplified models with appropriate assumptions allowed us to develop the basic framework of the optimal broadcast stochastic receding horizon control strategy. However in many physical and biological applications, typical models used to represent dynamics of an agent or a particle undergoing Brownian motion are in the form of continuous time stochastic differential equations.

Recent advances in stochastic receding horizon control policy have shown the stabilization of constrained linear systems with additive as well as multiplicative noise in both the states and the control actions (see [129, 73] and references therein). An extension of this policy for stabilizing constrained non-linear stochastic dynamical systems would not only facilitate emerging applications in small length scale dynamical systems but would also allow more rigorous analysis of closed-loop neuroprosthetic systems under various feedback scenarios studied in Chapter 4 and the design of closed-loop BMIs presented in Chapter 5 using non-linear stochastic models of cortical neurons such as the stochastic Izhikevich single neuron model.

# List of Publications

The presented dissertation was based on the following papers:

## Peer-reviewed journal articles

**Kumar, G.** and Kothare, M. V. (2013), "Trapping Brownian ensemble optimally using broadcast stochastic receding horizon control", *Provisionally Accepted (Revised and submitted), Automatica*

**Kumar, G.** and Kothare, M. V. (2013), "Broadcast stochastic receding horizon control of multi-agent systems", *Provisionally Accepted (Revised and submitted), Automatica*

**Kumar, G.** and Kothare, M. V. (2013), "On the continuous differentiability of inter-spike intervals of synaptically connected cortical spiking neurons in a neuronal network", *Accepted (13 May 2013), Neural Computation*

## Peer-reviewed conference proceedings

**Kumar, G.**, Schieber, M. H., Thakor, N. V. and Kothare, M. V. (June 2013), "Designing closed-loop brain-machine interfaces using optimal receding horizon control", *In Proceedings of the 2013 American Control Conference*, pages: 4933-4938, Washington DC, DC, USA

**Kumar, G.**, Aggarwal, V., Thakor, N. V., Schieber, M. H. and Kothare, M. V. (December 2011), "An optimal control problem in closed-loop neuroprostheses", *In Proceedings of the 2011* $50^{th}$ *Conference on Decision and Control and European Control Conference*, pages: 53-58, Orlando, FL, USA

**Kumar, G.**, Aggarwal, V., Thakor, N. V., Schieber, M. H. and Kothare, M. V. (June 2010), "Optimal parameter estimation of the Izhikevich single neuron model using experimental inter-spike interval (ISI) data", *In Proceedings of the 2010 American Control Conference*, pages: 3586-3591, Baltimore, MD, USA

**Kumar, G.**, Tiwari, P. Y., Marcopoli, V. and Kothare, M. V. (June 2009), "A study of a gun-turret assembly in an armored tank using model predictive control", *In Proceedings of the 2009 American Control Conference*, pages: 4848-4853, St. Louis, Missouri, USA

## Invited talks

**Kumar, G.** and Kothare, M. V. (November 2009), "Broadcast model predictive control of multi-cellular system", *CAST Plenary Session, In the 2009 AICHE Annual Meeting*, Nashville, TN, USA

## Refereed abstracts & talks

**Kumar, G.**, Schieber, M. H., Thakor, N. V. and Kothare, M. V. (November 2013), Designing closed-loop brain-machine interfaces using charge balanced biphasic stimulating currents", *In the 2013 Society for Neuroscience (SFN) Annual Meeting*, San Diego, CA, USA

**Kumar, G.**, Schiesser, W. E. and Kothare, M. V. (October 2012), A quantitative assessment of the Izhikevich neuron model against experimental data", *In the 2012 AICHE Annual Meeting*, Pittsburgh, PA, USA

**Kumar, G.** and Kothare, M. V. (October 2012), Regulating and Trapping an ensemble of Brownian particles by broadcast the stochastic receding horizon control policy", *In the 2012 AICHE Annual Meeting*, Pittsburgh, PA, USA

**Kumar, G.**, Thakor, N. V. and Kothare, M. V. (November 2011), Cortical neuronal network based neuroprosthetic finger control: A control theoretic approach", *In the 2011 Society for Neuroscience (SFN) Annual Meeting*, Washington DC, USA (Poster)

**Kumar, G.**, Thakor, N. V. and Kothare, M. V. (October 2011), Control of a motor intended neural prosthetic finger using a network of cortical motor neurons", *In the 2011 AICHE Annual Meeting*, Minneapolis, MN, USA

**Kumar, G.**, Aggarwal, V., Thakor, N. V., Schieber, M. H. and Kothare, M. V. (November 2010), A control approach towards closed-loop neural prosthesis", *In the 2010 Society for Neuroscience (SFN) Annual Meeting*, San Diego, CA, USA

**Kumar, G.**, Aggarwal, V., Thakor, N. V., Schieber, M. H. and Kothare, M. V.

(November 2010), Optimal parameter estimation of stochastic Izhikevich single neuron model using experimental inter-spike interval data", *In the 2010 AICHE Annual Meeting*, Salt Lake City, Utah, USA

**Kumar, G.**, Aggarwal, V., Thakor, N. V., Schieber, M. H. and Kothare, M. V. (November 2010), Design and control of a closed-loop neural prosthesis", *In the 2010 AICHE Annual Meeting*, Salt Lake City, Utah, USA

**Kumar, G.** and Kothare, M. V. (November 2010), A mathematical theory of manipulating suspended multiple Brownian particles simultaneously in a solution", *In the 2010 AICHE Annual Meeting*, Salt Lake City, Utah, USA

**Kumar, G.**, Aggarwal, V., Thakor, N. V. and Kothare, M. V. (November 2009), Optimal control of closed-loop neural prostheses", *In the 2009 AICHE Annual Meeting*, Nashville, TN, USA

# Bibliography

[1] L. F. Abbott. Lapicque's introduction of the integrate-and-fire model neuron (1907). *Brain Research Bulletin*, 50:303–304, 1999.

[2] S. Acharya, F. Tenore, V. Aggarwal, R. E. Cummings, M. H. Schieber, and N. V. Thakor. Decoding individuated finger movements using volume-constrained neuronal ensemble in the m1 hand area. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(1):15–23, 2008.

[3] V. Aggarwal. Developing a Brain-Computer Interface for Control of an Upper-Limb Neuroprosthesis. Master's thesis, Johns Hopkins University, Baltimore, MD, June 2007.

[4] V. Aggarwal, S. Acharya, F. Tenore, H. C. Shin, R. E. Cummings, M. H. Schieber, and N. V. Thakor. Asynchronous decoding of dexterous finger movements using m1 neurons. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(1):3–14, 2008.

[5] M. B. Ahrens, Q. J. M. Huys, and L. Paninski. Large-scale biophysical parameter estimation in single neurons via constrained linear regression. 2006.

[6] S. B. Andersson. Tracking a single fluorescent molecule with a confocal microscope. *Appl. Phys. B*, 80:809–816, 2005.

[7] S. S. Andrews, T. Dinh, and A. P. Arkin. Stochastic models of biological processes. *Encyclopedia of Complexity and System Science, Meyers, Robert (Ed.)*, 9:8730–8749, 2009.

[8] T. M. Apostol. *Mathematical Analysis*. Massachusetts:Addison-Wesley, 1957.

[9] A. Ashkin. History of optical trapping and manipulation of small-neutral particle, atoms, and molecules. *IEEE Journal on Selected Topics in Quantum Electronics*, 6(6):841–855, 2000.

[10] M. Bachar, J. Batzel, and S. Ditlevsen. *Stochastic Biomathematical Models with applications to neuronal modeling*. Springer, 2013.

[11] A. P. Batista, B. M. Yu, S. I. Ryu G. Santhanam, A. Afshar, and K. V. Shenoy. Cortical neural prosthesis performance improves when eye position is monitored. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(1): 24–31, 2008.

[12] H. Y. Benson, D. F Shanno, and R. J. Vanderbei. Interior-point methods for nonconvex nonlinear programming: Filter methods and merit functions. *Computational Optimization and Applications*, 23(2):257–272, 2002.

[13] E. Bizzi, N. Accornero, W. Chapple, and W. Hogan. Posture control and trajectory formation during arm movement. *J. Neurosci.*, 4:2738–2744, 1984.

[14] A. Blank, A. M. Okamura, and K. J. Kuchenbecker. Identifying the role of

proprioception in upper-limb prosthesis control: studies on targeted motion. *ACM Transactions on Applied Perception*, 7(3):1–23, 2010.

[15] S. Boyd and L. Vandenberghe, editors. *Convex Optimization*. Cambridge University Press, 2008.

[16] R. Brette and W. Gerstner. Adaptive exponential integrate-and-fire model as an effective description of neural activity. *J. Neurophysiol.*, 94:3637–3642, 2005.

[17] N. Brunel. Dynamics of sparsely connected networks of excitatory and inhibitory spiking neurons. *Journal of Computational Neuroscience*, 8:183–208, 2000.

[18] L. Buhry, F. Grassia, A. Giremus, E. Grivel, S. Renaud, and S. Saghi. Automated parameter estimation of the hodgkin-huxley model using the differential evolution algorithm: Application to neuromimetic analog integrated circuits. *Neural Computation*, 23(10):2599–2625, 2011.

[19] D. Bullock, P. Cisek, and S. Grossberg. Cortical networks for control of voluntary arm movements under variable force conditions. *Cerebral Cortex*, 8:48–62, 1998.

[20] A. Buonocore, L. Caputo, E. Pirozzi, and L.M. Ricciardi. On a stochastic leaky integrate-and-fire neuronal model. *Neural Computation*, 22(10):2558–2585, 2010.

[21] A. N. Burkitt. A review of the integrate-and-fire neuron model: I. homogeneous synaptic input. *Biol. Cybern*, 95:1–19, 2006.

[22] J. M. Carmena, M. A. Lebedev, R. E. Crist, J. E. O'Doherty, D. M. Santucci, D. F. Dimitrov, P. G. Patil, C. S. Henriquez, and M. A. L. Nicolelis. Learning to control a brainmachine interface for reaching and grasping by primates. *Plos Biol.*, 1(2):E46, 2003.

[23] F. Carpi and D. D. Rossi. Non invasive brain-machine interfaces. *Report*.

[24] J. K. Chapin, K. A. Moxon, R. S. Markowitz, and M. A. L. Nicolelis. Real-time control of a robot arm using simultaneously recorded neurons in the motor cortex. *Nature Neuroscience*, 2(7):664–670, 1999.

[25] I. Chattopadhyay and A. Ray. Supervised self-organization of homogeneous swarms using ergodic projections of Markov chains. *IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics*, 39(6):1505–1515, 2009.

[26] C. Clopath, R. Jolivet, A. Rauch, H-R. Luscher, and W. Gerstner. Predicting neuronal activity with simple models of the threshold type: adaptive exponential integrate-and-fire model with twp compartments. *Neurocomputing*, 70: 1668–1673, 2007.

[27] E. A. Coddington and N. Levinson. *Theory of ordinary differential equations*. New York:McGraw-Hill, 1955.

[28] E. A. Codling, M. J. Plank, and S. Benhamou. Random walk models in biology. *J R Soc Interface*, 5(25):813–834, 2008.

[29] A. E. Cohen and W. E. Moerner. Suppressing Brownian motion of individual biomolecules in solution. *Proc Natl Acad Sci USA*, 103(12):4362–4365, 2006.

[30] J. P. Cunnigham, P. Nuyujukian, V. Gilja, C. A. Chestek, S. I. Ryu, and K. V. Shenoy. A closed-loop human simulator for investigating the role of feedback-control in brain machine interfaces. *Journal of Neurophysiology*, 105:1932–1949, 2011.

[31] S. Dangi, S. Gowda, R. Hliot, and J. M. Carmena. Adaptive Kalman filtering for closed-loop brain-machine interface systems. In *2011 5th International IEEE/EMBS Conference on Neural Engineering (NER)*, pages 609–612, April-May 2011.

[32] P. Dayan and L. F. Abbott. *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems*. The MIT Press, 2001.

[33] M. de Kamps. A simple and stable numerical solution for the population density equation. *Neural Computation*, 15(9):2129–2146, 2003.

[34] J. Diedrichsen, R. Shadmehr, and R. B. Ivry. The coordination of movement: optimal feedback control and beyond. *Trends in Cognitive Sciences*, 14(1):31–39, 2009.

[35] S. Ditlevsen and P. Lansky. Estimation of the input parameters in the Ornstein-Uhlenbeck neuronal model. *Physical Review E*, 71:1–9, 2005.

[36] S. Ditlevsen and P. Lansky. Estimation of the input parameters in the Feller neuronal model. *Physical Review E*, 73:1–9, 2006.

[37] S. Ditlevsen and P. Lansky. Parameters of stochastic diffusion processes estimated from observations of first-hitting times: Application to the leaky integrate-and-fire neuronal model. *Physical Review E.*, 76:1–5, 2007.

[38] J. E. Doherty, M. A. Lebedev, P. J. Ifft, K. Z. Zhuang, S. Shokur, H. Bleuler, and M. A. L. Nicolelis. Active tactile exploration using a brain-machine-brain interface. *Nature*, 479:228–231, 2011.

[39] Y. Dong, S. Mihalas, A. Russell, R. E-Cummings, and E. Neibur. Estimating parameters of generalized integrate-and-fire neurons from the maximum likelihood of spike trains. *Neural Computation*, 23:2833–2867, 2011.

[40] S. Druckmann, T. K. Berger, S. Hill, F. Schrmann, H. Markram, and I. Segev. Evaluating automated parameter constraining procedures of neuron models by experimental and surrogate data. *Biol. Cybern.*, 99:371–379, 2008.

[41] R. Durrett. *Probability: Theory and Examples*. Thomson, third edition, 2005.

[42] B. Edwards, N. Engheta, and S. Evoy. Theory of simultaneous control of orientation and translational motion of nanorods using positive dielectrophoretic forces. *Journal of Applied Physics*, 98:1–7, 2005.

[43] C. Ethier, E. R. Oby, M. J. Bauman, and L. E. Miller. Restoration of grasp following paralysis through brain-controlled stimulation of muscles. *Nature*, 485:368–371, 2012.

[44] E. V. Evarts. Relation of pyramidal tract activity to force exerted during voluntary movement. *Journal of Neurophysiology*, 31:14–27, 1968.

[45] A. A. Faisal, L. P. J. Selen, and D. M. Wolpert. Noise in the nervous system. *Nature Reviews*, 9:292–303, 2008.

[46] O. Faugeras, J. Touboul, and B. Cessac. A constructive mean-field analysis of

multi-population neural networks with random synaptic weights and stochastic inputs. *Frontiers in Computational Neuroscience*, 3:1–28, 2009.

[47] E. E. Fetz. Operant conditioning of cortical unit activity. *Science*, 163:955–958, 1969.

[48] E. E. Fetz and M. A. Baker. Operantly conditioned patterns on precentral unit activity and correlated responses in adjacent cells and contralateral muscles. *J. Neurophysiol.*, 36:179–204, 1973.

[49] E. E. Fetz and D. V. Finocchio. Correlations between activity of motor cortex cells and arm muscles during operantly conditioned response patterns. *Exp. Brain Res.*, 23:217–240, 1975.

[50] D. Finkel. Direct optimization algorithm: Matlab routine. www4.ncsu.edu/ definkel/research/index.html, 2004.

[51] N. Fourcaud and N. Brunel. Dynamics of the firing probability of noisy integrate-and-fire neurons. *Neural Computation*, 14:2057–2110, 2002.

[52] G. J. Gage, K. A. Ludwig, K. J. Otto, E. L. Ionides, and D. P. Kipke. Naïve coadaptive cotical control. *Journal of Neural Engineering*, 2:52–63, 2005.

[53] W. V. Geit, E. D. Schutter, and P. Achard. Automated neuron model optimization techniques: a review. *Biol. Cybern.*, 99:241–251, 2008.

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

[55] A. P. Georgopoulos, A. B. Schwartz, and R. E. Kettner. Neuronal population coding of movements direction. *Science*, 233:1416–1419, 1986.

[56] W. Gerstner and W. Kistler. *Spiking Neuron Models*. New York:Cambridge University Press, 2002.

[57] W. Gerstner and R. Naud. How good are neuron models? *Science*, 326:379–380, 2009.

[58] Y. H. Ghallab and W. Badawy. *Lab-on-a-chip: Techniques, Circuits, and Biomedical Applications*. Artech House, first edition edition, 2010.

[59] V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, M. K. Churchland, M. T. Kaufman, J. C. Kao, S. I. Ryu, and K. V. Shenoy. A high-performance neural prosthesis enabled by control algorithm design. *Nature Neuroscience*, 15:1752–1757, 2012.

[60] V. Gilja, P. Nuyujukian, C. A. Chestek, J. P. Cunningham, B. M. Yu, J. M. Fan, S. I. Ryu, and K. V. Shenoy. A brain machine interface control algorithm designed from a feedback control perspective. In *2012 IEEE/EMBS Conference*, pages 1318–1322, August-September 2012.

[61] C. Gosse and V. Croquette. Magnetic tweezers: micromanipulation and force measurements at the molecular level. *Biophysical Journal*, 82:3314–3329, 2002.

[62] D. G. Grier. A revolution in optical manipulation. *Nature*, 424:21–27, 2003.

[63] T. L. Hanson, B. Omarsson, J. E. O'Doherty, I. A. Peikon, M. A. Lebedev,

and M. A. L. Nicolelis. High-side digitally current controlled biphasic bipolar microstimulator. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(3):331–340, 2012.

[64] A. Hanuschkin, S. Kunkel, M. Helias, A. Morrison, and M. Diesmann. A general and efficient method for incorporating precise spike times in globally time-driven simulations. *Frontiers in Neuroinformatics*, 4:1–19, 2010.

[65] E. Haskell, D. Q. Nykamp, and D. Tranchina. Population density methods for large-scale modelling of neural networks with realistic synaptic kinetics: cutting the dimension down to size. *Network: Computation in Neural Systems*, 12:141–174, 2001.

[66] N. G. Hastopoulos and J. P. Donoghue. The science of neural interface surface. *Annu. Rev. Neurosci.*, 32:249–266, 2009.

[67] R. D. Hayes, J. H. Byrnea, S. J. Coxb, and D. A. Baxter. Estimation of single-neuron model parameters from spike train data. *Neurocomputing*, 65-66: 517–529, 2005.

[68] R. Heliot, K. Ganguly, J. Jimenez, and J. C. Carmena. Learning in closed-loop brain-machine interfaces: modeling and experimental validation. *IEEE Transactions on Systems, Man, and Cybernetics-Part B:Cybernetics*, 40(5): 1387–1397, 2010.

[69] A. V. M. Herz, T. Gollisch, C. K. Machens, and D. Jaeger. Modelling single-neuron dynamics and computations: A balance of detail and abstraction. *SCIENCE*, 314:80–85, 2006.

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

[71] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. van der Smagt, and J. P. Donoghue. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 485:372–375, 2012.

[72] A. L. Hodgkin and A. F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.*, 117:500–544, 1952.

[73] P. Hokayem, E. Cinquemani, D. Chatterjee, F. Ramponi, and J. Lygeros. Stochastic receding horizon control with output feedback and bounded controls. *Automatica*, 48(1):77–88, 2012.

[74] F. C. Hoppensteadt and E. M. Izhikevich. *Weakly connected neural networks*. Springer, 1997.

[75] D. R. Humphrey, E. M. Schmidt, and W. D. Thompson. Predicting measures of motor performance from multiple cortical spike trains. *Science*, 170:758–762, 1970.

[76] E. M. Izhikevich. Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 14(6):1569–1572, 2003.

[77] E. M. Izhikevich. Which model to use for cortical spiking neurons? *IEEE Transactions on Neural Networks*, 15(5):1063–1070, 2004.

[78] E. M. Izhikevich, editor. *Dynamical systems in Neuroscience: The geometry of excitability and bursting*. The MIT Press, 2007.

[79] E. M. Izhikevich and G. M. Edelman. Large-scale model of mammalian thalamocortical systems. *PNAS*, 105(9):3593–3598, 2008.

[80] R. Jolivet and W. Gerstner. Predicting spike times of a detailed conductance-based neuron model driven by stochastic spike arrival. *Journal of Physiology - Paris*, 98:442–451, 2004.

[81] R. Jolivet, A. Rauch, H-R. Luscher, and W. Gerstner. Predicting spike timing of neocortical pyramidal neurons by simple threshold models. *J Comput Neurosci*, 21:35–49, 2006.

[82] R. Jolivet, A. Roth, F. Schrmann, W. Gerstner, and W. Senn. special issue on quantitative neuron modeling. *Biol. Cybern.*, 99:237–239, 2008.

[83] R. Jolivet, F. Schrmann, T. K. Berger, R. Naud, W. Gerstner, and A. Roth. The quantitative single-neuron modeling competition. *Biol. Cybern.*, 99:417–426, 2008.

[84] D. R. Jones, C. D. Perttunen, and B. E. Stuckman. Lipschitzian optimization without the Lipschitz constant. *Journal of optimization theory and application*, 79(1):157–181, 1993.

[85] A. A. Julius, A. Halasz, M. S. Sakar, H. Rubin, V. Kumar, and G. J. Pappas.

Stochastic modeling and control of biological systems: The lactose regulation system of Escherichia Coli. *Special Issue on Systems Biology*, pages 51–65, 2008.

[86] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering: Series D*, 82:35–45, 1960.

[87] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, editors. *Principles of Neural Science*. Mc Graw Hill, 2000.

[88] I. Karatzas and S. E. Shreve. *Brownian motion and stochastic calculus*. Springer, second edition edition, 2000.

[89] S. P. Kim, J. C. Sanchez, Y. N. Rao, D. Erdogmus, J. M. Carmena, M. A. Lebedev, M. A. L. Nicolelis, and J. C. Principe. A comparison of optimal mimo linear and nonlinear models for brain-machine interfaces. *J. Neural Eng.*, 3: 145–161, 2006.

[90] B. W. Knight. Dynamics of encoding in neuron populations: Some general mathematical features. *Neural Computation*, 12(3):473–518, 2000.

[91] S. Koyoma, S. M. Chase, A. S. Whitford, M. Velliste, A. B. Schwartz, and R. E. Kass. Comparison of brain-computer interface decoding algorithms in open-loop and closed-loop control. *Journal of Computational Neuroscience*, 29 (1-2):73–87, 2010.

[92] H. J. Kushner. On the stability of stochastic dynamical systems. *Proc Natl Acad Sci USA*, 53(1):8–12, 1964.

[93] H. J. Kushner. *Stochastic Stability and Control*. Academic Press, 1967.

[94] H. Kwakernaak and R. Sivan. *Linear Optimal Control Systems*. Wiley - Interscience, 1972.

[95] W. H. Kwon and S. Han. *Receding Horizon Control*. Springer, Berlin, 2005.

[96] P. Lansky and S. Ditlevsen. A review of the methods for signal estimation in stochastic diffusion leaky integrate-and-fire neuronal models. *Biol. Cybern.*, 99: 253–262, 2008.

[97] P. Lansky, P. Sanda, and J. He. The parameters of the stochastic leaky integrate-and-fire neuronal model. *J. Comput. Neurosci.*, 21:211–223, 2006.

[98] M. A. Lebedev and M. A. L. Nicolelis. Brain-machine interfaces: past, present and future. *Trends in Neurosciences*, 29(9):536–546, 2006.

[99] M. A. Lebedev, J. M. Carmena, J. E O'Doherty, M. Zacksenhouse, C. S. Henriquez, J.C. Principe, and M. A. L. Nicolelis. Cortical ensemble adaptation to represent velocity of an artificial actuator controlled by a brain-machine interface. *J. Neurosci.*, 25(19):4681–93, 2005.

[100] M. A. Lebedev, A. J. Tate, T. L. Hanson, Z. Li, J. E. O'Doherty, J. A. Winans, P. J. Ifft, K. Z. Zhuang, N. A. Fitzsimmons, D. A. Schwarz, A. M. Fuller, J. H. An, and M. A. L. Nicolelis. Future developments in brain-machine interface research. *Clinics (Sao Paulo)*, 66(S1):25–32, 2011.

[101] Z. Li, J. E. O'Doherty, T. L. Hanson, M. A. Lebedev, C. S. Henriquez, and M. A. L. Nicolelis. Unscented Kalman filter for brain-machine interfaces. *Plos one*, 4(7):e6243, 2009.

[102] Z. Li, J. E. O'Doherty, M. A. Lebedev, and M. A. L. Nicolelis. Adaptive decoding for brain-machine interfaces through Bayesian parameter updates. *Neural Computation*, 23(12):3162–3204, 2011.

[103] M. A. Maier and L. E. Shupe adn E. E. Fetz. Recurrent neural networks of integrate-and-fire cells simulating short-term memory and wrist movement tasks derived from continuous dynamic networks. *Journal of Physiology - Paris*, 97: 601–612, 2003.

[104] M. A. Maier and L. E. Shupe adn E. E. Fetz. Dynamic neural network models of the premotoneuronal circuitry controlling wrist movements in primates. *Journal of Computational Neuroscience*, 19:125–146, 2005.

[105] S. McKennoch, T. Voegtlin, and L. Bushnell. Spike-timing error backpropagation in theta neuron networks. *Neural Computation*, 21:9–45, 2009.

[106] S. Mihalas and E. Neibur. A generalized linear integrate-and-fire neural model produces diverse spiking behaviors. *Neural Computation*, 21:704–718, 2009.

[107] S. Mihalas and E. Niebur. A generalized linear integrate-and-fire model produces diverse spiking behaviors. *Neural Computation*, 21(3):704–718, 2009.

[108] K. D. Miller and F. Fumarola. Mathematical equivalence of two common forms of firing rate models of neural networks. *Neural Computation*, 24:25–31, 2012.

[109] J. Modolo, A. Garenne, J. Henry, and A. Beuter. Development and validation of a neural population model based on the dynamics of a discontinuous membrane potential neuron models. *Journal of Integrative Neuroscience*, 6:625–655, 2007.

[110] C. T. Moritz and E. E. Fetz. Volitional control of single cortical neurons in a brain-machine interface. *J. Neural Eng.*, 8(2):025017:1–15, 2011.

[111] C. T. Moritz, S. I. Perlmutter, and E. E. Fetz. Direct control of paralyzed muscles by cortical neurons. *Nature*, 486:639–643, 2008.

[112] G. H. Mulliken, S. Musallam, and R. A. Anderson. Decoding trajectories from posterior parietal cortex ensembles. *J. Neurosci.*, 28:12913–12926, 2008.

[113] P. Mullowney and S. Iyengar. Parameter estimation for a leaky integrate-and-fire neuronal model from isi data. *J. Comput. Neurosci.*, 24:179–194, 2008.

[114] R. Naud, T. K. Berger, B. Bathellier, and W. Gerstner. Quantitative single-neuron modeling: Competition 2009. http://www.incf.org/community/competitions/spike-time-prediction/2009, 2009.

[115] W. L. Nelson. Physical principles for economies of skilled movements. *Biological Cybernetics*, 45:135–147, 1983.

[116] K. C. Neuman and S. M. Block. Optical trapping. *Review of Scientific Instruments*, 75(9):2787–2809, 2004.

[117] M. A. L. Nicolelis. Brain-machine interfaces to restore motor function and probe neural circuits. *Nature Reviews Neuroscience*, 4:417–422, 2003.

[118] M. A. L. Nicolelis and M. A. Lebedev. Principles of neural ensemble physiology underlying the operation of brain-machine interfaces. *Nature Reviews*, 10:530–540, 2009.

[119] D. Q. Nykamp and D. Tranchina. A population density approach that facilitates large-scale modeling of neural networks: analysis and an application to orientation tuning. *Journal of Computational Neuroscience*, 8:19–50, 2000.

[120] L. Odehner and H. H. Asada. Kalman filter for inhomogeneous population Markov chains with application to stochastic recruitment control of muscle actuators. *In the Proceedings of the 2010 American Control Conference, Baltimore, MD*, pages 4774–4780, 2010.

[121] L. Odhnar, J. Ueda, and H. H. Asada. Stochastic Optimal Control Laws for Cellular Artificial Muscles. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pages 1554–1559, April 2007.

[122] L. U. Odhner and H. H. Asada. Stochastic recruitment control of large ensemble systems with limited feedback. *Journal of Dynamic Systems, Measurements, and Control*, 132(4):41008–410016, 2010.

[123] B. Oksendal. *Stochastic Differential Equations*. Springer, 6th edition, 3rd printing edition, 2005.

[124] A. Omurtag, B. W. Knight, and L. Sirovich. On the simulation of large population of networks. *Journal of Computational Neuroscience*, 8:51–63, 2000.

[125] M. W. Oram, M. C. Wiener, R. Lestienne, and B. J. Richmond. Stochastic nature of precisely timed spike patterns in visual system neuronal responses. *J. Neurophysiol.*, 81:3021–3033, 1999.

[126] L. Paninski, S. Shoham, M. R. Fellows, N. G. Hatsopoulos, and J. P. Donoghue.

Superlinear population encoding of dynamic hand trajectory in primary motor cortex. *The Journal of Neuroscience*, 24(39):8551–8561, 2004.

[127] J. P. Pfister, T. Toyoizumi, D. Barber, and W. Gerstner. Optimal spike-timing-dependent plasticity for precise action potential firing in supervised learning. *Neural Computation*, 18:1318–1348, 2006.

[128] J. W. Pillow, L. Paninski, and E. P. Simoncelli. Maximum likelihood estimation of a stochastic integrate-and-fire neural model. *Adv. Neural Information Processing Systems*, 16, 2004.

[129] J. A. Primbs. Stochastic receding horizon control of constrained linear systems with state and control multiplicative noise. *IEEE Transactions on Automatic Control*, 54:221–230, 2009.

[130] R. Probst, Z. Cummins, C. Ropp, E. Waks, and B. Shapiro. Flow control of small objects on chip. *IEEE Control Systems Magazine*, pages 26–53, April, 2012.

[131] J. A. Pruszynski and S. H. Scott. Optimal feedback control and the long-latency stretch response. *Exp. Brain Res.*, 218:341–359, 2012.

[132] R. Q. Quiroga and S. Panzeri. Extracting information from neuronal populations: information theory and decoding approaches. *Nature Reviews*, 10: 173–185, 2009.

[133] L. M. Ricciard and L. Sacerdote. The Ornstein-Uhlenbeck process as a model for neuronal activity. *Biol. Cybern.*, 35:1–9, 1979.

[134] L. M. Ricciard and L. Sacerdote. First-passage-time density and moments of the Ornstein-Uhlenbeck process. *J. App. Prob.*, 25, 1988.

[135] L.M. Ricciardi and S. Sato. First-passage-time density and moments of the Ornstein-Uhlenbeck process. *J. App. Prob.*, 25:43–57, 1988.

[136] H. Risken. *The Fokker-Planck Equation*. Springer, second edition edition, 1996.

[137] C. Rossant, D. F. M. Goodman, J. Platkiewicz, and R. Brette. Automatic fitting of spiking neuron models to electrophysiological recordings. *Frontiers in Neuroinformatics*, 4(2):1–10, 2010.

[138] C. Rossant, D. F. M. Goodman, B. Fontaine, J. Platkiewicz, A. K. Magnusson, and R. Brette. Fitting neuron models to spike trains. *Frontiers in Neuroscience*, 5(9):1–8, 2011.

[139] J. C. Sanchez and J. C. Prncipe. *Brain-Machine Interface Engineering*. Morgan and Claypool Publishers, 2007.

[140] R. W. H. Sargent. Optimal control. *Journal of Computational and Applied Mathematics*, 124:361–371, 2000.

[141] H. Scherberger. Neural control of motor prostheses. *Current Opinion in Neurobiology*, 19:629–633, 2009.

[142] M. H. Schieber. Individuated finger movements of rhesus monkeys: A means of quantifying the independence of the digits. *Journal of Neurophysiology*, 65(6): 1381–1391, 1991.

[143] W. E. Schiesser and G. W. Griffiths. *A Compendium of Partial Differential Equations Models: Method of Lines Analysis with Matlab*. Cambridge University Press, Cambridge, UK, 2009.

[144] E. M. Schmidt. Single neuron recording from motor cortex as a possible source of signals for control of external devices. *Annual Biomedical Engineering*, 8: 339–349, 1980.

[145] A. B. Schwartz, D. M. Taylor, and S. I. Helms Tillery. Extraction algorithms for cortical control of arm prosthetics. *Current opinion in neurobiology*, 11: 701–707, 2001.

[146] M. D. Serruya, N. G. Hastopoulos, L. Paninski, M. R. Fellows, and J. P. Donoghue. Instant neural control of a movement signal. *Nature*, 416:141–142, 2002.

[147] R. Shadmehr. Computational approaches to motor control. *Encyclopedia of Neuroscience*, 3:9–17, 2009.

[148] R. Shadmehr, M. A. Smith, and J. W. Krakauer. Error correction, sensory prediction, and adaptation in motor control. *Annual Review of Neuroscience*, 33:89–108, 2010.

[149] O. Shriki, D. Hansel, and H. Sompolinsky. Rate models for conductance-based cortical neuronal networks. *Neural Computation*, 15:1809–1841, 2003.

[150] L. Sirovich and B. Knight. Spiking neurons and the first passage problem. *Neural Computation*, 23:1–29, 2011.

[151] W. R. Softky and C. Koch. Cortical cells should fire regularly, but do not. *Neural Computation*, 4:643–646, 1992.

[152] A. J. Suminski, D. C. Tkach, A. H. Fagg, and N. G. Hatsopoulos. Incorporating feedback from multiple sensory modalities enhances brain-machine interface control. *The Journal of Neuroscience*, 30(50):16777–16787, 2010.

[153] D. Sussillo, P. Nuyujukian, J. M. Fan, J. C. Kao, S. D. Stavisky, S. I. Ryu, and K. V. Shenoy. A recurrent neural networks for closed-loop intracortical brain-machine interface decoders. *J. Neural Eng.*, 9:1–10, 2012.

[154] D. M. Taylor, S. I. H. Tiller, and A. B. Schwartz. Direct cortical control of 3d neuroprosthetic devices. *Science*, 416:1829–1832, 2002.

[155] E. Todorov. Optimality principles in sensorimotor control. *Nature Neuroscience*, 7:907–915, 2004.

[156] E. Todorov and M. I. Jordon. Optimal feedback control as a theory of motor coordination. *Nature Neuroscience*, 5(11):1226–1235, 2002.

[157] J. Touboul. Bifurcation analysis of a general class of nonlinear integrate-and-fire neurons. *SIAM J. APPL. MATH.*, 68(4):1045–1079, 2011.

[158] J. Touboul. On the simulation of nonlinear bidimensional spiking neuron models. *Neural Computation*, 23(7):1704–1742, 2011.

[159] J. Touboul and O. Faugeras. The spike trains probability distributions: a stochastic calculus approach. *Journal of Physiology - Paris*, 101:78–98, 2007.

[160] J. Touboul and O. Faugeras. A characterization of the first hitting time of double integral processes to curved boundaries. *Adv. Appl. Prob.*, 40:501–528, 2008.

[161] H. C. Tuckwell, editor. *Stochastic Processes in the Neurosciences*. Regional Conference Series In Applied Mathematics. Society for Industrial and Applied Mathematics, Philidelphia, PA, 1989.

[162] J. Ueda, L. Odhner, and H. H. Asada. Broadcast feedback of stochastic cellular actuators inspired by biological muscle control. *International Journal of Robotics Research, Special Issue on Bio-Robotics*, February 2007.

[163] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, 61:89–101, 1989.

[164] R. J. Vanderbei and D. F Shanno. An interior-point algorithm for nonconvex nonlinear programming. *Computational Optimization and Applications*, 13:231–252, 1999.

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

[166] T. P. Vogels, R. Rajan, and L. F. Abbott. Neural network dynamics. *Annual Reviews of Neuroscience*, 28:357–376, 2005.

[167] M. Volgushev and U. T. Eysel. Noise makes sense in neural computing. *Science*, 290(5498):1908–1909, 2000.

[168] D. J. Weber, B. M. London, J. A. Hokanson, C. A. Ayers, R. A. Gaunt, R. R. Torres, B. Zaaimi, and L. E. Miller. Limb-state information encoded by peripheral and central somatosensory neurons: Implications for an afferent interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(5): 501–513, 2011.

[169] J. Wessberg, C. R. Stambaugh, J. D. Kralik, P. D. Beck, M. Laubach, J. K. Chapin, J. Kim, Mandayam A. Srinivasan, S. James Biggs, and M. A. L. Nicolelis. Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. *Nature*, 408:361–365, 2000.

[170] J. R. Wolpaw and D. J. McFarland. Control of a two-dimensional movement signal by a noninvasive brain-computer interface in humans. *Proceedings of the National Academy of Sciences*, 101:17849–17854, 2004.

[171] L. B. Wood, A. Das, R. D. Kamm, and H. H. Asada. A stochastic broadcast feedback approach to regulating cell population morphology for microfluidic angiogenesis platforms. *IEEE Transactions on Biomedical Engineering*, 56(9): 2299–2303, 2009.

[172] W. Wu, M. J. Black, Y. Gao, E. Bienenstock, M. Serruya, and J. P. Donoghue. Inferring hand motion from multi-cell recordings in motor cortex using a Kalman filter. In *SAB02Workshop on Motor Control in Humans and Robots: On the Interplay of Real Brains and Artificial Devices*, pages 66–73, 2002.

[173] W. Wu, D. Mumford, Y. Gao, E. Bienenstock, and J. P. Donoghue. Modeling and decoding motor cortical activity using a switching Kalman filter. *IEEE Transactions on Biomedical Engineering*, 51(6):933–942, 2004.

[174] W. Wu, M. J. Black, D. Mumford, Y. Gao, E. Bienenstock, and J. P. Donoghue. Bayesian population decoding of motor cortical activity using a Kalman filter. *Neural Computation*, 18:80–118, 2006.

[175] B. B. Yellen, O. Hovorka, and G. Friedman. Arranging matter by magnetic nanoparticle assemblers. *Proc Natl Acad Sci USA*, 102(25):8860–8864, 2005.

[176] M. Zacksenhouse, M. A. Lebedev, J. M. Carmena, J. E. O'Doherty, C. Henriquez, and M. A.L. Nicolelis. Cortical modulations increase in early sessions with brain-machine interface. *Plos ONE*, 7:1–10, 2007.

# Biographical Information

Gautam Kumar earned a M.S. and a Ph.D. in Chemical Engineering from Lehigh University in 2008 and 2013 with a focus on emerging applications of optimal control theory in neural and small length scale dynamical systems. He received Bachelor of Technology in Chemical Engineering from Indian Institute of Technology, Kanpur, India in 2005. Kumar was awarded Rossin Doctoral Fellowship from Lehigh University in 2011. His research interests include mathematical and computational biology, bioinformatics, multi-scale modeling of biological and neurophysiological processes, understanding neural circuits and mechanisms that give rise to specific cognitive processes, and optimal control problems in small length scale dynamical systems including stochastic and quantum systems. After receiving his doctorate from Lehigh University, Kumar joined Department of Electrical and Systems Engineering, Washington University in St. Louis as a postdoctoral scholar.