A Novel Framework of Online, Task-Independent Cognitive State Transition Detection and Its Applications

By

Xiaoxu Kang

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy.

Baltimore, Maryland

September, 2014

Abstract

Complex reach, grasp, and object manipulation tasks require sequential, temporal coordination of a movement plan by neurons in the brain. Detecting cognitive state transitions associated with motor tasks from sequential neural data is pivotal in rehabilitation engineering.

The cognitive state detectors proposed thus far rely on task-dependent (TD) models, i.e., the detection strategy exploits *a priori* knowledge of the movement goals to determine the actual states, regardless of whether these cognitive states actually depend on the movement tasks or not. This approach, however, is not viable when the tasks are not known *a priori* (e.g., the subject performs many different tasks) or when there is a paucity of neural data for each task. Moreover, some cognitive states (e.g., holding) are invariant to the tasks performs.

I first develop an offline, task-dependent cognitive state transition detector and a kinematics decoder to show the feasibility of distinguishing between cognitive states based on their inherent features extracted via a hidden Markov model (HMM) based detection framework. The proposed framework is designed to decode both cognitive states and kinematics from ensemble neural activity. The proposed decoding framework is able to a) automatically differentiate between baseline, plan, and movement, and b) determine novel holding epochs of neural activity and also estimate the epoch-dependent kinematics. Specifically, the framework is mainly composed of a hidden Markov model (HMM) state decoder and a switching linear system (S-LDS) kinematics decoder. I take a supervised approach and use a generative framework of neural activity and kinematics. I demonstrate the decoding framework using neural recordings from ventral premotor (PMv) and dorsal premotor (PMd) neurons of a non-human primate executing four complex reach-to-grasp tasks along with the corresponding kinematics recording. Using the HMM state decoder, I demonstrate that the transitions between neighboring epochs of neural activity, regardless of the existence of any external kinematics changes, can be detected with high accuracy (>85%) and short latencies (<150 ms). I further show that the joint angle kinematics can be estimated reliably with high accuracy (mean = 88%) using a S-LDS kinematics decoder. In addition, I demonstrate that the use of multiple latent state variables to model the within-epoch neural activity variability can improve the decoder performance. This unified decoding

framework combining a HMM state decoder and a S-LDS may be useful in neural decoding of cognitive states and complex movements of prosthetic limbs in practical brain-computer interface implementations.

I then develop a real-time (online) task-independent (TI) framework to detect cognitive state transitions from spike trains and kinematic measurements. I applied this framework to 226 single-unit recordings collected via multi-electrode arrays in the premotor dorsal and ventral (PMd and PMv) regions of the cortex of two non-human primates performing 3D multi-object reach-to-grasp tasks, and I used the detection latency and accuracy of state transitions to measure the performance. I found that, in both online and offline detection modes, (i) TI models have significantly better performance than TD models when using neuronal data alone, however (ii) during movements, the addition of the kinematics history to the TI models further improves detection performance. These findings suggest that TI models may be able to more accurately detect cognitive state transitions than TD under certain circumstances. The proposed framework could pave the way for a TI control of prosthesis from cortical neurons, a beneficial outcome when the choice of tasks is vast, but despite that the basic movement cognitive states need to be decoded.

Based on the online cognitive state transition detector, I further construct an online task-independent kinematics decoder. I constructed our framework using single-unit recordings from 452 neurons and synchronized kinematics recordings from two non-human primates performing 3D multi-object reach-to-grasp tasks. I find that (i) the proposed TI framework performs significantly better than current frameworks that rely on TD models ($p = 0.03$); and (ii) modeling cognitive state information further improves decoding performance. These findings suggest that TI models with cognitive-state-dependent parameters may more accurately decode kinematics and could pave the way for more clinically viable neural prosthetics.

Primary Reader: Dr. Nitish Thakor

Secondary Reader: Dr. Sridevi Sarma

Thesis Committee:

Dr. Nitish Thakor (primary adviser, reader),

Department of Biomedical Engineering,

Johns Hopkins University

Dr. Sridevi Sarma (adviser, reader),

Department of Biomedical Engineering,

Johns Hopkins University

Dr. Marc Schieber (adviser),

Department of Neurology, Neurobiology and Anatomy,

University of Rochester Medical Center

Acknowledgements

I deeply appreciate all of my mentors for their patience and support through my journey in the PhD program. I was very fortunate to have the opportunities to work with scientists and researchers from various backgrounds and thus develop a better grasp and understanding of the big picture of where to take my thesis. The rich exposure is immensely beneficial in the real world, as I will be able to make judicious choices when I come across a research topic, and also be able to discuss intelligently with other researchers and also explain them in plain languages to laymen.

I also much appreciate all of my family, mentors, and friends for supporting me through the ups and downs of my life in the past 7 years in the USA. As I am living in the alien land different from my country where I was born, their support and encouragement has allowed me to be brave enough to try new things and experiment with my own potentials and limitations.

I constantly make mistakes. But through mistakes and failures, I have learned the right way and become wiser. After I fall and get back on my feet, I feel that I am stronger each time and have a better understanding about life and what is important to add to the meaning of my life. I believe that everyone is only a passenger on this planet with limited time. What one can achieve in a limited time is not limited in any way but only by one's own imagination and aspiration. I would like to make the world a better place with my sweats, efforts and intelligence. I would like to dedicate my life to be a life-time technology enabler to change the course of history and improve significantly the quality of life for billions of people worldwide. But I also know that I cannot get there alone, but with the support of others' kindness and help, which I greatly appreciate.

Dedication

To my family, friends and mentors.

Contents

This page is intentionally left blank.

List of Tables

List of Figures

Chapter 1 Introduction

1.1 Overview of Brain-Computer Interface

1.1.1 BCI Research History

1.1.2 BCI for Target Movement Reconstruction

1.1.3 Brain Anatomy

1.1.4 Brain Pathway and Circuits

1.2 Thesis Motivation

1.3 Thesis Overview

1.1 Overview of Brain-Computer Interface

The ability to communicate and interact with objects/devices via thoughts alone is a long lasting fantasy of human societies. With the advancement of cognitive sciences and technology, this fantasy has been gradually transformed from fairy tales and myths to technologies and tools that we can use to capture and monitor one's mind and thoughts and use it to control external devices or paralyzed body parts, or brain-computer interfaces (BCIs) or brain-machine interfaces (BMIs).

BCI, as a means to connect brain (or more specifically brainwaves) with computers and BMI as means to connect brain to machines such as robots and prosthesis, are emerging fields. For example, these technologies may enable disabled people to find means to substitute lost motor functions with the help of computers and various machineries. For example, BCIs enable completely paralyzed patients to use brain signals to control their paralyzed limbs or external prosthetic devices. A BCI activates and controls electronic or mechanical devices with various forms of brain signals mainly, including but not limited to:

- Action potentials from nerve cells or nerve fibers [1];
- Synaptic and extracellular field potentials [2, 3];

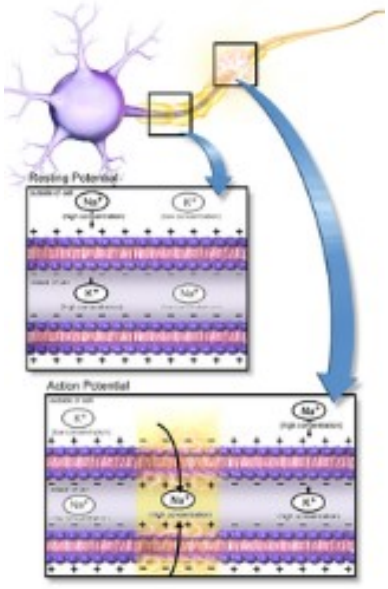


Fig. 1-1 Action potential propagation along an axon ("Blausen 0011 ActionPotential Nerve" by BruceBlaus. When using this image in external sources it can be cited as:Blausen.com staff. "Blausen gallery 2014". Wikiversity Journal of Medicine. DOI:10.15347/wjm/2014.010. ISSN 20018762. - Own work. Licensed under Creative Commons Attribution 3.0 via Wikimedia Commons -

http://commons.wikimedia.org/wiki/File:Blausen_0011_ActionPotential_Nerve.png#mediaviewer/File:Blausen_0011_ActionPotential_Nerve.png).

- Electrocorticograms [4];

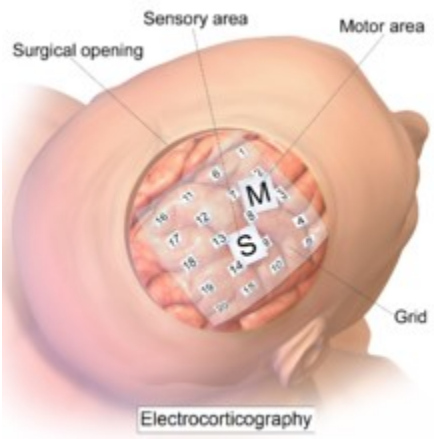


Fig. 1-2 Intracranial electrode grid for electrocorticography ("Intracranial electrode grid for electrocorticography" by BruceBlaus. When using this image in external sources it can be cited as:Blausen.com staff. "Blausen gallery 2014". Wikiversity Journal of Medicine. DOI:10.15347/wjm/2014.010. ISSN 20018762. - Own work. Licensed under Creative Commons Attribution 3.0 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:Intracranial_electrode_grid_for_electrocorticography.png#mediaviewer/File:Intracranial_electrode_grid_for_electrocorticography.png).

- Slow cortical potential (SCP) of the EEG [5];

Fig. 1-3 Generalized 3 Hz spike and wave discharges reflecting seizure activity. ("Spike-waves". Licensed under Creative Commons Attribution-Share Alike 2.0 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:Spike-waves.png#mediaviewer/File:Spike-waves.png).

- EEG and MEG oscillations, mainly sensorimotor rhythm (SMR), also called mu-rhythm [6];
- P300 and other event-related brain potentials [7];
- BOLD response in functional magnetic resonance imaging [8];
- Near infrared spectroscopy (NIRS) measuring cortical blood flow [9];
- Functional magnetic resonance imaging (fMRI) measuring subjects' responses to different visual stimuli.

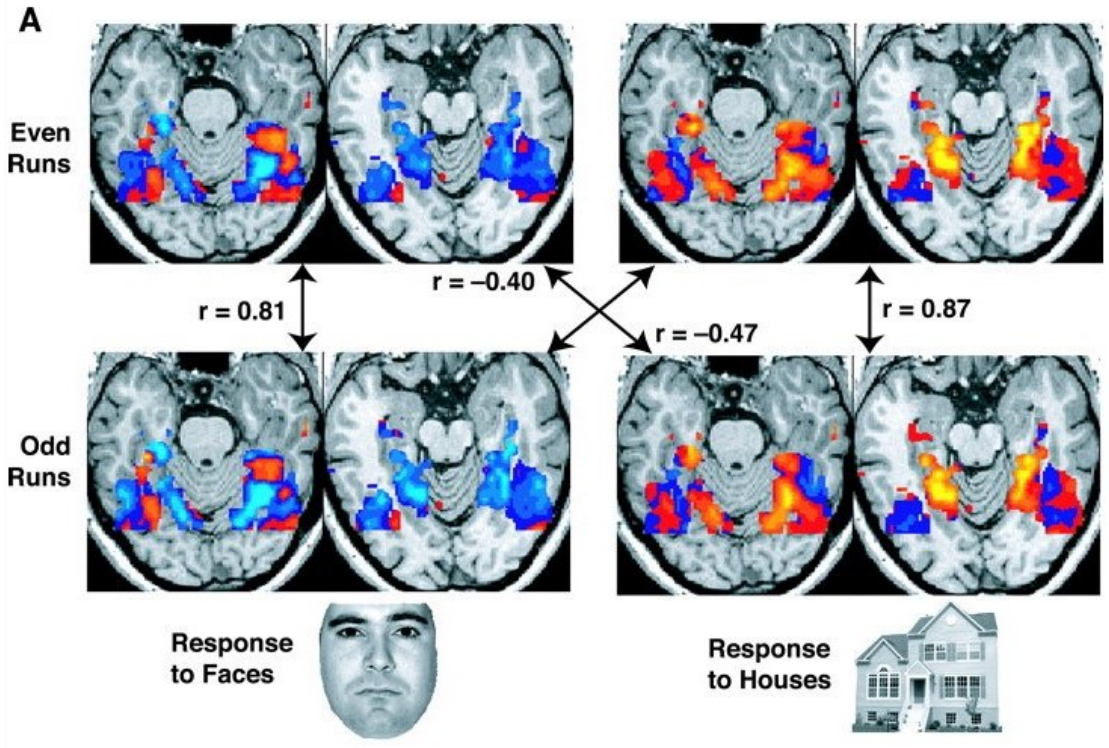


Fig. 1-4 These fMRI images are from a study showing parts of the brain lighting up on seeing houses and other parts on seeing faces. The 'r' values are correlations, with higher positive or negative values indicating a better match ("Haxby2001" by National Institute of Mental Health - US Department of Health and Human Services: National Institute of Mental Health. Licensed under Public domain via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:Haxby2001.jpg#mediaviewer/File:Haxby2001.jpg).

The success of BCIs can have significant impact to the society, especially to those patients who suffer from severe neuromuscular injuries, neurodegenerative diseases, or loss of motor functions such as amputees.

1.1.1 BCI Research History

Hans Berger discovered human EEG and speculated about the possibility of reading “thoughts” from analyzing and recording EEG traces. Grey Walter constructed the first automatic frequency analyzer to discriminate “thoughts” and language in the human EEG [10]. Fetz was one of the first to discuss invasive operant conditioning of cortical spike trains in animals [11], the idea that subsequently formed the basis for training animals to develop conditioned cortical responses to operate computer interfaces such as movement of computer cursor under neuronal control.

BCI research has seen significant progress in recent years via implanting multielectrode grids on animal and human subjects [12, 13, 14]. The goal of these research experiments is to re-construct intended skilled

movements from neuronal firing activities online. Various decoding methods have been applied or developed during the process. “Sparse coding” approaches of motor learning [15] and directional coding vectors of motor neurons [16] have been applied to reconstruct complex movements online from the spiking activity of a relatively few motor neurons. Carmena et al [13] showed that in their monkey subjects, after extensive training of a reaching and grasping movement, 32 neurons were sufficient to execute the target movement with an artificial prosthetic limb. Chapin et al [17] demonstrated that rats can use “thoughts” (extracellular firing of cortical neurons) move a lever with an artificial arm in a box for rewards.

BCIs have also seen significant applications in helping patients in paralysis to communicate with the outside world. Amyotrophic Lateral Sclerosis (ALS) is a progressive motor disease resulting in patients losing motor functions or complete paralysis [18]. Birbaumer et al [19] developed a BCI system for ALS patients with slow cortical potentials (SCP). However, the BCIs based on SCP need long training periods, and the letter selection in a BCI typing or texting paradigm and thus communication speed is slow (~one minute per letter). BCIs have also been applied in epileptic patients [20] as patients tried spelling or performing imagery tasks with cortically implanted electrodes. Results indicated that it was possible to differentiate imagination of hand, tongue, and mouth movements using the Electrocorticogram (ECoG).

1.1.2 BCIs for Target Movement Reconstruction

BCI research has progressed significantly from research related with target movement reconstruction from neuronal spike trains or synaptic filed potentials [21, 22]. Previous research has shown that with enough training and implementation of learning algorithms, monkeys can move cursors on screens to track targets or control an artificial hand to move in four directions directed by spike activity. Further research has also shown that complex movements can be reconstructed from a small number of selected neurons [23, 22, 14, 24]. Similar observations have been made on human subjects. A single quadriplegic patient was implanted with a floating multielectrode array to record neuronal spike trains and local field potentials [21]. With enough training, the patient was able to use his “thoughts” to control a computer cursor to move in several directions.

1.1.3 Brain Anatomy

Human brain can be viewed as hierarchical distributed computing systems, with specialized sub-systems for certain tasks [25]. The recent advancement in cognitive sciences and neuroscience has helped us identify the topographical maps of brains corresponding to their specialized cognitive functions, by studying the various effects of brain injuries and observing brain activities using non-invasive (brain imaging methods, EEG) and invasive methods (multi-electrode array recording of neuronal spike trains).

Fig. 1-5 Human subject with EEG recording electrodes arranged around his head ("EEG mit 32 Electroden" by Aschoeke - Image:EEG 32 electrodes.jpg, modifications by Kristina. Licensed under Creative Commons Attribution-Share Alike 3.0 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:EEG_mit_32_Electroden.jpg#mediaviewer/File:EEG_mit_32_Electroden.jpg).

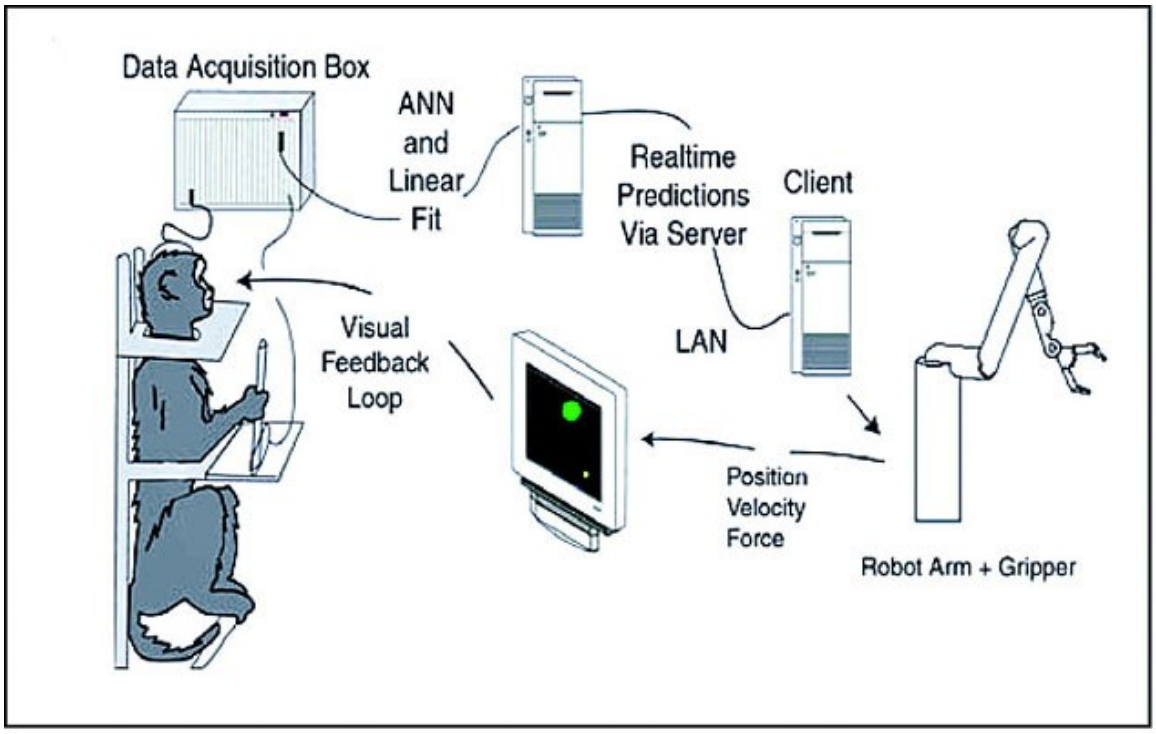


Fig. 1-6 Design of an experiment in which brain activity from a monkey was used to control a robotic arm ("Brain-computer interface (schematic)" by see source - This file is adapted from figure '1' from:Carmena, J.M., Lebedev, M.A., Crist, R.E., O'Doherty, J.E., Santucci, D.M., Dimitrov, D.F., Patil, P.G., Henriquez, C.S., Nicolelis, M.A.L. (2003) Learning to control a brain-machine interface for reaching and grasping by primates. PLoS Biology, 1: 193-208. PLoS is an open-source, peer-reviewed scientific journal publisher. – Figure 1. doi:10.1371/journal.pbio.0000042. Licensed under Creative Commons Attribution 2.5 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:Brain-computer_interface_(schematic).jpg#mediaviewer/File:Brain-computer_interface_(schematic).jpg).

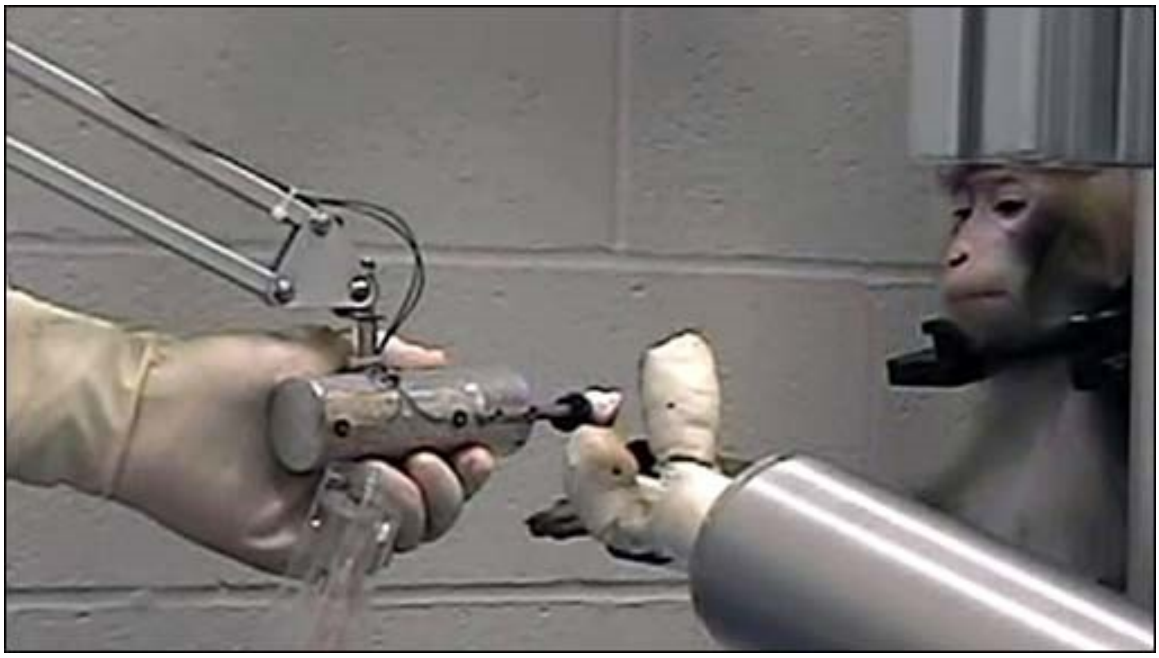


Fig. 1-7 Monkey operating a robotic arm with brain–computer interfacing (Schwartz lab, University of Pittsburgh) ("Monkey using a robotic arm" by BBC Courtesy of the University of Pittsburgh Medical Center. The image appeared with a BBC news stream about the monkey and robotic arm: Monkey's brain controls arm, and on a number of other websites (see 'Other information' below).. Licensed under Fair use of copyrighted material in the context of Brain-computer interface via Wikipedia - http://en.wikipedia.org/wiki/File:Monkey_using_a_robotic_arm.jpg#mediaviewer/File:Monkey_using_a_robotic_arm.jpg).

The brain can be categorized into two groups: the cerebral cortex and sub-cortical regions. The cerebral cortex is evolutionarily newer than the sub-cortical regions, representing the largest and most complex part of the human brain. In comparison, the sub-cortical regions are mainly in charge of more basic functions such as respiration, heart rates and temperature regulation. Many sub-cortical functions are not controlled by one's consciousness. Since BCIs are driven by one's conscious control of brain activities, or, "thoughts", to drive the external devices, current BCI work has mainly focused on the cerebral cortex area.

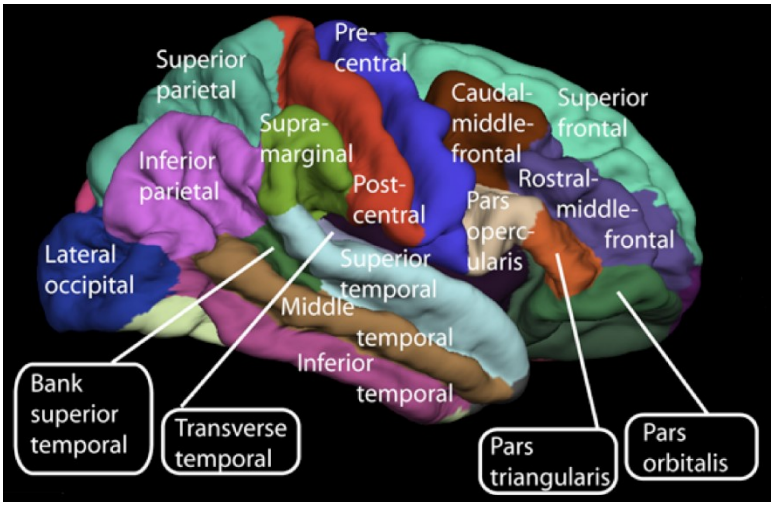


Fig. 1-8 Lateral surface of the human cerebral cortex ("Lateral surface of cerebral cortex - gyri" by Patric Hagmann et.al. - Hagmann P, Cammoun L, Gigandet X, Meuli R, Honey CJ, et al. (2008) Mapping the Structural Core of Human Cerebral Cortex. PLoS Biol 6(7): e159. doi:10.1371/journal.pbio.0060159. Licensed under Creative Commons Attribution 2.5 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:Lateral_surface_of_cerebral_cortex_-_gyri.png#mediaviewer/File:Lateral_surface_of_cerebral_cortex_-_gyri.png).

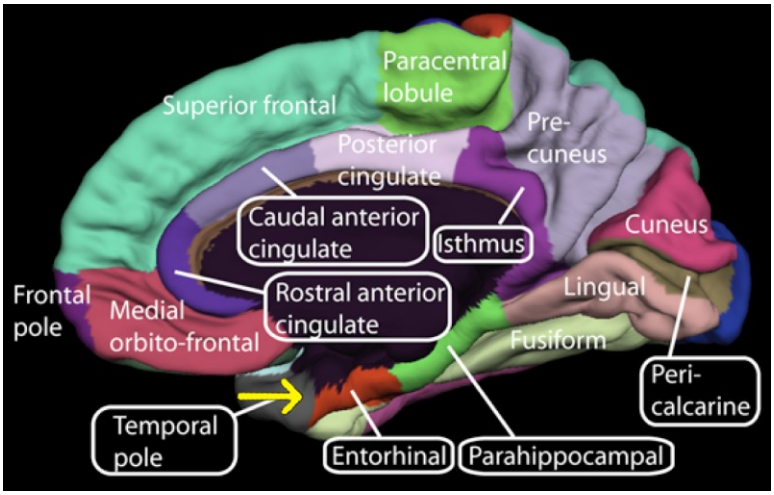


Fig. 1-9 Medial surface of the human cerebral cortex ("Medial surface of cerebral cortex - entorhinal cortex" by Hagmann P, Cammoun L, Gigandet X, Meuli R, Honey CJ, et al. - File:Medial surface of cerebral cortex - gyri.png. Licensed under Creative Commons Attribution 2.5 via Wikimedia Commons - http://commons.wikimedia.org/wiki/File:Medial_surface_of_cerebral_cortex_-_entorhinal_cortex.png#mediaviewer/File:Medial_surface_of_cerebral_cortex_-_entorhinal_cortex.png).

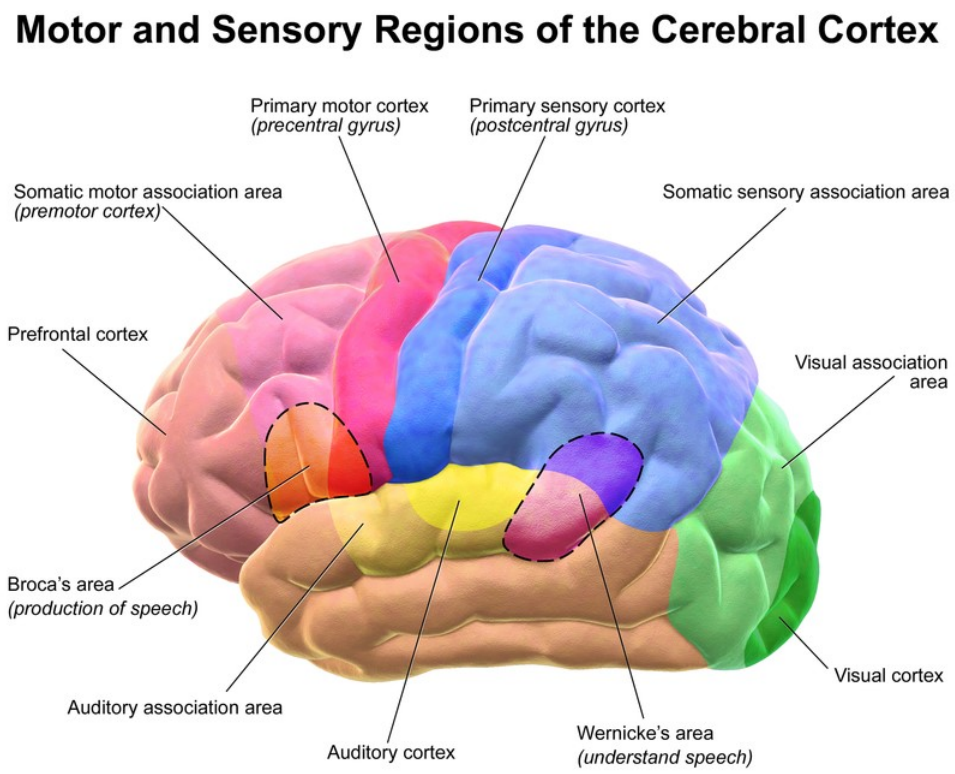


Fig. 1-10 Motor and Sensory Regions of the Cerebral Cortex ("Blausen 0102 Brain Motor&Sensory" by BruceBlaus - Own work. Licensed under Creative Commons Attribution 3.0 via Wikimedia Commons).

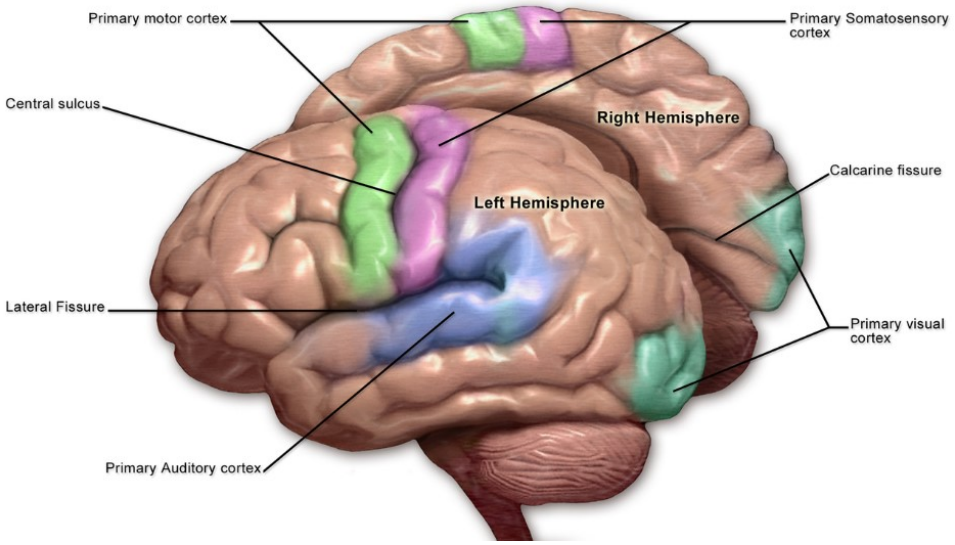


Fig. 1-11 Motor and Sensory Regions of the Cerebral Cortex. ("Blausen 0103 Brain Sensory&Motor" by BruceBlaus - Own work. Licensed under Creative Commons Attribution 3.0 via Wikimedia Commons)

1.1.4 Human Brain Cortical Divisions

The cerebral cortex has two nearly symmetrical hemispheres: left and right hemispheres. Each hemisphere is typically divided into four “lobes”: the frontal lob, parietal lobe, occipital lobe, and temporal lobe.

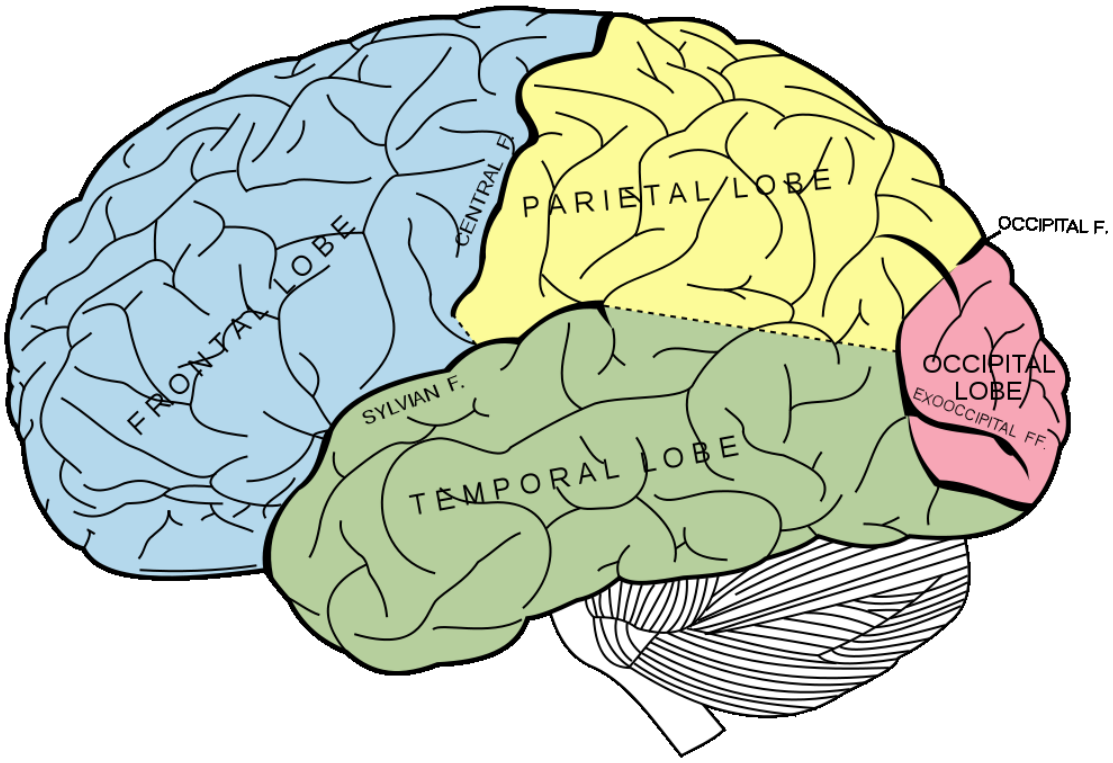


Fig. 1-12 The four lobes of the cerebral cortex ("Gray728" by Mysid - Vectorized in CorelDraw by Mysid, based on the online edition of Gray's Anatomy. Licensed under Public domain via Wikimedia Commons).

Different lobes have specialized functions. For example, an area of the occipital lobe is in charge of visual reception, visual-spatial processing, movement, and color recognition, and parietal lobe contains areas specializing in somatosensation, hearing, language, attention and spatial cognition. The frontal lobe is to control attention, abstract thinking, and behavior, problem solving tasks and physical reactions and personality. The temporal lobe controls auditory and visual memories, language, hearing and speech.

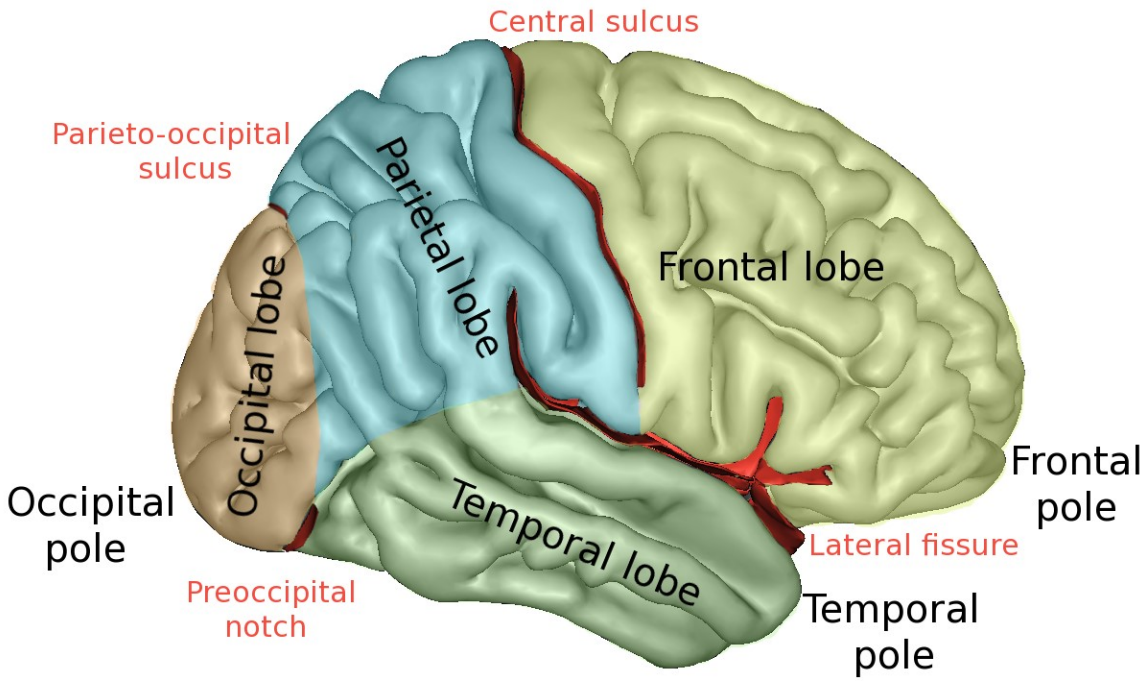


Fig. 1-13 Lataeral surface of cerebrum. 4 lobes are shown ("LobesCaptsLateral" by Sebastian023 - File:LobesCapts.png. Licensed under Creative Commons Attribution-Share Alike 3.0 via Wikimedia Commons)

1.2 Thesis Motivation

Neural prosthetics is an emerging scientific field that uses signals from cortical neurons to decode motor intent and execution [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. Several kinematic decoding systems to decode movements from neuronal ensemble activity in motor related cortical areas have been proposed in the past twenty years [1, 2, 3, 10, 4, 5, 6, 8, 9].

An important problem in the neural prosthetics design is to estimate “cognitive states,” i.e., the sequence of internal neural states that occurs between the movement onset and termination [10]. In neural prosthetics, cognitive states are internal states that occur either before or during movements but are invariant to movement tasks [11]. For example, during 3D reach-to-grasp movements, cognitive states characterize the baseline idle state, the planning of a movement, the execution of a movement, and finally the holding state [12] (Fig.1-14). These states are task independent, i.e. these states occur regardless of the specific target objects reached at the end of each task. On the other hand, grasping different objects requires task dependent movement execution.

To date, prior work has focused on detecting cognitive state transition using task-dependent (TD) models [13, 5, 14, 15]. That is, in order to detect a cognitive state, the task must be either known *a priori* or determined in real-time. If the neural prosthetics must work with subjects performing a large variety of tasks, a new TD model would be needed for each novel movement task, thus making the paradigm rapidly unmanageable. Furthermore, even if the computation would not pose a hurdle, one needs data from each task to estimate the models, despite the fact that several cognitive states (e.g., idle, holding, etc.) are invariant to the specific task being performed [11].

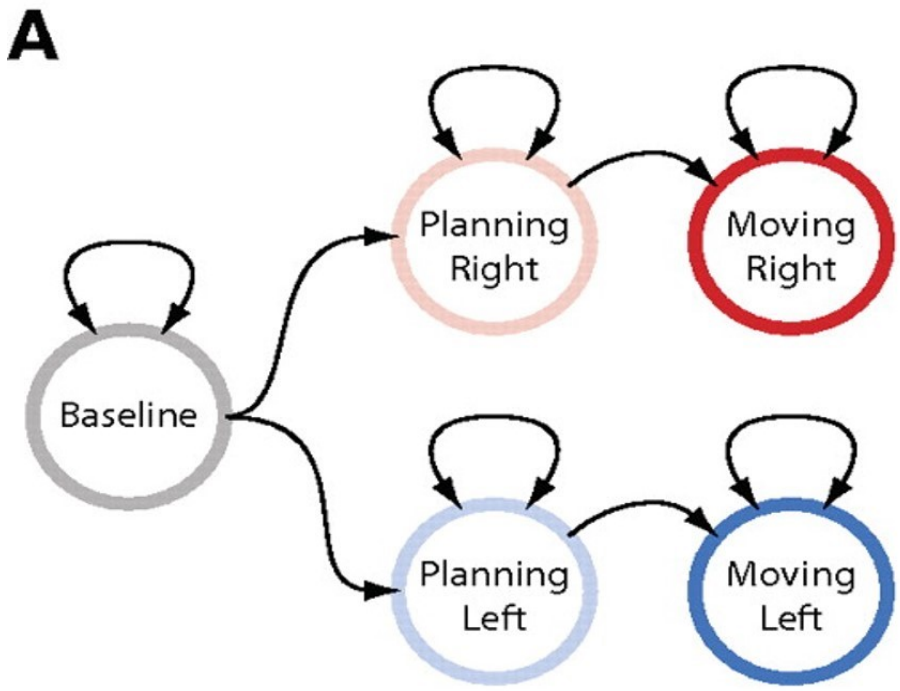


Fig. 1-14 Task-dependent 5-state HMMs.

Therefore, we propose a real-time TI framework to detect cognitive state transitions from spike trains and kinematic measurements, where (1) a TI model is used with neuronal spike train measurements alone to detect cognitive states when the subject does not move; and, (2) when movements are detected, a TI model is used with both neuronal spike train and kinematic measurements to detect cognitive states [14, 16, 15]. We hypothesize that TI models will allow for a more accurate detection of cognitive state transitions than the TD models.

We assume that the brain evolves according to a hidden Markov model (HMM), wherein the HMM's outputs are cortical spike trains and – only when the subject is moving – kinematic measurements. The HMM's outputs are modeled with generalized linear point process models [17] with history dependencies. Similar state-space techniques have also been used [18, 19, 20, 17, 21, 13] in neural signal modeling. Related models have also been developed and applied in the fields of neural information processing and neural data analysis [22, 23, 24, 25, 26, 27, 28].

We constructed the real-time TI cognitive state transition detection framework using 226 single-unit recordings collected via multi-electrode arrays in premotor dorsal and ventral (PMd and PMv) cortical areas of two non-human primates performing 3D multi-object reach to-grasp tasks [12]. We compared the latency-accuracy performance [29] of the TI vs. TD models both in offline and online detection modes.

The key questions we are trying to answer are:

- Is it possible to detect cognitive state transitions offline?
- Is it possible to detect cognitive state transitions online?
- If yes, what are the applications that can benefit from having cognitive state transitions?

So the aims of the thesis are as follows:

- **A1**: Task-dependent, offline, manually defined cognitive state transition detector and kinematics decoder

This is to establish the feasibility of detecting cognitive state transitions. We built a task-dependent model, using an offline batch processing method, to identify the distinct features of each cognitive state and their transitions. We demonstrate the proposed decoding framework using neural recordings from ventral premotor (PMv) and dorsal premotor (PMd) neurons of a non-human primate executing four complex reach-to-grasp tasks along with the corresponding kinematics recordings. Using the HMM state decoder, we demonstrate that the transitions between neighboring epochs, regardless of the existence of any external kinematics changes, can be detected with high accuracy and short latencies. We further show that the hand joint angle kinematics can be estimated reliably with high accuracy. In addition, we demonstrate that using multiple latent state variables to model the within-epoch neural activity variability can improve the decoding performance of the HMM state decoder.

- **A2**: Task-independent, online, data-driven cognitive state transition detector

Complex reach, grasp, and object manipulation tasks require sequential, temporal coordination of a movement plan by neurons in the brain. Detecting cognitive state transitions associated with motor tasks from sequential neural data is pivotal in rehabilitation engineering. The cognitive state detectors proposed thus far reply on task-dependent (TD) models, i.e., the detection strategy exploits *a priori* knowledge of the movement goals to determine the actual states, regardless of whether these cognitive states actually depend on the movement tasks or not. This approach, however, is not viable when the tasks are not known *a priori* (e.g., the subject performs many different tasks) or when there is a paucity of neural data for each task. Moreover, some cognitive states (e.g., holding) are invariant to the tasks performs. We propose a real-time (online) task-independent (TI) framework to detect cognitive state transitions from spike trains and kinematic measurements. We applied this framework to 226 single-unit recordings collected via multi-electrode arrays in the PMd and PMv regions of the cortex of two non-human primates performing 3D multi-object reach-to-grasp tasks, and we used the detection latency and accuracy of state transitions to measure the performance. We found that, in both online and offline detection modes, (i) TI models have significantly better performance than TD models when using neuronal data alone, (ii) during movements, the addition of the kinematics history to the TI models further improves detection performance. These findings suggest that TI models may accurately detect cognitive state transitions. Our framework could pave the way for a TI control of prosthesis from cortical neurons.

- **A3**: Task-independent, online, data-driven cognitive-state-based kinematics decoder

Decoding kinematics from neural measurements for prosthetic control is currently limited to a small set of predefined tasks. The decoding algorithms proposed thus far rely on task-dependent (TD) encoding models, i.e., the decoding algorithms exploit *a priori* knowledge of the movement tasks. This approach, however, is not viable when the movement tasks are not known *a priori*, and is not scalable when there are a large number of tasks. Moreover, cognitive state information during tasks, such as planning or moving, is not utilized in current encoding models and thus neuronal variability due to cognitive state is not explicitly characterized. Here, we propose a real-time (online) task-independent (TI) decoding algorithm that constructs different encoding models for different cognitive states - not for different movement tasks. We constructed our framework using single-unit recordings from 452 neurons and synchronized kinematics recordings from two non-human primates performing 3D multi-object reach-to-grasp tasks. We find that (i) our TI framework

performs significantly better than current frameworks that rely on TD models ($p = 0.03$); and (ii) modeling cognitive state information further improves decoding performance. These findings suggest that TI models with cognitive-state-dependent parameters may more accurately decode kinematics and could pave the way for more clinically viable neural prosthetics.

1.3 Thesis Overview

The rest of the thesis is organized along the three aims mentioned above: chapter 2 discusses the experimental set-up related with brain signals and kinematics recording in non-human subjects; chapters 3 describes the theoretical backgrounds (hidden Markov models, point process models); chapters 4 and 5 in details discuss the offline and online task-independent cognitive state transition detector; chapter 6 describes the online task-independent kinematics decoding framework; chapter 7 finally describes the significance of the work, limitations, and future directions.

Chapter 2 Experiments

2.1 Behavioral Tasks

2.2 Microelectrode Array Implantation

2.3 Acquisition of Neurophysiological Signal

2.1 Behavioral Tasks

All studies were carried out by Dr. Schieber and his students under a protocol approved by the University of Rochester Institutional Animal Care and Use Committee. Two male rhesus monkeys (Macaca mulatta) were trained to sit in a polycarbonate primate chair that restrained the neck, torso and legs, and to follow visual cues to manipulate, grasp, and reach towards one of four target objects. The four objects include sphere, perpendicular mounted cylinder (mallet), pushbutton, or peripheral coaxial cylinder (pull handle), arranged in a planar circle at 60º intervals around the center home cylindrical object (home object) (Fig. 2-1). Each object was mounted on the end of a 2-3', 0.5" diameter aluminum rod, and manipulation of the object operated a micro-switch mounted at the opposite end of the rod. Each monkey used its right hand to perform the movements and the left upper extremity was restrained within the primate chair.

Each trial included four cognitive states marked by three visual cues during the experiments (Fig. 2-2). The baseline began with each monkey holding the home object, which was aligned with the monkey's right shoulder, for a random period between 230 and 1130 milliseconds. At the beginning of planning was a visual cue indicated by a blue LED next to one of the four peripheral objects, which instructed the monkeys to release the home object and reach towards the instructed peripheral object. The onset of movement (OM) was marked as the moment the monkeys released the home object. During the movement state, as soon as a monkey touched the instructed peripheral object, the monkeys were required to grasp and manipulate the instructed object: either rotate the sphere object 45º, pull the mallet object, depress the pushbutton object, or

to pull a peripheral mounted cylinder object against its own small spring load [12]. The monkeys were required to hold the instructed target object for 1000 milliseconds during the holding state.

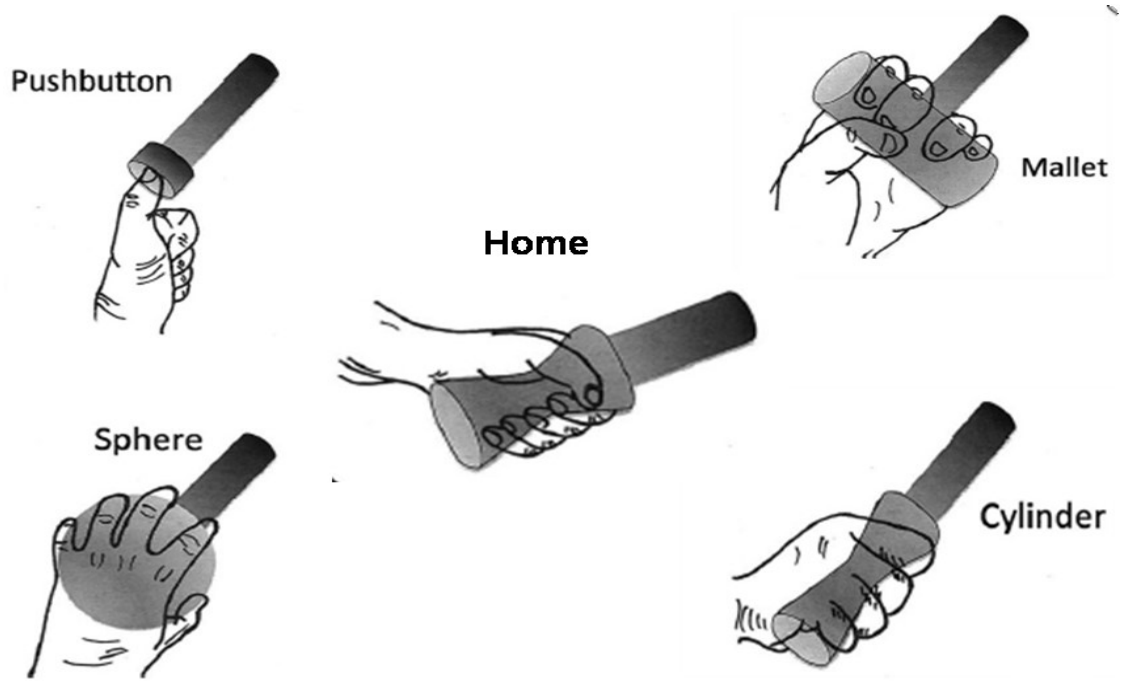


Fig. 2-1 Experimental set-up and task apparatus. The center is the home object and the four objects around the home objects are four possible target objects.

Fig. 2-2 The timeline of an experimental trial including the four cognitive states marked with average durations. The detection windows is in the size of 2*W (W=50 milliseconds) for each cognitive state transition.

2.2 Microelectrode Array Implantation

All studies were carried out under a protocol approved by Dr. Marc Schieber at the University of Rochester Institutional Animal Care and Use Committee. Each of the two monkey subjects was implanted with multiple floating microelectrode arrays (FMAs; MicroProbes for Life Sciences) in cortical motor areas (in the left hemisphere) using sterile techniques and isoflurane anesthesia. In order to sample neural activity at different depths down the anterior bank of the central sulcus, each of the FMAs we used incorporated

electrodes of various lengths (Because the length of each electrode on a FMA can be specified from 1 mm to 10 mm at the time of manufacture). Each FMA consisted of 16 parylene-C insulated platinum/iridium recording microelectrodes of different lengths (FMAs, MicroProbes, Gaithersburg, MD), varying from 1.5 to 8.0 mm in monkey Y and from 1.0 to 6.0 mm in monkey X, arranged in a 4×4 triangular matrix on a 1.95×2.45 mm ceramic chip. Two additional low impedance microelectrodes on each array served as reference and ground electrodes. After craniotomy and durotomy, each FMA was advanced slowed into the cortex at a location selected based on direct visualization of the hemispheric surface (Figs. 2-3).

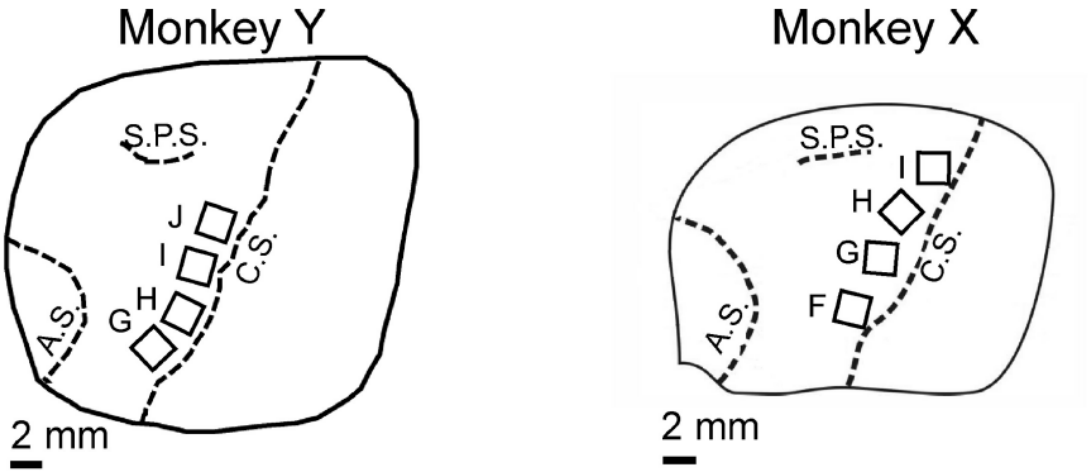


Fig. 2-3 Recording locations. The drawings were traced from intraoperative photographs, showing the locations of FMAs implanted in two monkey subjects.

After all arrays had been implanted, the dura matter was closely loosely and covered with Duragen (Integra), after which the craniotomy was closed with methylmethacrylate. Array connectors were imbedded in additional methylmethacrylate, and a polycarbonate chamber was mounted over the array connectors. The entire implant was fixed to the skull with circumferentially placed titanium bone screws also embedded in methylmethacrylate, along with head-holding post. Postoperatively, each money received a 3 d course of banamine ($1.1 mg\cdot kg^{-1}\cdot d^{-1}$) for pain and a 2-6 weeks course of ceftriaxone ($50 mg\cdot kg^{-1}\cdot d^{-1}$) for infection prophylaxis and was maintained for several weeks on phenytoin ($10 mg\cdot kg^{-1}\cdot d^{-1}$) for seizure prophylaxis. After a recovery period of at least 1 week, each monkey returned to performing the behavioral task described above in daily sessions, now with the head fixed.

2.3 Acquisition of Neurophysiological Signals

All studies were carried out under a protocol approved by Dr. Marc Schieber at the University of Rochester Institutional Animal Care and Use Committee. Single-unit neuronal activity was recorded using a Plexon (Dallas, TX) data acquisition system from 2 floating microelectrode arrays (FMAs) in Monkey #1: one in PMd and one in PMv, respectively, as well as 4 FMAs in Monkey #2: two in PMd, and two in PMv, respectively. There were two days of neuronal recordings from each monkey. There were totally 148 PMv neurons, and 304 PMd respectively. Each FMA included 16 electrodes and up to four single-units could be discriminated per electrode.

Signals from the microelectrodes (impedance, ~0.5 $M\Omega$) were amplified $20\times$ by a head stage and then hardware filtered for spikes (100 Hz to 8 kHz). Neuronal spiking activity from each of the 64 recording electrodes was amplified from 1000 to 32000 times. Waveforms crossing a threshold selected online by an investigator were sampled at 40 kHz and saved for offline sorting performed with Offline Sorter by Plexon. After offline sorting, spike clusters with a waveform signal-to-noise ratio (SNR) > 3.0 and with no inter-spike intervals (ISIs) of 1 ms or less were considered well-isolated single-unit (SU) recordings, whereas spike clusters with SNR <=3 or ISIs <=1 ms were considered multiunit (MU) recordings.

Chapter 3 Hidden Markov Models

3.1 Statistical Markov Models

3.2 Emitting Statistical Markov Models

3.3 Hidden Markov Models (HMMs)

3.4 Inference in a HMM

3.4.1 Inferring the distribution of $S_{1:N}$

3.4.1.1 Forward Pass

3.4.1.2 Backward Pass

3.4.2 Inferring the most probable $S_{1:N}$

3.5 Parameter Estimation in HMMs

3.6 Chapter Summary

3.1 Hidden Markov Models

A Hidden Markov Model (HMM) is one form of probabilistic graphic models in which the observations are modeled as outputs of a discrete state switching process. The observations are modeled using a vocabulary base or a support set. Each discrete state has a characteristic pattern of transitioning to other states and emitting observations with various probabilities. Inference and learning real world problems using HMMs have been most popular in the speech recognition field for the last several decades. Recently, research in the biomedical engineering field has begun to use HMMs to help with system identification and classification. This chapter is meant to introduce the technical basics of HMMs. In this chapter, commonly used algorithms in HMMs are introduced and derived in detail, establishing the mathematical symbols and formalism to be used for later chapters.

3.2 Statistical Markov Models

A probabilistic Markov Model is one kind of a (finite) state machine or finite state automaton. A state machine is a computational mathematical model used to design computer programs and logic circuits. An abstract state machine can have finite number of states, and at every time point, the state machine is in one and only one state, which is usually called the current state. The current state of the state machine can change from one to another triggered by various events or conditions. Such a switch between states is called a transition. A finite state machine is defined by the list of the states within it, and the triggering conditions for each transition (Fig. 3-1).

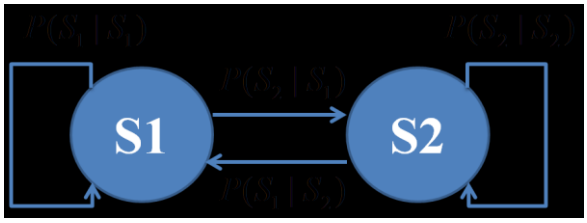


Fig. 3-1 An example of a simple two-state state machine with probabilistic transitions between states.

A probabilistic Markov model is a state machine that generates a discrete state sequence with a unique start state and end state. The state sequence can be denoted as $S_{0:N+1}$, where S_0 is the start state, S_{N+1} is the unique end state, and $S_t \in \{1,...,C\}$ for $1 \le t \le N$. Each state sequence can be generated with a probability, which can be denoted as

$$p(S_{1:N}) = \prod_{t=0}^{N} p_t(S_{t+1}|S_t) \tag{3.1}$$

When a Markov chain is homogeneous, the time index can be ignored and the probability of observing a discrete state sequence $S_{1:N}$ following the Markov property and is given as

$$p_t(S_{t+1}|S_t) = q_{s_t s_{t+1}} \tag{3.2}$$

$$p(S_{1:N}) = \prod_{i=0}^{N} q_{S_t S_{t+1}} \tag{3.3}$$

where $q_{s',s}$ is a transition probability strictly following all the canonical properties of a probability distribution. Since a non-homogeneous Markov chain can be transformed into a homogeneous one by

appending the time-index to each discrete state, we only deal with homogeneous Markov chains without loss of generality. The system states of a Markov chain are fully observable.

3.3 Emitting Statistical Markov Models

Besides the transitions between system states, a more complicated type of statistical Markov models is one that has emitting states. An emitting state is a discrete state $S \in \{1,....,C\}$ that emits an output observation $o \in \Theta$, where Θ is a set, following a strict probabilistic distribution over the set O. The probabilistic emitting distribution can be denoted by $P(o \mid S)$. The set elements can be discrete e.g. $\mathrm{O} = \{1,....,K\}$ or continuous e.g. $\mathrm{O} = R^D$. Formally, an emitting statistical Markov model is a weighted graph $G(V,E)$ where each state is a vertex v in the graph. For each vertex denoted by $v \in V - \{START, END\}$ in the graph is associated with a probability distribution on the output set O. A homogeneous emitting statistical Markov model generates both a discrete state sequence $S_{1:N}$ and $O_{1:N}$ simultaneously. The joint probability of observing both the state sequence and the output sequence can be denoted as:

$$P(S_{1:N}, O_{1:N}) = P(O_{1:N}|S_{1:N})P(S_{0:N+1}) = \left(\prod_{t=0}^{N-1} q_{S_t S_{t+1}} p(O_{t+1}|S_{t+1})\right) q_{S_N, END} \tag{3.4}$$

3.4 Hidden Markov Model

When the discrete state sequence $S_{1:N}$ is partially or completely unobservable, an emitting statistical Markov model becomes a hidden Markov model. The probability of observing an output sequence can be computed by marginalizing over all possible state sequence $S_{1:N}$ weighted by the probability of each state sequence as

$$P(o_{1:N}) = \sum_{S_{1:N}} P(S_{1:N}, o_{1:N}) = \sum_{S_{1:N}} q_{S_{N,END}} \prod_{t=0}^{N} q_{S_t, S_{t+1}} P(o_{t+1} \mid S_{t+1}) \tag{3.5}$$

A HMM is composed of the following components:

V: a vertex set $\{1,...,C,START,END\}$.

$q_{s',s}$ A transition probability matrix of size $(C+2)\times(C+2)$.

O: An output set defining all possible observations.

$P(o \mid S)$: an emitting probability distribution over O for each state $s \in V$ in the HMM.

Here we denote the set of the variables to describe the parameters of a HMM as $\Theta = \{q_{s',s}, P(o \mid S)\}$. For example, if $P(o \mid S)$ follows a Gaussian distribution, Θ includes the set of means and covariance associated with each state denoted as

$$\Theta = \{q_{s',s}, P(o \mid S)\} \tag{3.6}$$

where the upper script s indicates the state dependency of the parameters to states.

3.5 Inference in a HMM

The inference problems in a HMM can be summarized as estimating the values of hidden variables from known variables including output observations. The variables in an HMM are $S_{1:N}$ and $o_{1:N}$.

There are two common inference problems in a statistical graphical model as follows.

1. Computing the conditional probabilistic distribution of a hidden state sequence over the observations, i.e., estimating $P(S_{1:N} \mid o_{1:N})$;

2. Computing the most likely hidden state sequence, i.e., $\hat{S}_{1:N} = \arg\max_{S_{1:N}} (S_{1:N} \mid o_{1:N})$.

The first inference problem often uses techniques such as Expectation Maximization (EM). The second inference problem is to find the most fit set of values for the hidden random variables under the constraints of the observations. For example, recognizing the most likely sequence of words given the recorded acoustics, or recognizing the most likely sequence of surgical movements given the recorded kinematics. Both inference problems have been well studied with popular solutions and will be described briefly below.

3.5.1 Inferring the Probability Distribution of $S_{1:N}$

In order to infer the probabilistic distribution of $S_{1:N}$, formally, we need to calculate the conditional joint probability distribution of the state sequence,

$$P(S_{1:N} \mid o_{1:N}, \varsigma) = \prod_{t=0}^{N} P(S_{t+1} \mid S_t, o_{1:N}, \varsigma) \tag{3.7}$$

where the Markov property defines that given the current state, the future and past states are independent. Formally,

$$P(S_{N+1} = s \mid S_1 = s_1, S_2 = s_2, ..., S_N = s_N) = P(S_{N+1} = s \mid S_N = s_N). \tag{3.8}$$

Here the grammar ς is defined by a graph G on which the inference is made. During inference process, not all states are active at the same time given the times of the observations. For example, in recognizing a sentence containing a word dive : = / d / ai / v / and the time stamps for the corresponding acoustic sound are between 3 sec and 4.5 sec, then only the states corresponding to the phonemes d, ai, v should be active between 3 and 4.5 sec. A boolean b_t^s indicator can be set up for each state. For the states that can't be active during certain period, the corresponding $b_t^s = 0$. In cases where the time information is not available, $b_t^s = 1$ for all states and all time.

The joint distribution of S_t, S_{t+1} and $o_{1:N}$ for each time *t* is sufficient for estimating the probability distribution of a state sequence given the grammar and the observations. It can be denoted as

$$\gamma_t^{s,s'} = P(S_t = s, S_{t+1} = s', o_{1:N}). \tag{3.9}$$

It can be re-written using the chain rule and the Markov property as follows:

$$
\begin{aligned}
&\gamma_t^{s,s'} = P\left(S_t = s, S_{t+1} = s', o_{1:N}\right) \\
&= P\left(S_t = s, S_{t+1} = s', o_{1:t}, o_{t+1:N}\right) \\
&= P\left(S_t = s, o_{1:t}\right) P\left(S_{t+1} = s', o_{t+1:N} \mid S_t = s, o_{1:t}\right) \\
&= P\left(S_t = s, o_{1:t}\right) P\left(S_{t+1} = s' \mid S_t = s, o_{1:t}\right) P\left(o_{t+1:N} \mid S_{t+1} = s', S_t = s, o_{1:t}\right) \\
&= P\left(S_t = s, o_{1:t}\right) q_{s,s'} P\left(o_{t+1:N} \mid S_{t+1} = s'\right) \\
&= \alpha_t^s \beta_{t+1}^{s'}
\end{aligned}
\tag{3.10}
$$

where

$$
\begin{aligned}
&\alpha_t^s = P\left(S_t = s, o_{1:t}\right) \\
&\beta_{t+1}^{s'} = P\left(o_{t+1:N} \mid S_{t+1} = s'\right)
\end{aligned}
\tag{3.11}
$$

The computation of the joint probability distribution requires the computation of a forward probability denoted as α_t^s and a backward probability denoted as $\beta_{t+1}^{s'}$. Both the forward and backward probabilities can be calculated recursively using dynamic programming as in the forward-backward algorithm. Below will be a brief description of the forward-backward algorithm.

3.5.1.1 Forward Pass

The forward pass is used to calculate the forward probability $\alpha_t^s = P\left(S_t = s, o_{1:t}\right)$. Using the law of total probability by marginalizing over all $S_{t-1} = s'$,

$$
\begin{aligned}
&\alpha_t^s = P\left(S_t = s, o_{1:t}\right) \\
&= \sum_{s'} P\left(S_t = s, S_{t-1} = s', o_{1:t}\right) \\
&= \sum_{s'} P\left(S_t = s, S_{t-1} = s', o_{1:t-1}, o_t\right) \\
&= \sum_{s'} P\left(S_{t-1} = s', o_{1:t-1}\right) P\left(S_t = s \mid S_{t-1} = s', o_{1:t-1}\right) P\left(o_t \mid S_t = s, S_{t-1} = s', o_{1:t-1}\right) \\
&= \sum_{s'} P\left(S_{t-1} = s', o_{1:t-1}\right) P\left(S_t = s \mid S_{t-1} = s'\right) P\left(o_t \mid S_t = s\right) \\
&= \sum_{s':(s',s)\in E(G)} \alpha_{t-1}^{s'} q_{s',s} P_\Theta\left(o_t \mid s\right)
\end{aligned}
\tag{3.12}
$$

The base case for t = 0 is $\alpha_0^{START} = 1$ and $\alpha_0^s = 0$ for all other states. Using the equation above, one can iteratively calculate α_t^s for each state S for t from 1 to *N*.

3.5.1.2 Backward Pass

Similar to the computation of the forward probability, the backward probability can be computed using the law of total probability by marginalizing over all future states $S_{t+1} = s'$ as follows:

$$\begin{aligned}
&\beta_t^s = P\left(o_{t:N} \mid S_t = s\right) \\
&= P\left(o_t, o_{t+1:N} \mid S_t = s\right) \\
&= P\left(o_t \mid S_t = s\right) P\left(o_{t+1:N} \mid o_t, S_t = s\right) \\
&= P\left(o_t \mid S_t = s\right) \sum_{s':(s,s')\in E(G)} P\left(o_{t+1:N}, S_{t+1} = s' \mid o_t, S_t = s\right) \\
&= P\left(o_t \mid S_t = s\right) \sum_{s':(s,s')\in E(G)} P\left(S_{t+1} = s' \mid o_t, S_t = s\right) P\left(o_{t+1:N} \mid S_{t+1} = s', o_t, S_t = s\right) \\
&= P\left(o_t \mid S_t = s\right) \sum_{s':(s,s')\in E(G)} P\left(S_{t+1} = s' \mid S_t = s\right) P\left(o_{t+1:N} \mid S_{t+1} = s'\right) \\
&= P\left(o_t \mid S_t = s\right) \sum_{s':(s,s')\in E(G)} q_{s,s'} \beta_{t+1}^{s'}
\end{aligned} \tag{3.13}$$

The base case is $\beta_{N+1}^{END} = 1$ and $\beta_{N+1}^{s} = 0$ for all other states. The backward probability can be computed recursively for all times going from backward N to 1.

The forward-backward procedure for a HMM is summarized in Algorithm 2. The inputs of the algorithm include a grammar ς, the parameter set Θ, and the observation sequence $o_{1:N}$. The outputs of the algorithm is the pairwise joint probability of states given observations $\gamma_t^{s,s'}$ for $t \geq 1$, and the marginal probability $\gamma_t^s = P\left(S_t = s, o_{1:N}\right)$. The posterior probabilities $\hat{\gamma}_t^s = P\left(S_t = s \mid o_{1:N}\right)$ can be further computed as follows:

$$\hat{\gamma}_t^s = P\left(S_t = s \mid o_{1:N}\right) = \frac{P\left(S_t = s \mid o_{1:N}\right)}{\sum_s P\left(S_t = s \mid o_{1:N}\right)} \tag{3.14}$$

Algorithm 1: $\log add\left(x, y\right)$

Note: Given $\log\left(x\right) = lx$, $\log\left(y\right) = ly$, find $\log\left(x + y\right)$

$$\begin{aligned}
&\log\left(x + y\right) \\
&= \log\left(x * \left(1 + y / x\right)\right) \\
&= \log\left(x\right) + \log\left(1 + \exp\left(\log\left(y\right) - \log\left(x\right)\right)\right) \\
&= lx + \log\left(1 + \exp\left(ly - lx\right)\right)
\end{aligned}$$

Algorithm 2: HMM Forward-Backward $\left(G, \left[b_{1:N}^{1:C}\right], \Theta, o_{1:N}\right)$

Note: both the forward and backward probabilities are in log scale.

$\alpha_0^s \leftarrow -\infty \forall s;$
$\alpha_0^{START} \leftarrow 0;$

for $t \leftarrow 1$ **to** N **do**

$\alpha_t^s \leftarrow -\infty \forall s;$

for $(s', s) \in E(G)$ **do**

if $b_t^s == true$ **then**

$$\alpha_t^s \leftarrow \log add\left(\alpha_t^s, \alpha_{t-1}^{s'} + \log q_{s',s} + \log P_\Theta\left(o_t \mid s\right)\right);$$

end

end

$\beta_{N+1}^s \leftarrow -\infty \forall s;$
$\beta_{N+1}^{END} \leftarrow 0;$

for $t \leftarrow N$ **to 1 do**

$\beta_t^s \leftarrow -\infty \forall s;$
$\gamma_t^s \leftarrow -\infty \forall s;$

for $(s, s') \in E(G)$ **do**

$$\beta_t^s \leftarrow \log add\left(\beta_t^s, \log P_\Theta\left(o_t \mid s\right) + \log q_{s,s'} + \beta_{t-1}^{s'}\right);$$

$$\gamma_t^{s,s'} \leftarrow \alpha_t^s + \log q_{s,s'} + \beta_{t+1}^{s'};$$

$$\gamma_t^s \leftarrow \log add\left(\gamma_t^s, \gamma_t^{s,s'}\right);$$

end

end

end

3.5.2 Inferring the Most Probable $S_{1:N}$

The problem of computing the most likely state sequence $\hat{S}_{1:N}$ given $o_{1:N}$, denoted as $\hat{S}_{1:N} = \arg\max_{S_{1:N}} P(S_{1:N} \mid o_{1:N}) = \arg\max_{S_{1:N}} P(S_{1:N}, o_{1:N})$ can be solved using standard dynamic programming referred as the Viterbi algorithm. The Viterbi algorithm calculates $\hat{\alpha}_t^s$ denoted as

$$\hat{\alpha}_t^s = \max_{S_{1:t-1}} P(S_{1:t-1}, S_t = s, o_{1:t}) \tag{3.15}$$

This can be done recursively in a similar fashion to Algorithm 2 using the chain rule and Markov property as follows:

$$\begin{aligned}
\hat{\alpha}_t^s &= \max_{S_{1:t-1}} P(S_{1:t-1}, S_t = s, o_{1:t}) \\
&= \max_{s'} \left[\max_{S_{1:t-2}} P(S_{1:t-2}, S_{t-1} = s', S_t = s, o_{1:t}) \right] \\
&= \max_{s'} \left[\max_{S_{1:t-2}} P(S_{1:t-2}, S_{t-1} = s', S_t = s, o_{1:t-1}, o_t) \right] \\
&= \max_{s'} \left[\max_{S_{1:t-2}} P(S_{1:t-2}, S_{t-1} = s', o_{1:t-1}) q_{s',s} P_\Theta(o_t \mid s) \right] \\
&= P_\Theta(o_t \mid s) \left[\max_{s'} q_{s',s} \max_{S_{1:t-2}} P(S_{1:t-2}, S_{t-1} = s', o_{1:t-1}) \right] \\
&= P_\Theta(o_t \mid s) \left[\max_{s'} q_{s',s} \hat{\alpha}_{t-1}^{s'} \right]
\end{aligned} \tag{3.16}$$

Moreover, for each state s, we may keep track of the previous state s' that attains the maximum of the function above as

$$\hat{P}_t^s = \arg\max_{s'} q_{s',s} \hat{\alpha}_{t-1}^s \tag{3.17}$$

When $t = N+1$, $\alpha_{N+1}^{END} = \max_{S_{1:N}} P(S_{1:N}, o_{1:N})$. (3.18)

Thus, the most likely state sequence $\hat{S}_{1:N}$ can be computed by backtracking the best previous state starting from END state at $N+1$. The detailed algorithm is shown in Algorithm 3.

Algorithm 3: Vitebi $(G, \Theta, o_{1:N})$

$\hat{\alpha}_0^s \leftarrow -\infty \forall s;$

$\hat{\alpha}_0^{START} \leftarrow 0;$

for $t \leftarrow 1$ **to** N **do**

$\alpha_t^s \leftarrow -\infty \forall s;$

for $(s', s) \in E(G)$ **do**

$\ell \quad \hat{} + \log q_{s',s} + \log P_\Theta(o_t \mid s);$

if $\ell \quad \hat{}$ **then**

$\hat{\alpha}_t^s \leftarrow \ell$

$\hat{P}_t^s \leftarrow s';$

end

end

$\hat{S}_{N+1} \leftarrow END$

for $t \leftarrow N$ **to 1 do**

$\hat{S}_t \leftarrow \hat{P}_{t+1}^{\hat{S}_{t+1}}$

end

return $\hat{S}_{1:N}$

end

3.6 Parameter Estimation for HMM; the Baum Welch Algorithm

The parameter estimation process is to find the set of parameters $\hat{\Theta}_{ML}$ under which the likelihood of the observed data will be optimized. Formally, assuming we are given M

sequences of observations $\left[o^j_{1:N_j} \right]_{j=1}^{M}$, each observed sequence has a grammar ς_j including an unweighted graph G_j of emitting states which generated $o^j_{1:N_j}$. All possible state-sequences $S_{1:N_j}$ must belong to the grammar ς_j.

For an observation sequence $o^j_{1:N_j}$, its likelihood can be denoted as

$$\ell = \log\left(\sum_{S_{1:N_j} \in \zeta_j} q_{S_{N_j,END}} \prod_{t=1}^{N_j} q_{s_{t-1},S_t} P_\Theta\left(o^j_t \mid S_t\right) \right) \tag{3.19}$$

The optimal set of parameters $\hat{\Theta}_{ML}$ is defined as follows:

$$\hat{\Theta}_{ML} = \arg\max_{\Theta} \sum_{j=1}^{M} \ell \tag{3.20}$$

Because the total likelihood of the observed data sequences is a non-convex function in Θ. Directly solving the problem requires various heuristics such as simulated annealing.

An alternative approach is to calculate the complete data likelihood of $\left\{ S_{1:N_j}, o^j_{1:N_j} \right\}$, assuming the state sequence $S_{1:N_j}$ for the j^{th} was given. The complete data likelihood for $\left\{ S_{1:N_j}, o^j_{1:N_j} \right\}$ is

$$\ell\left(o^j_{1:N_j} \right) = \log\left(q_{S_{N,END}} \prod_{t=1}^{N_j} q_{S_{t-1},S_t} P_\Theta\left(o^j_t \mid S_t\right) \right) \tag{3.21}$$

Since $S_{1:N_j}$ is unknown, the complete log-likelihood above is a random variable. In order to remove the randomness, the expectation of $\ell\left(o^j_{1:N_j} \right)$ is calculated under the posteriori distribution $P_{\Theta_c}\left(S_{1:N_j} \mid o^j_{1:N_j}, \zeta_j \right)$ where Θ_c is the current set of HMM parameters. This method is called Expectation-Maximization (EM). This step is the E-step of the EM procedure.

$$E_{\Theta_c}\left[\ell \qquad \right] = E_{\Theta_c}\left[\log\left(q_{S_{N_j},END}\prod_{t=1}^{N_j} q_{S_{t-1},S_t} P_\Theta\left(o_t^j \mid S_t\right)\right)\right]$$
$$= \underbrace{\sum^{N_j} E_\Theta\left[\log q_{s,s'}\right]}_{(1)} + \underbrace{\sum^{N_j} E_\Theta\left[P_\Theta\left(o_t^j \mid S_t\right)\right]}_{(2)} \tag{3.22}$$

The first term can be simplified further as follows:

$$E_{\Theta_c}\left[\log q_{S_t,S_{t+1}}\right] = \sum_{s,s'} P\left(S_t = s, S_{t+1} = s' \mid o_{1:N_j}^j\right)\log q_{s,s'} = \sum_{s,s'} \hat{\gamma}_{t,j}^{s,s'} \log q_{s,s'} \tag{3.23}$$

Similarly the second term can be written as

$$E_{\Theta_c}\left[P_\Theta\left(o_t^j \mid S_t\right)\right] = \sum_s \gamma_{t,j}^s \log P_\Theta\left(o_t^j \mid s\right) \tag{3.24}$$

The quantities $\gamma_{t,j}^{s,s'}$ and $\gamma_{t,j}^s$ can be calculated with Algorithm 2.

The M-step of the EM procedure computes $\hat{\Theta}$ such that

$$\hat{\Theta} = \arg\max \sum_{j=1}^{M} E_{\Theta_c}\left[\ell \qquad \right]. \tag{3.25}$$

Furthermore, $\hat{\Theta}$ satisfies

$$\sum_{j=1}^{M} \ell \qquad \qquad \sum_{j=1}^{M} \ell \qquad . \tag{3.26}$$

In other words, the iterative procedures in EM always improve the log-likelihood over the current set of parameters. The estimation process is referred as the Baum Welch algorithm.

The optimal set of parameters can be defined now as follows:

$$\hat{\Theta} = \arg\max\left[\sum_{s,s'}\sum_{t,j} \gamma_{t,j}^{s,s'} \log q_{s,s'} + \sum_{s}\sum_{t,j} \gamma_{t,j}^{s} \log P_\Theta\left(o_t^j \mid s\right)\right] \tag{3.27}$$

The optimization over $\hat{q}_{s,s'}$ can be done via optimizing the first summation under the constraint that $\sum_{s'} q_{s,s'} = 1$. The resulting optimal $\hat{q}_{s,s'}$ can be denoted as follows:

$$\hat{q}_{s,s'} = \frac{\sum_{t,j} \hat{\gamma}_{t,j}^{s,s'}}{\sum_{s''} \sum_{t,j} \hat{\gamma}_{t,j}^{s,s''}} \tag{3.28}$$

When one assumes $P_{\Theta}(o \mid s) = P_{\Theta^s}(o)$ where Θ^s is a state specific parameter that determines the probability distribution of the observations, one may calculate the optimal $\hat{\Theta}^s$ separately for each s from optimizing over the second term of the equation above and get the solutions as follows:

$$\hat{\Theta}^s = \arg\max_{\Theta^s} \sum_{t,j} \hat{\gamma}_{t,j}^{s} \log P_{\Theta^s}\left(o_t^j\right) \tag{3.29}$$

A closed form of the solution above is possible when $P_{\Theta}(\bullet)$ follows certain probability distribution. One commonly used distribution is the Gaussian density. Gaussian densities can be expressed with only a few parameters and also are powerful enough to easily approximate a large class of distributions. Formally, when $P_{\Theta^s}(o) = \mathrm{N}\left(\mu_s \quad \Sigma^s\right)$, the state-dependent optimal HMM parameter can be expressed as follows:

$$\hat{\Theta}^s = \arg\max_{\Theta^s} \sum_{t,j} \hat{\gamma}_{t,j}^{s} \left(-\frac{1}{2} \log\left|\Sigma^s\right| - Trace\left[\Sigma^{s-1}\left(o_t^j - \mu^s\right)\left(o_t^j - \mu^s\right)^T \right] \right) \tag{3.30}$$

EM improves the likelihood at each step, and one needs to repeatedly perform the updates mentioned above to iteratively increase the log-likelihood reaching a plateau or reach the error range pre-specified. The final estimate $\hat{\Theta}^s = \left\{ \hat{\mu}^s \quad \hat{\Sigma}^s \right\}$ can be calculated using matrix operations and denoted as follows:

$$\begin{aligned} N_s &= \sum_{t,j} \hat{\gamma}_{t,j}^{s} \\ \hat{\mu}^s &= \frac{1}{N_s} \sum_{t,j} \hat{\gamma}_{t,j}^{s} \, o_t^j \\ \hat{\Sigma}^s &= \frac{1}{N_s} \sum_{t,j} \hat{\gamma}_{t,j}^{s} \, o_t^j o_t^{jT} - \hat{\mu}^s \, \hat{\mu}^{sT} \end{aligned} \tag{3.31}$$

3.7 Chapter Summary

The basic concepts of HMMs are developed from the first principles. Commonly used inference and learning algorithms for HMMs are reviewed. Maximum likelihood estimates of HMM parameters can be obtained using the Baum Welch Algorithm utilizing dynamic programming. In a similar fashion, the optimal state sequence given observations can also be solved using the Viterbi algorithm.

Chapter 4 Hidden Markov Model for Decoding Cognitive States and Kinematics in Primate Reach-to-Grasp Tasks Using Premotor Cortical Ensembles

4.1 Introduction

4.2 Methods

4.2.1 Experimental Setup

4.2.2 Framework Overview

4.2.3 Preprocessing

4.2.4 Cognitive State Transition Detection

4.2.5 Detection Accuracy Measurement

4.2.6 Conversion to Joint Angles from Positional Kinematics

4.2.7 State-Dependent Decoding of Kinematics

4.2.8 Virtual Worlds in MSMS

4.3 Results

4.3.1 Dimension Reduction vs. Drop Out Analysis

4.3.2 Cognitive State Transition Detection Accuracy

4.3.3 Cognitive State Transition Detection Latency

4.3.4 Kinematics Decoding Accuracy

4.4 Discussion and Conclusions

4.4.1 Cognitive State and Kinematics Estimation

4.4.2 Main Contributions

4.4.3 Limitations and Future Directions

4.5 Bibliography

4.1 Introduction

Advanced dexterous prosthetics research has undergone rapid development as a potential solution to upper limb amputation. Neural prosthetics uses signals from cortical neurons to decode motor intent and execution [1–10]. Research summarized in [3], [11–13] has demonstrated that non-human primates such as monkeys are able to learn to gain control of motor prosthetics using ensemble neural activity of motor related cortical areas.

An important problem in the neural prosthetics design is to detect transitions between epochs of neural activity [14], [15] and estimate epoch-dependent neuroprosthetics kinematics which enables prosthetic interfaces to operate autonomously. Epochs of neural activity, or, cognitive states, are the changing internal states that govern the process from movement generation to movement completion, and are distinguished by the changing dynamics of neural activity. For example, in an instructed reach-to-grasp behavioral task, the epochs of neural activity may coincide with task trial epochs such as a baseline epoch, a planning epoch, and a movement epoch, as well as a holding epoch. It is essential to estimate these different cognitive states during the control of neural prosthetics because users may employ different control policies in these different epochs, and thus the inherent dynamics of the corresponding neural activity may be different. A well-designed clinically viable neural prosthetic control unit may use different decoding strategies to adapt to different cognitive states for optimal performance in practical scenarios [16].

The prosthetic interfaces described in [2], [4], [6], [9], [10], [17–19] ignored the cognitive states in the experiments. These systems must be manually turned on when the prosthetic is supposed to move and turned off when the prosthetic is not supposed to move so that undesired movements are avoided.

Significant scientific advances have been made, primarily in kinematics decoding and to some extent cognitive state decoding; however, a complete decoding system for both cognitive states and epoch-based kinematics has not been fully developed. Several research groups have tried to address the estimation problem of detecting changing cognitive states from neural activity [14–16], [20–24]. The work of Shenoy et al [14] developed a finite state machine model using an off-line, ad hoc approach and classified neural activity from the Parietal Reach Region (PRR) into three discrete epochs including a baseline epoch, a plan epoch and a reach epoch. They demonstrated that a supervised decoder could improve overall system decoding accuracy.

Achtman et al [15] used neural activity to estimate a planning epoch. Hudson et al [16] presented a supervised discrete state neural decoder and demonstrated its effectiveness in decoding two epochs including a baseline epoch and a reach epoch, using the neural signals from the PRR of a rhesus monkey. Srinivasan et al [20] presented a hybrid framework for cognitive state estimation and demonstrated its application using ensemble spiking activity from point process models of primary motor cortex (M1). Kemere et al [22] used a supervised HMM approach and achieved high decoding performance in estimating three epochs including a baseline epoch, a planning epoch and a movement epoch, using the neural signals from PMd and M1. Abeles et al [23] studied single-unit activity from the frontal areas around the upper limb of the arcuate sulcus using a HMM method to segment cognitive state sequences following a known external stimulus. Seidemann et al [24] used the HMM method to detect the cognitive states in the frontal cortex of a monkey. Wu et al [21] demonstrated a recursive Bayesian estimate procedure based on a Kalman filter.

The decoding framework proposed here aims to address the problem of decoding cognitive states and kinematics. The criteria we used to judge the performance of such a decoding framework are as follows. First, the decoding system needs to be able to estimate the cognitive states and epoch transitions with at least comparable decoding accuracies and latencies to previous research. In addition, the decoding framework needs to be able to estimate the corresponding kinematics (in our case wrist, hand and finger joint angles) with at least comparable decoding accuracies to previous research.

In this work, the decoding framework is composed of three parts including a preprocessing unit, a HMM state decoder and a switching linear discriminant system (S-LDS) kinematics decoder. In the preprocessing unit, we demonstrate that the performance of dimension reduction leads to higher decoding accuracies than obtained by drop out analysis. The HMM state decoder uses one or multiple latent state variables to represent an epoch of ensemble neural activity. The transition between epochs is modeled as a Markov process. The firing rates are modeled as a mixture of Gaussians whose parameters are conditioned on the corresponding latent state variables. The decoding framework treats ensemble neural activity as the observations of a generative process conditioned on the latent cognitive states, and uses a supervised approach to learn the correspondence between latent state variables and epochs of ensemble neural activity. The S-LDS kinematics decoder further models the epochs of joint angle kinematics with an epoch-dependent linear system based on the decoded cognitive state sequence information from the HMM state decoder. Similar to the HMM state

decoder, the transition between cognitive states is modeled as a Markov process in the S-LDS kinematics decoder. We demonstrate the decoding framework using neural recordings from ventral premotor (PMv) and dorsal premotor (PMd) neurons of a non-human primate executing four complex reach-to-grasp tasks along with the corresponding kinematics recordings. Using the HMM state decoder, we demonstrate that the transitions between neighboring epochs, regardless of the existence of any external kinematics changes, can be detected with high accuracy and short latencies. We further show that the hand joint angle kinematics can be estimated reliably with high accuracy. In addition, we demonstrate that using multiple latent state variables to model the within-epoch neural activity variability can improve the decoding performance of the HMM state decoder.

4.2 Methods

4.2.1 Experimental Setup

Our collaborators Dr. Marc Schieber trained an adult male rhesus monkey (*Macaca mulatta*) to perform reach-to-grasp tasks involving one center home object and four peripheral target objects, as described in [25]. All studies have been approved by the University of Rochester Institutional Animal Care and Use Committee. The monkey was trained to sit in a primate chair that restrained the neck, torso and legs, and to follow visual cues to manipulate, grasp, and reach towards one of four objects - sphere, perpendicular mounted cylinder (mallet), pushbutton, or peripheral coaxial cylinder (pull handle) - arranged in a planar circle at 60º intervals (Fig. 4-1). Each object was mounted on the end of a 2-3', 0.5" diameter aluminum rod, and manipulation of the object operated a micro-switch mounted at the opposite end of the rod. The monkey used its right hand to perform the tasks.

Each trial is separated into four epochs, including the baseline, planning, moving and holding epochs, marked by five behavioral events during the experiments (Fig. 4-2). The baseline epoch begins with the monkey holding a home cylindrical object (home object), which was aligned with the monkey's right shoulder, for a random period of 230 to 1130 ms. It is followed by the planning epoch, which begins with a cue indicated by a blue LED next to one of the peripheral objects to instruct the monkey to release the home object and reach towards the instructed peripheral object. The beginning of the movement epoch is the onset

of movement (OM). As soon as the monkey grasped the instructed peripheral object, the monkey was required to grasp and manipulate the instructed object: either rotate the sphere object 45º, pull the mallet object, or depress the pushbutton object, or to pull peripheral mounted cylinder object against its own small spring load (Fig. 4-1). The holding epoch starts with the illumination of a green LED next to the instructed object. The monkey held the instructed object in its final position for 1000 ms until a blue LED was turned off to indicate the end of the trial. If the monkey performed the trials correctly, the monkey would get water in reward; if the monkey made errors, such as not completing the trial or grasping the wrong object, the monkey got nothing in reward.

Of the several weeks of data collection, our collaborators Dr. Marc Schiber and his team chose two individual data sets (09_18_09 and 10_02_09). Single-unit activity was recorded using a Plexon (Dallas, TX) data acquisition system from PMd and PMv cortical areas (Fig. 4-3). There were two floating microelectrode arrays (FMAs): one in PMd (array D) and one in PMv (array C). Each FMA includes sixteen electrodes and up to four single-units can be discriminated per electrode. We were able to record the activity of 74 well-isolated neurons in PMv and 70 in PMd. We combined all neurons together during our analysis since our work is not concerned with the individual cortical neurons, yielding a data set with 144 neurons (Fig. 4-4).

The Vicon motion capture system (Vicon Motion Systems, Oxford, UK), used by Dr. Marc Schieber, was composed of 18 cameras and tracked upper-limb kinematics using thirty 3-mm diameter optically-reflective markers (Optitrack, Eugene, OR). There were four markers on the dorsal aspect of the forearm, six on the dorsal surface of the hand, two over the first metacarpal, two between the PIP and distal DIP joints from digits 2 through 5. Details of the marker placement can be found in [18].

The motion capture data was post-processed offline in Vicon's Nexus software. An articulated model of the arm, hand and fingers was used to convert the marker position data from Cartesian space to 18 Euler joint angles of the wrist, thumb, and fingers, as described in [18]. The kinematics and neural data were synchronized using 8-bit behavioral event marker codes generated by the TEMPO (Reflective Computing, St. Louis, MO) behavioral control system.

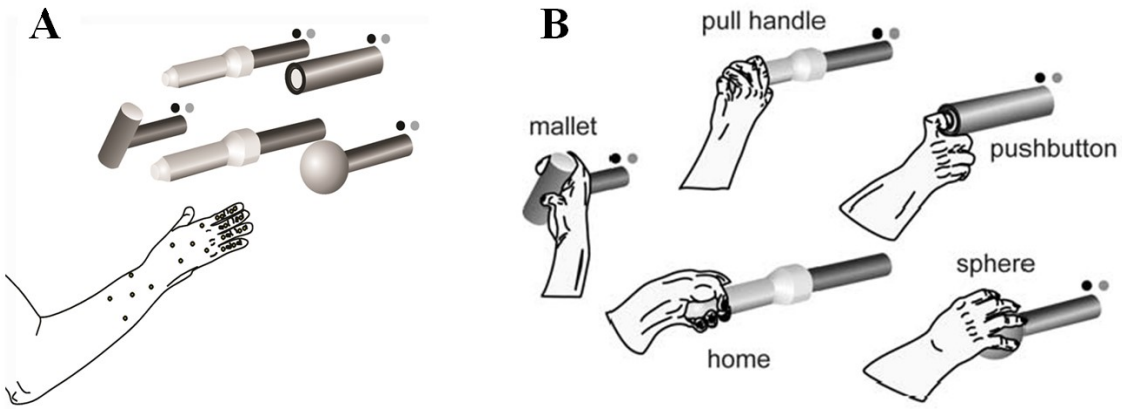


Fig. 4-1 Experimental set-up and task apparatus. (A) The experiment set-up includes a center home object (a cylinder) and four peripheral objects (sphere, mallet, pushbutton, pull handle) as reach-to-grasp task targets. (B) The monkey was instructed with cues to release the home object and reach towards, grasp, and manipulate one of the four peripheral objects arranged in a planar circle. Blue LEDs indicate the target object and green LEDs indicate the beginning of holding period on the target object. Four different hand movement types, namely, pulling a handle, pushing a button, rotating a mallet, and grasping a sphere, are present during the manipulation of different objects.

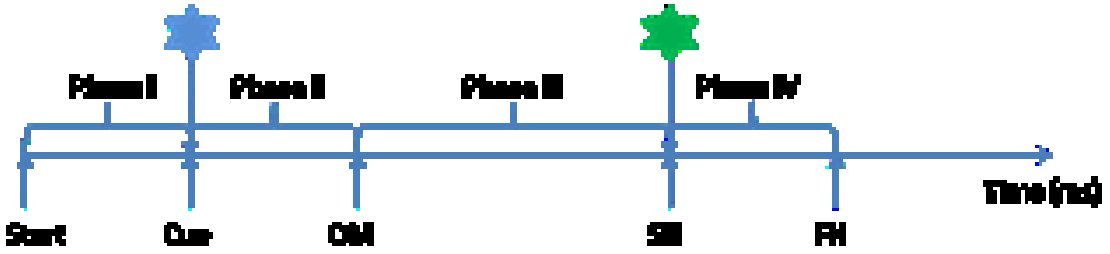


Fig. 4-2 The timeline of an experimental trial. Each trial is composed of four epochs based on behavioral events provided to the monkey during the experiments. Here "start" means the start of the experimental trial; "Cue" indicates the illumination of the blue LED near a target object which is to instruct the monkey to reach and manipulate the corresponding target; "OM" refers to the onset of movement; "SH" indicates the illumination of the green LED which is to instruct the monkey to start holding onto the target object; "FH" refers to the ending signal of the experimental trial, which is turning off the blue LED.

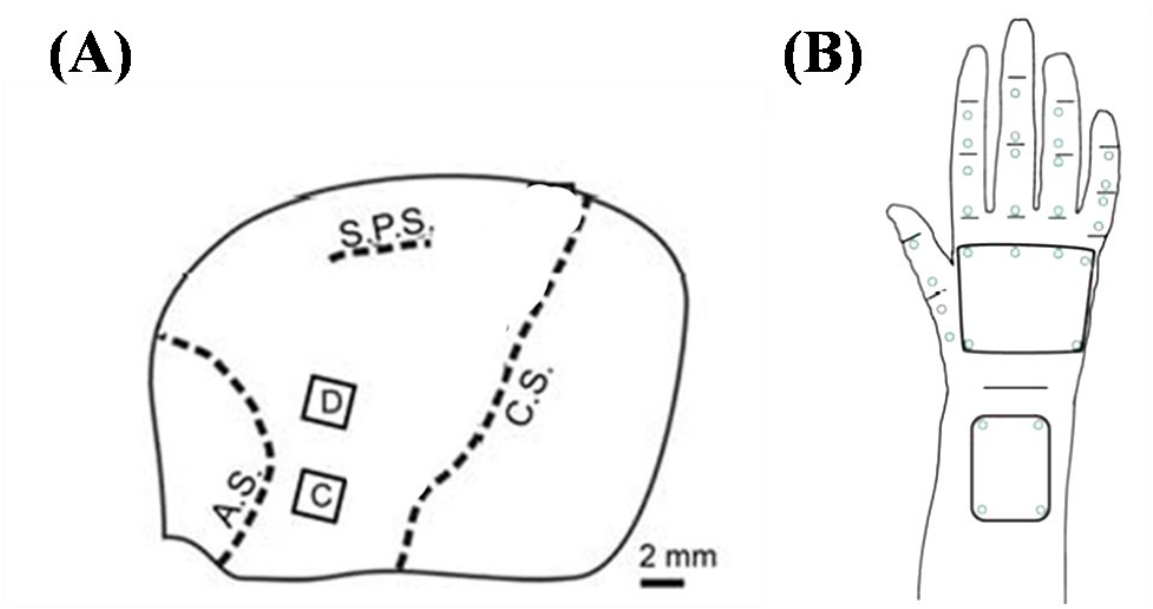
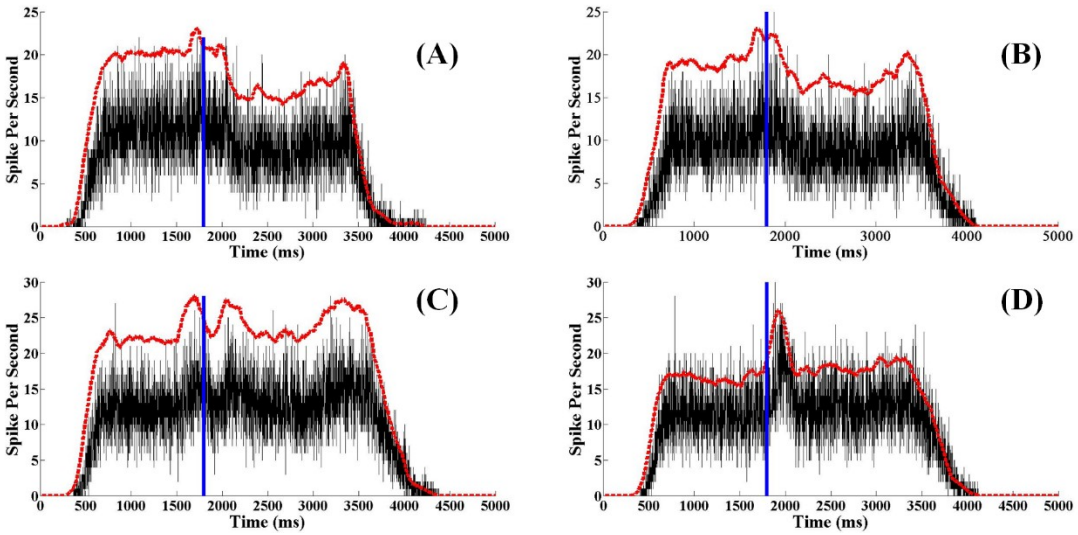


Fig. 4-3 (A) The array locations for the monkey. Spiking activity was recorded from two FMAs, which were implanted in PMv and PMd areas contralateral to the trained hand. Each FMA had 16 spike recording sites. Each recording site can differentiate up to 4 neurons. (B) A Vicon motion capture system was used to track the 3-D position of thirty optically-reflective markers placed on the monkey's right forearm, palm, and fingers. Euler joint angles of the wrist and fingers were calculated using an articulated model of the upper arm.



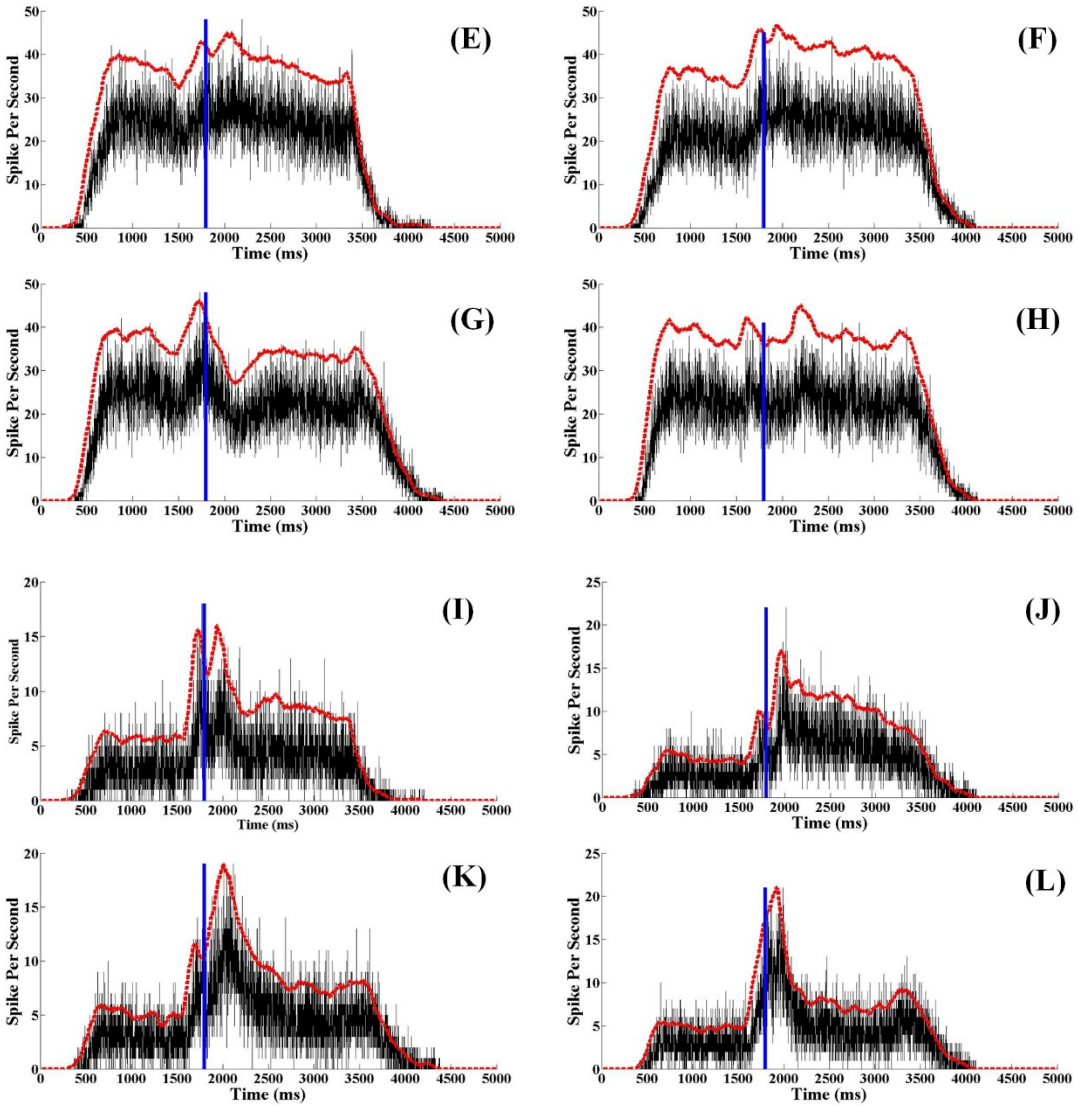


Fig. 4-4 The peristimulus time histograms (PSTHs) and filtered signals from PMd and PMv, as well as M1 for comparison purposes only. (A)-(D) are from PMd, (E)-(H) are from PMv, and (I)-(L) are from one example array in M1. The neural activities are all aligned with the onset of movement (OM), which is represented by a vertical blue line. The neural responses from PMd, PMv and M1 are distinct from each other. In our analysis, we only included PMd and PMv.

4.2.2 Framework Overview

The main skeleton of the decoding framework is shown in Fig. 4-5. The ensemble neural activity recorded from PMv and PMd was first preprocessed, and entered the HMM state decoder for cognitive state estimation. The decoded cognitive state information was passed into the S-LDS kinematics decoder to further

estimate the epoch-based kinematics. The estimated kinematics and cognitive states were compared with the true recordings for measuring the decoding accuracy and estimation latency.

4.2.3 Preprocessing

We considered the fact that the neurons in similar brain regions would most likely be interconnected within complex neural networks, and as such their firing rates would likely is correlated. This might lead to redundant information in the recorded ensemble neural activity as the inputs to the following state and kinematics decoding system. In order to remove the redundant information among the neurons of interest, we used the LDA [26] method to perform dimension reduction while preserving as much of the class discriminatory information as possible. The input vector representing the ensemble neural activity is projected into a low-dimensional vector (dimension = d) via a linear projection:

$$Y = WX \tag{4.1}$$

where Y is the neuron firing rate vector representing the low-dimensional neural activity, X is the neuron firing rate vector representing the original neural activity, and W is the linear projection. In order to find the optimal low dimensional representations Y, different reductions, namely, d = 3, 9, 15, and 21 were analyzed.

We hypothesized that the LDA method would provide a better neural information extraction than the drop out analysis and lead to a higher decoding accuracy. In order to validate this hypothesis, we also implemented the drop out analysis and randomly selected d neurons whose ensemble activities were used as inputs to the HMM state decoder. All the decoding accuracy results were averaged via a 5-fold cross-validation. We then compared these results with the decoding results from using the LDA method with the same degree of reduction. In order to test whether the decoding accuracies obtained via the LDA method were statistically significantly different from those obtained using the drop out analysis, we carried out student paired-sample t-test at the 5% significant level.

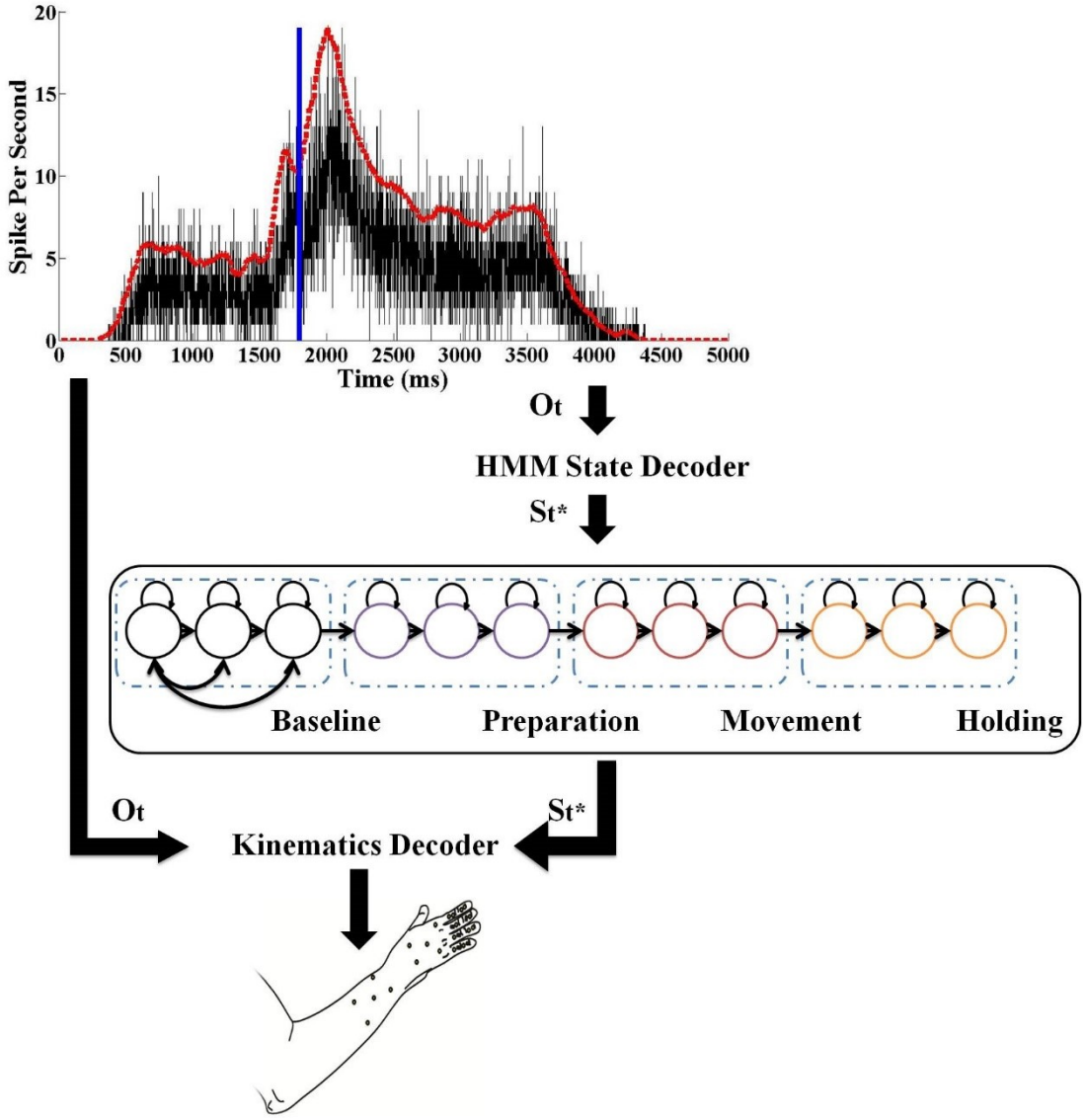
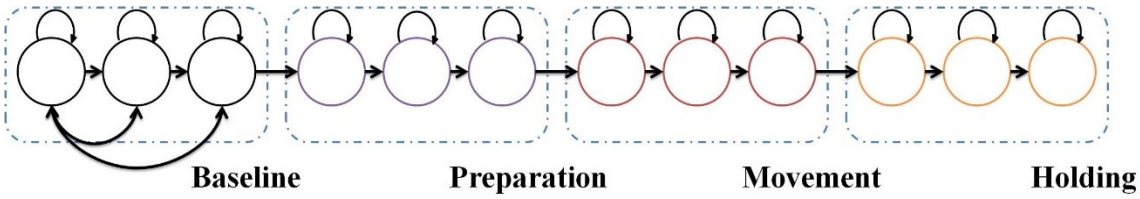
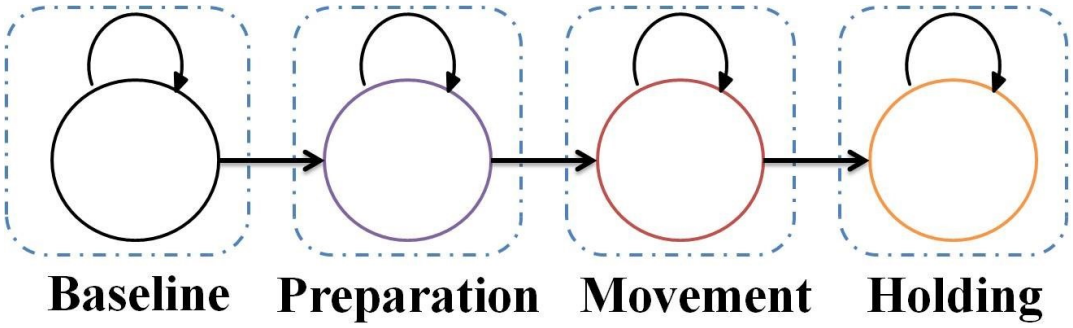


Fig. 4-5 Overview of the decoding framework. After the initial preprocessing, the neural activity enters the HMM state decoder and the corresponding cognitive state sequences and transition times are estimated. The decoded cognitive state information is incorporated in the following kinematics decoder to further estimate the epoch-based kinematics. Here O_t represents the neural activity, and S_t^* the decoded cognitive states. The decoding performance of the proposed framework is judged in the following three main areas: decoding accuracy of cognitive states, estimation latency of cognitive states, and decoding accuracy of kinematics.



(A)



(B)

Fig. 4-6 A Bayesian network representation of the HMM-based state decoder structures. (A) Each epoch was represented by three latent state variables, which formed an individual left-to-right HMM. Each latent state variable generated observations following an emission probabilistic distribution modeled with a mixture of three Gaussians with either a diagonal covariance matrix or a full covariance matrix. The arrows between the latent state variables represented the transitions between different states. The loops around each discrete state represented the states could return to itself. For future reference, we name this model as an extended HMM. (B) A simple HMM model similar to the extended HMM, except each epoch was represented by only one latent state variable.

4.2.4 Cognitive State Transition Detection

We took a supervised approach and used a HMM scheme to decode four distinct epochs of neural activity including baseline, planning, movement and holding. Our HMM decoding framework explicitly modeled each epoch of neural activity with one or several latent state variables. We hypothesized that using multiple latent state variables to model each epoch may improve the cognitive state decoding accuracy. In order to test this idea, we implemented two cognitive state decoding schemes: a complex HMM decoding scheme where each epoch of neural activity was represented by three latent state variables, and a simple HMM decoding scheme where each epoch of neural activity was represented by one latent state variable (Fig. 6). For future reference, we refer to the simple model with one latent state variable per epoch as the simple HMM model, and the model with three latent state variables per epoch as the extended HMM. The latent state variable, S_p, changed with time and was modeled as a first-order Markov process. To simplify the computation, we assumed that the probability of transition from state i to state j at time t, A_{ij}, only depended on the state i and state j and not on the time t

$$P(S_{t+1} = j \mid S_t = i) = A_{ij} \tag{4.2}$$

State changes were considered in a 100-ms moving window with a step size of 1 ms. In other words, if we denote the neuron firing rate vector sampled at time t as $v(t)$, one observation for the HMM decoding framework is

$$V(t) = \left[v(t+1) \quad v(t+2) \quad \ldots \quad v(t+100) \right] \tag{4.3}$$

and the next observation is

$$V(t+1) = \left[v(t+2) \quad v(t+2) \quad \ldots \quad v(t+101) \right] \tag{4.4}$$

We denote the number of spikes of the m-th neuron recorded in a 100-ms moving window starting at time t by $v_t(m)$, which is further divided by the width of the window to make it fit for modeling using a mixture of Gaussians, denoted by $v_t'(m)$. The activity of the ensemble of N neurons is denoted by a vector

$$v'_t = \left[v_t'(1) \quad v_t'(2) \quad \ldots \quad v_t'(N) \right] \tag{4.5}$$

Besides modeling cognitive state transitions, it was also necessary to model the mapping between the latent state variables and the observations, namely, epochs of neural activity in the form of neuron firing rate vectors. We modeled the neural activity conditioned on the latent state variables using a mixture of Gaussians with the means and standard deviations calculated from the neural activity. A mixture of Gaussians enabled us to approximate the unknown probabilistic distribution of neural activity with only a few parameters, which helped to get robust model parameter estimation.

In terms of model structures, we also considered the differences in firing rates of different reach-to-grasp tasks. To address the task-dependence of neural activity, we constructed a model with separate states corresponding to each reach-to-grasp task. Fig. 6 depicts the structure of two models including the extended HMM and the simple HMM, where the latent state variables are represented by circles, the relationship between latent state variables are represented by the directional arrows connecting the circles, and the mapping between the neural activity observations and the corresponding latent state variables are represented by the directional arrows connecting the circles and mixtures of Gaussians. Both the extended HMM and the simple HMM were initiated with separate state transition matrices, where state transitions followed a left-to-right manner, as shown in Fig. 6. Based on the *a posteriori* likelihood (APL) of a state (j) at time t, $\mathrm{P}\left(S_t = j \mid v'_{1:t}\right)$, the model chooses the most probable state based on the observations from time 1 to time t.

In order to avoid the potential underflow problem, we normalized the probability out of recursive multiplication each time. Implementation details are included in the [26].

We took a supervised way for parameter estimation which used the method of expectation maximization (EM), an iterative approach to estimate the parameter values with limited training data [26]. The objective of EM is to maximize $P(S \mid O, \lambda)$, the probability of a latent state sequence $S = S_1 S_2 ... S_N$ (N is the number of observations) conditioned on certain model parameters λ and neural activity observations $O = O_1 O_2 ... O_N$. We pre-defined the model structure and used the training data to estimate the model parameters.

Since we first applied the LDA method to the neural inputs to reduce redundant correlated information among the neural inputs, we assumed the elements in the inputs to the HMM-based state decoder were independent of each other. In order to test whether this assumption was held on our unique datasets, we considered two cases: a diagonal covariance matrix and a full covariance matrix to model the emission probability distribution. The difference lies in that a diagonal covariance matrix assumes that the elements in the inputs of the HMM-based state decoder were independent of each other whereas a full covariance matrix does not assume so.

4.2.5 Cognitive State Transition Detection Accuracy Measurement

For each of the four reach-to-grasp tasks, namely, pushing a button (#trials=186), pulling a cylinder (#trials=173), pulling a mallet (#trials=187), grasping a sphere (#trials=184), mutually exclusive feature sets were used for training (80%) and testing (20%) for each task. All the results here were averaged via a 5-fold cross-validation. For a testing trial with k frames, the decoded state sequence

$$S = S_1 S_2 ... S_k \tag{4.6}$$

was compared to the true state sequence

$$S' = S'_1 S'_2 ... S'_k \tag{4.7}$$

The decoding accuracy of each trial is equal to the number of frames with correctly decoded states divided by the total number of frames in the trial,

$$accuracy = \frac{\sum_{i=1}^{k} L(S_i' \quad S_i)}{k} * 100\%$$
$$L(S_i' \quad S_i) = \begin{cases} 1 & S_i' = S_i \\ 0 & S_i' \neq S_i \end{cases} \tag{4.8}$$

While the decoding accuracy can show us the overall performance of the decoding scheme, real world applications would also have concerns about the latencies between the real epoch transition times and the estimated epoch transition times. In this work, we implemented an offline Viterbi decoding algorithm [27], [28]. There are online Viterbi decoding algorithms which can be used for real-time applications [29], and we leave it for future work. However, the latency performance of the proposed decoding scheme may indicate how well the corresponding online decoding system would perform on our datasets. Here we defined the decoding latency (in milliseconds) for each epoch transition of interest as:

Latency = the true state transition time - the estimated state transition time.

Latencies were measured for three transitions between four epochs including baseline, planning, movement and holding. All the results were averaged via a 5-fold cross-validation.

4.2.6 Conversion to Joint Angles from Positional Kinematics

The Vicon motion tracking system recorded the 3D positions of motion tracking markers in the Cartesian coordinate space. We converted the kinematics data into 18 joint angles including the three angles of wrist (the wrist flexion-extension angle, the wrist abduction-adduction (ab-ad) angle, the wrist rotation angle), as well as the flexion-extension angle, the ab-ad angle, and the opposition angle for each of five fingers. The detailed steps of conversion is included in [18].

4.2.7 State-Dependent Decoding of Kinematics

Since there are four epochs, a linear dynamical system may not be able to generalize the sequence whose dynamics changes over time. Examples of such data include human motion such as walking, running, and dancing each of which exhibits complex constantly evolving dynamics. Switching linear dynamical systems (S-LDSs) are powerful models capable of describing a physical process governed by state equations that switch from time to time [30].

S-LDSs are able to model both discrete S and continuous latent states X. The discrete state transitions are modeled with a Markov process. The probability distribution of continuous latent variables X depends on the discrete latent state variables S. Similarly, the probability distribution of the observations O are conditioned on both continuous and discrete latent state variables.

Different from most problems using S-LDS models, however, the transitions between the discrete states $S_{1:N}$ were first decoded from the HMM state decoder and known during in both the training and testing stages of the S-LDS kinematics decoder. The continuous states $X_{1:N}$ were also available during the training stage in order to learn the corresponding mapping between neural activity and kinematics. Therefore, there was no need of employing commonly used probabilistic inference methods of S-LDSs. Instead, we trained a linear time invariant (LTI) Kalman filter [31] whose parameters were independent of time but epoch-dependent for each epoch. The state space equations are as follows:

$$\begin{aligned} x_{t+1} &= F^i x_t + u_t^i \\ O_t &= H^i x_t + z_t^i \end{aligned} \tag{4.9}$$

where each epoch is represented as $S_t = i \ \ (i = 1,2,3,4)$, and each 18-dimension joint angle vector per 5 ms was modeled a continuous latent state variable x_t. The firing rate vectors during the previous 5 ms were modeled as the observations O_t. The transition matrices (F^i, H^i) were both conditioned on the discrete latent state variables. The state noise and the measurement noise follow epoch-dependent independent and identically distributed (i.i.d.) Gaussian distribution $u_t^i \sim N(0, \ \Sigma^i{}_u)$ and $z_t^i \sim N(0, \ \Sigma^i{}_z)$. The state noise and the measurement noise are also independent from each other. The decoding algorithm follows the Kalman filter algorithm, which includes two steps, namely, prediction and update. The details about the algorithm can be found in [31].

4.2.8 Virtual World in MSMS

We simulated the real-world experimental setup using a virtual environment created by Musculoskeletal Modeling Software (MSMS) [18] which was developed by the University of Southern California. MSMS includes detailed models for the monkey's upper extremity and objects used in experiments, which can be controlled via specifying the individual positions or joint angles in the upper extremity. The virtual

environment in MSMS is programmable and can be modified to different experimental settings easily. In order to visualize the differences between the estimated and true joint angle kinematics, reconstructed upper extremity was overlaid with the actual upper extremity as the monkey performed different reach-to-grasp tasks.

4.3 Results

4.3.1 Dimension Reduction vs. Drop-Out Analysis

The decoding accuracy of the extended HMM state decoder was compared using both the LDA method and drop out analysis as the preprocessing step over a wide range of dimensions in the low-dimensional neuron firing rate vector space. Here we only presented the results with $d = 9$ (Fig. 4-7) since very similar results were observed for other dimensions. The decoding accuracies using the LDA method as the preprocessing step were consistently better than that of the drop out analysis. We also carried out the paired sample student t-test using the 5% significant level, and the p-value was 0.0242, indicating that the differences in the decoding accuracies were statistically significant.

4.3.2 Cognitive State Transition Detection Accuracy

Both the extended HMM state decoder and the simplified HMM were trained offline using ensemble neural activity from premotor neural activity and corresponding recordings of the epoch transition time points.

As shown in the first two rows in Table I, the HMM state scheme achieved high decoding accuracies across the four different reach-to-grasp tasks using the extended HMM state decoder. This shows that the extended HMM state decoder can capture the statistical differences of firing rates from premotor cortical neurons during different epochs. Comparing the decoding accuracies between the extended HMM state decoder using a diagonal covariance matrix and one with a full covariance matrix, the former model had better decoding accuracies. The average decoding accuracy difference was 3.8% ± 0.74% across four behavioral tasks.

The decoding results in the first row and the third row in Table I show that the extended HMM state decoder had better decoding performance than the simple HMM model across four different reach-to-grasp tasks. The average decoding accuracy difference was 9.47% ± 4.67%. This result indicates that accounting for the variability within each epoch may improve the overall descriptive capacity of a decoding model. Even though the simple HMM state decoder has far fewer parameters than the extended HMM state decoder, the descriptive power of the simple HMM state decoder is more limited the extended HMM state decoder, and thus cannot fully capture the changing dynamics in the firing rate within each epoch.

The performance of the decoding schemes was found to increase slightly with the decrease in the degrees of dimension reduction. As shown in the Fig. 8, over a wide range of dimensions d of the low-dimensional space, the performance of the decoding system only increased slightly, which held true for four different reach-to-grasp tasks. On the other hand, the training time and decoding time using a large degree of dimension reduction was much smaller than that without dimension reduction. These results suggest that using dimension reduction as the preprocessing unit may improve the computational efficiency of the overall decoding system.

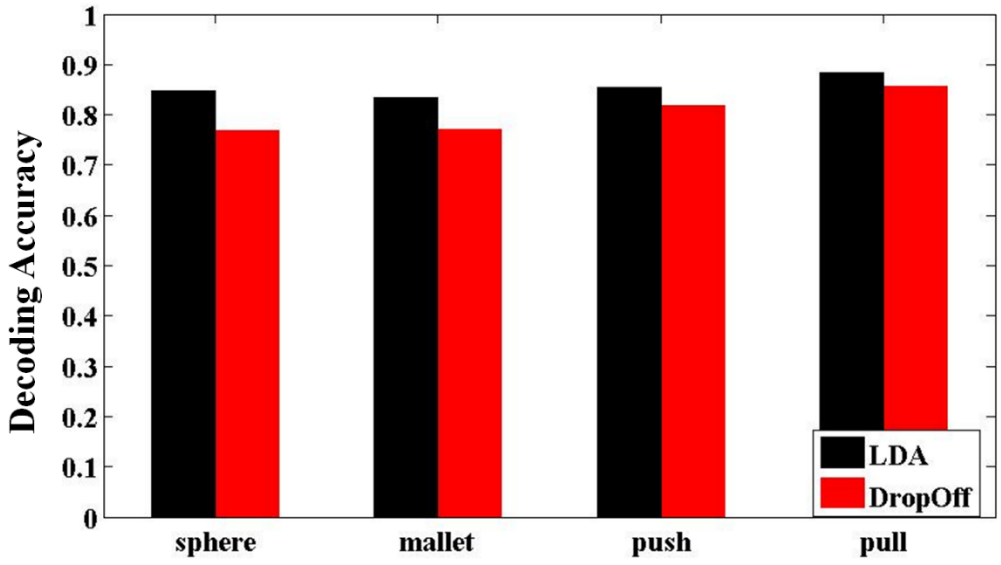


Fig. 4-7 The comparison of decoding accuracies between the LDA method (left, dark color) and simple drop out analysis (right, light color) in four experimental tasks. The decoding system using the LDA method has consistently better performance than the corresponding decoding system using the drop out analysis. Here

the results of four different behavioral tasks using the dimension $d = 9$ of the low-dimensional space are presented. The *paired*-sample student *t*-test using these two groups of results led to a *p* value of 0.0242, indicating that the difference is statistically significant.

Table 4-1 The decoding accuracy results for each of four reach-to-grasp tasks (the dimension of the low dimensional space is 21) using four different experiment settings in the Gaussian mixture modeling: (1) LDA with 3 states per epoch using a diagonal covariance; (2) LDA with 3 states per epoch and using a full covariance; (3) LDA with 1 state per epoch and using a diagonal covariance.

Settings\Task	Mallet	Pull	Push	Sphere
LDA (3S, diagonal)	0.862 (±0.016)	0.887 (±0.020)	0.883 (±0.010)	0.852 (±0.039)
LDA (3S, full)	0.836 (±0.010)	0.850 (±0.035)	0.839 (±0.030)	0.815 (±0.036)
LDA (1S, diagonal)	0.794 (±0.060)	0.844 (±0.066)	0.755 (±0.048)	0.712 (±0.051)

Table 4-2 The absolute decoding latency results (in milliseconds) of the start of the second, third, and fourth epoch for behavior tasks (the dimension of the low dimensional space is 21) using a full covariance matrix in the mixture of Gaussians modeling and LDA with 3 states per epoch.

Transition\Task	Mallet	Pull	Push	Sphere
1st--2nd epoch	33 (±46)	114 (±61)	106 (±90)	165 (±98)
2nd--3rd epoch	53 (±99)	11 (±32)	44 (±75)	60 (±49)
3rd--4th epoch	59 (±66)	32 (±70)	60 (±84)	24 (±79)

Note: all the results above are the average from five-fold cross validation.

Table 4-3 The average epoch durations (in milliseconds) for four tasks.

Settings\Task	Mallet	Pull	Push	Sphere
1st epoch	715.97	725.45	717.39	745.93
2nd epoch	246.88	257.15	290.66	255.82
3rd epoch	1285.68	1253.95	1386.34	1237.52
4th epoch	848.00	1055.49	718.05	488.55

Note: all the results above are the average from all trials in each task.

4.3.3 Cognitive State Transition Detection Latency

Since real-life application performance of the decoding system critically depends on decoding latencies, we also measured the latencies for detecting epoch transitions (from baseline to planning, from planning to movement, and from movement to holding) using the extended HMM state decoder. Table II shows the decoding latencies for the epoch transitions. All of the average latencies across four different tasks were much smaller than 215 ms, which was the average primate response time. For the transition from the baseline to planning epoch, the average decoding latency in the task of pulling a mallet was the smallest (33 ms ± 46 ms), and that of grasping a sphere was the largest (165 ms ± 98 ms). The transition from the planning epoch to the movement epoch had the smallest latency in the task of pulling a cylinder (11 ms ± 32 ms), and the largest latency in the task of grasping a sphere (60 ms ± 49 ms). In contrast, the transition from the movement epoch to the holding epoch had the smallest latency in the task of grasping a sphere (24 ms ± 79 ms), and the largest latency in the task of pulling a mallet (60 ms ± 84 ms).

Comparing the average epoch durations of four different behavioral tasks, the decoding latency, on average, accounts for 5.12% of the total duration of two neighboring epochs. More specifically, the average decoding latency for the transition between the baseline to planning epochs accounts for 10.56% of the total duration of the baseline and planning epochs, 2.7% for the transition between the planning and movement epochs, and 2.11% for the transition between the movement and the holding epochs.

We find that the decoding latencies not only vary with the reach-to-grasp tasks, but also change with the degrees of dimension reductions. As shown in Fig. 9, the decoding latencies for three epoch transitions decrease slowly with the decrease in the degrees of dimension reduction in the task of grasping a mallet. Similar trends were also observed other three reach-to-grasp tasks.

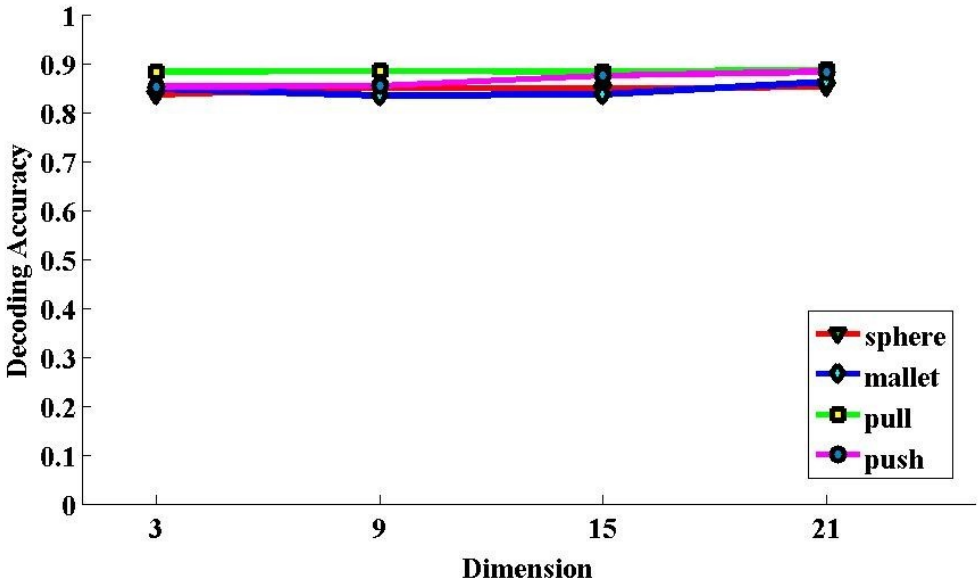


Fig. 4-8 Decoding accuracy vs. the degrees of dimension reduction. The performance of the HMM decoder remains stable over a wide range of degrees of dimension reduction.

4.3.4 Kinematics Decoding Accuracy

The S-LDS decoder was trained offline using recorded neural signals from premotor cortex and corresponding joint angle kinematics. Table IV shows the mean correlation coefficients for continuous prediction of finger, hand, and wrist joint angle kinematics during the reach-to-grasp tasks. As is seen from Table IV, the S-LDS kinematics decoder estimated the kinematics of all joint angles with high accuracy (mean = 88.0%, standard derivation = 2.05%).

Both the actual joint angles recorded from real-world experiments and the decoded joint angles from the S-LDS kinematics decoder were then transmitted into the virtual environment in the MSMS, and the experimental trials were replicated with animation. Fig. 4-10 shows the reconstructed upper extremity (blue shadow arm) overlaid with the actual upper extremity (gray solid arm) as the monkey performed different reach-to-grasp tasks. As shown in our simulation results, the decoded output closely resembles the original

movements of the upper extremity, especially the subtle movements of the hand and fingers during different behavioral tasks.

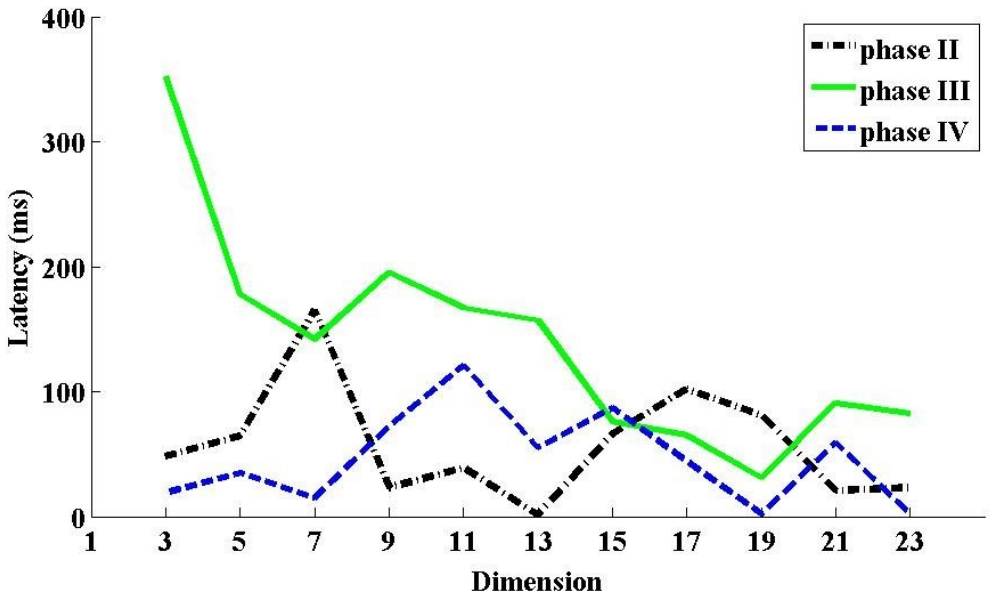
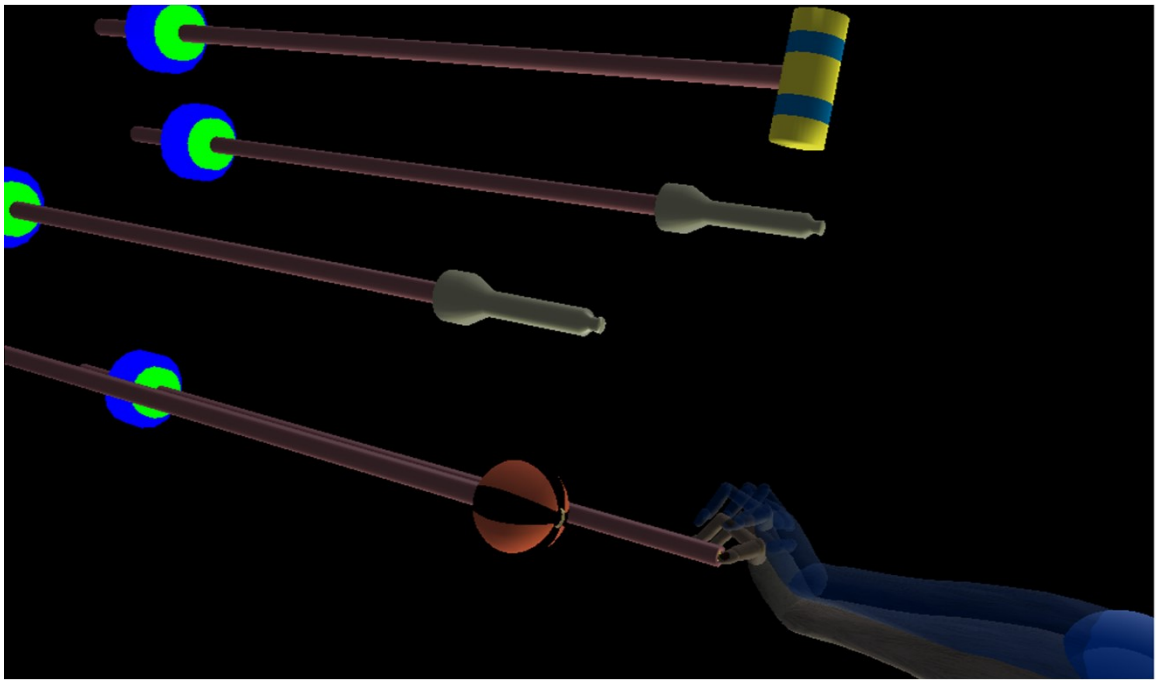


Fig. 4-9 State decoding latencies for three epoch transitions vs. the dimension of the lower dimensional representation in the task of grasping a mallet. The latencies of the HMM decoder rapidly decrease over a wide range of dimensions of the low dimensional space using the LDA with full covariance matrices.



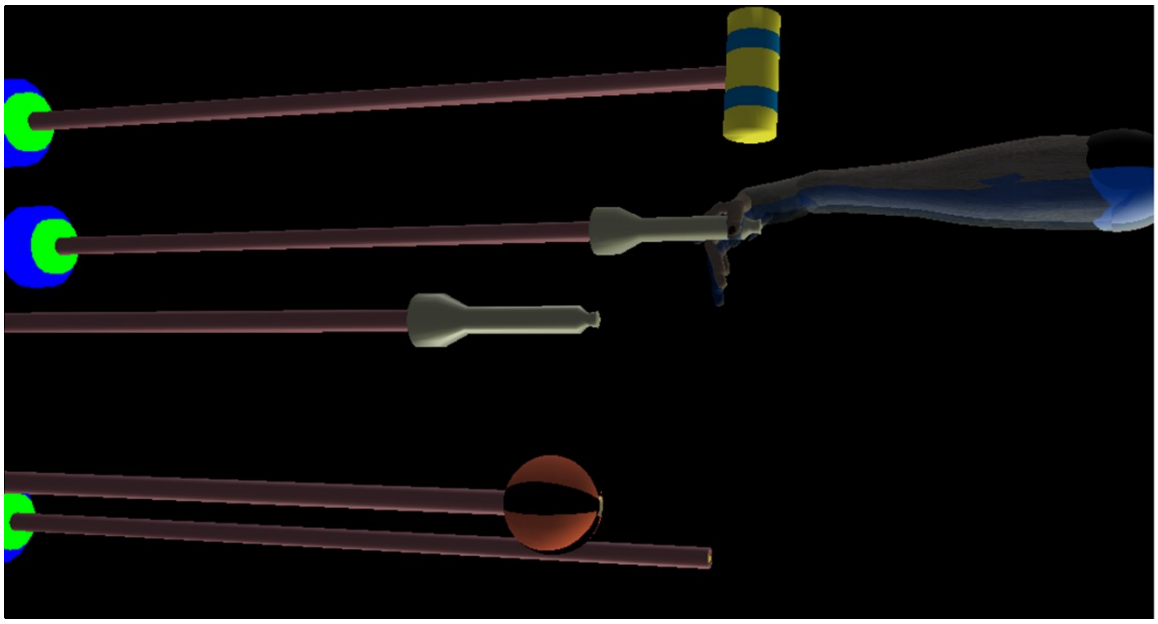


Fig. 4-10 Both the actual kinematics recorded during real-world experiments and the decoded joint angles were transmitted into the virtual environment in MSMS to simulate the real experiments. The example movements are pushing a button and grasping a cylinder.

Table 4-4 DECODING ACCURACY (CORRELATION COEFFICIENT) FOR 18-JOINT ANGLES IN FOUR DIFFERENT REACH-TO-GRASP TASKS

	PUSH	MALLET	SPHERE	PULL
Wrist	0.897	0.878	0.898	0.893
Thumb	0.899	0.884	0.898	0.896
Index	0.890	0.875	0.827	0.822
Middle	0.893	0.868	0.870	0.857
Ring	0.893	0.886	0.891	0.885
Pinky	0.891	0.877	0.881	0.894
Total	0.890	0.868	0.885	0.877

Note that the correlation values presented in the table above were averaged across all trials of each task.

4.4 Discussion and Conclusions

4.4.1 Cognitive State Transition and Kinematics Estimation

Studies have shown that subjects may employ distinct control policies during different epochs of the experiments. Neural activity in the planning epoch has been shown to tune based on the next movement such

as the target position [32] and the movement speed [33]. Common epochs of neural activity may include baseline, planning, movement and holding. Since the dynamics and encoded information in distinct epochs of neural activity may be different, neural prosthetics with only a single decoder throughout movements may have sub-optimal performance or even be impractical for practical applications. On the other hand, autonomous detection of cognitive state transitions and corresponding kinematics estimation during movements may lead to significant performance improvement of neural prosthetics by enabling the use of different controllers/decoders which are optimally designed for each unique cognitive state. Furthermore, autonomous detection of cognitive state transitions may have additional benefits including faster response and better computational efficiency towards a clinically practical neural prosthetic design.

Previous research has primarily focused on movement target decoding assuming known cognitive state transitions [14–16], [20], [21], [23], [24] or detecting transitions of cognitive states including baseline, planning, and movement from neural activity [22]. In contrast to previous works, this work has several key differences. First, this work uses a supervised approach instead of a unsupervised way employed by most previous work [14–16], [20], [21], [23], [24]. The supervised approach enables us to model the transitions between cognitive states and the correspondence between cognitive states and kinematics in a structurally meaningful way and get robust parameter estimation.

Furthermore, compared to recent work which also took a supervised approach in cognitive state detection, the proposed decoding framework mainly includes a HMM state decoder and a S-LDS kinematics decoder, which enables us to decode both the cognitive state transitions and epoch-dependent kinematics. In addition, the proposed HMM state decoder is able to decode novel cognitive state, the holding epoch, besides other cognitive states mentioned in previous works at high accuracy. Detecting the transition from the movement to holding state may potentially be very useful in practical neural prosthetics implementations for two reasons. The holding state in the reach-to-grasp tasks corresponds to static wrist, hand, and finger positions in the grasping experimental phase. Traditional neural prosthetics designed with a single decoder throughout movements tend to produce movements with tremor in the holding state [34]. Autonomous detection of the holding state means a proper decoder may be used the holding state to remove the tremor. In addition, advanced classifiers can be further incorporated for target object prediction and recognition, which

may also significantly improve the practical performance of neural prosthetics. Moreover, the HMM state decoder in this work is able to detect the transitions of cognitive states with short latency.

4.4.2 Main Contributions

Our work has resulted in several contributions. We first demonstrate that the LDA method provides better neuronal information extraction than the drop out analysis. The LDA method projects the high-dimensional firing rate vector to a low-dimensional space, removing redundant information such as correlation information in the neuronal inputs and preserving the core neuronal information. The drop out analysis, on the other hand, is based on the assumption that there is high correlation in all of the neuronal inputs. Without detailed studies of the underlying neuronal network and weighting the importance of different neurons, the drop out analysis may lead to a loss in the neuronal information and subsequently the decrease in the overall performance of the decoding system.

Another contribution is that we demonstrate that it is feasible to automatically decode multiple distinct cognitive states and corresponding high-dimensional kinematics using a generative decoding framework including a HMM state decoder and a S-LDS kinematics decoder. Our decoding framework uses a supervised approach to learn the correspondence between the latent state variables and epochs of neural activities as well as epochs of kinematic data. We design the HMM state decoder's structure to best fit our experimental design which includes four distinct epochs. We use a mixture of three Gaussians with either a diagonal covariance matrix or a full covariance matrix to model the emission probabilistic distributions of each latent state variable. Two HMM state decoders are presented including the extended HMM state decoder and the simple one. We find that it significantly improves the decoding accuracy by considering the variability of neural activity within each epoch and incorporating multiple latent state variables within each epoch. The extended HMM can capture the changing dynamics within each epoch and at the same time not require many parameters to estimate, which further leads to a more robust parameter estimation in the extended HMM and more descriptive power in capturing the dynamics within each epoch. The S-LDS kinematics decoder is trained with epoch-dependent parameter estimation.

We also study in detail the structure of the decoding schemes by varying the degrees of dimension reduction, the structure of covariance matrices in the mixture of Gaussians, and the number of latent state

variables in each epoch. We find that in general, a decoding system with few parameters tends to have a better decoding performance because of more robust parameter estimation. This result is confirmed in the decoding performance comparison between the full covariance matrix and the diagonal covariance matrix. However, the decrease in the number of parameters in a decoding system may also compromise the descriptive capacity of the model, and thus lead to a worse decoding performance. This can be seen in the comparison between the decoding accuracies of decoding systems using the three latent state variables per epoch and one latent state variable per epoch.

Practical clinical implementations of neural prosthetics may prefer a decoding scheme with as small latency as possible. Thus, even though the decoding accuracy of an HMM with the low dimension equal to 3 is comparable to one with the low dimension equal to 21, the latter model produces much smaller latencies, and therefore may be more suitable for real life applications. On the other hand, the low dimension equal to 3 uses shorter training and decoding time. This result indicates that when using the proposed decoding framework, there is a trade-off between the decoding performance measured by decoding latencies and the computational efficiency brought by the initial dimension reduction approach.

Furthermore, this work demonstrates that it is possible to decode state-dependent kinematics of the fingers, hand and wrist with high accuracy using the premotor cortical neural ensembles. Related Studies showed that when monkeys performed highly similar reaches under tight behavioral control, which is true in our experimental design. The neural activity in PMd was predominantly stable over time in all measured properties such as firing rate, directional tuning, and contribution to the S-LDS kinematics decoder [35]. The changes in PMd neural activity were mainly correlated with the behavioral changes. There is, thus, a stable relationship between the premotor neural activity and observed behavior kinematics [35]. Our findings about the high decoding capability of PM ensembles has significant implications for the design of neural prosthetic systems because it suggests that a general purpose brain-machine interface can employ multiple cortical areas for motor control.

4.4.3 Limitations and Future Directions

The decoding schemes can be further extended in various ways. First, the decoding schemes may use other emission probability distributions such as the Poisson distribution. In addition, the proposed framework

can be used to decode other cognitive states such as delay periods commonly seen in animal behavioral experiments.

One limitation of this work is that the decoding schemes are currently implemented offline instead of online. It will be part of our future work to create an equivalent online decoding system. Another limitation lies in the limitation of the HMM method. Even though an HMM indicates the interval when a cognitive state may change, it lacks sufficient details for a robust study of how the changes are occurring. The dynamic information embedded in the neural activity is considered critical in the field of neurophysiology, as it is rich with coding details of how cognition is translated to motor activity and control, which is expected to be imperative to actual prosthetic construction. In order to understand the latent neural network driving the change of cognitive states, connectivity models may be incorporated into the current HMM decoding frameworks in future.

The decoding schemes present in this work not only can be applied in the neural prosthesis field, but also can be used in the a wide range of operations such as computer assisted robotic surgeries and tele-operations of robotics, where detecting the cognitive states and estimating the kinematics of operators may assist the control of surgical robots.

4.5 Bibliography

[1] E. E. Fetz, “Real-time control of a robotic arm by neuronal ensembles.,” Nat. Neurosci., vol. 2, no. 7, pp. 583–4, Jul. 1999.

[2] 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 brain-machine interface for reaching and grasping by primates.,” PLoS Biol., vol. 1, no. 2, p. E42, Nov. 2003.

[3] J. P. Donoghue, “Connecting cortex to machines: recent advances in brain interfaces.,” Nat. Neurosci., vol. 5 Suppl, pp. 1085–8, Nov. 2002.

[4] S. Musallam, B. D. Corneil, B. Greger, H. Scherberger, and R. A. Andersen, “Cognitive control signals for neural prosthetics.,” Science, vol. 305, no. 5681, pp. 258–62, Jul. 2004.

[5] M. A. Nicolelis, “Actions from thoughts.,” Nature, vol. 409, no. 6818, pp. 403–7, Jan. 2001.

[6] G. Santhanam, S. I. Ryu, B. M. Yu, A. Afshar, and K. V. Shenoy, “A high-performance brain-computer interface.,” Nature, vol. 442, no. 7099, pp. 195–8, Jul. 2006.

[7] A. B. Schwartz, “Cortical neural prosthetics.,” Annu. Rev. Neurosci., vol. 27, pp. 487–507, Jan. 2004.

[8] S. H. Scott, “Converting thoughts into action,” Nature, vol. 442, no. July, pp. 9–10, 2006.

[9] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows, and J. Donoghue, “Instant neural control of a movement signal,” Nature, vol. 416, no. March, pp. 141–142, 2002.

[10] D. M. Taylor, S. I. H. Tillery, and A. B. Schwartz, “Direct cortical control of 3D neuroprosthetic devices.,” Science, vol. 296, no. 5574, pp. 1829–32, Jun. 2002.

[11] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. a Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn, and J. P. Donoghue, “Neuronal ensemble control of prosthetic devices by a human with tetraplegia.,” Nature, vol. 442, no. 7099, pp. 164–71, Jul. 2006.

[12] P. R. Kennedy and R. A. Bakay, “Restoration of neural output from a paralyzed patient by a direct brain connection.,” Neuroreport, vol. 9, no. 8, pp. 1707–11, Jun. 1998.

[13] P. R. Kennedy, R. A. Bakay, M. M. Moore, K. Adams, and J. Goldwaithe, “Direct control of a computer from the human central nervous system.,” IEEE Trans Rehabil Eng, vol. 8, no. 2, pp. 198–202, Jun. 2000.

[14] K. V. Shenoy, D. Meeker, S. Cao, S. A. Kureshi, B. Pesaran, C. A. Buneo, A. P. Batista, P. P. Mitra, J. W. Burdick, and R. A. Andersen, “Neural prosthetic control signals from plan activity.,” Neuroreport, vol. 14, no. 4, pp. 591–6, Mar. 2003.

[15] N. Achtman, A. Afshar, G. Santhanam, B. M. Yu, S. I. Ryu, and K. V. Shenoy, “Free-paced high-performance brain-computer interfaces.,” J. Neural. Eng, vol. 4, no. 3, pp. 336–47, Sep. 2007.

[16] N. Hudson and J. W. Burdick, “Learning Hybrid System Models for Supervisory Decoding of Discrete State, with applications to the Parietal Reach Region,” Int IEEE/EMBS Conf Neural Eng, pp. 587–592, May 2007.

[17] S. Acharya, F. Tenore, V. Aggarwal, R. Etienne-Cummings, M. H. Schieber, and N. V. Thakor, “Decoding individuated finger movements using volume-constrained neuronal ensembles in the M1 hand area.,” IEEE Trans Neural Syst Rehabil Eng, vol. 16, no. 1, pp. 15–23, Feb. 2008.

[18] V. Aggarwal, M. Kerr, A. G. Davidson, R. Davoodi, G. E. Loeb, M. H. Schieber, and N. V. Thakor, “Cortical control of reach and grasp kinematics in a virtual environment using musculoskeletal modeling software,” Int IEEE/EMBS Conf Neural Eng, pp. 388–391, Apr. 2011.

[19] V. Aggarwal, S. Acharya, F. Tenore, H.-C. Shin, R. Etienne-Cummings, M. H. Schieber, and N. V. Thakor, “Asynchronous decoding of dexterous finger movements using M1 neurons.,” IEEE Trans Neural Syst Rehabil Eng, vol. 16, no. 1, pp. 3–14, Feb. 2008.

[20] L. Srinivasan, U. T. Eden, S. K. Mitter, and E. N. Brown, “General-purpose filter design for neural prosthetic devices.,” J. Neurophysiol., vol. 98, no. 4, pp. 2456–75, Oct. 2007.

[21] W. Wu, M. J. Black, D. Mumford, Y. Gao, E. Bienenstock, and J. P. Donoghue, “Modeling and decoding motor cortical activity using a switching Kalman filter.,” IEEE Trans. Biomed. Eng., vol. 51, no. 6, pp. 933–42, Jun. 2004.

[22] C. Kemere, G. Santhanam, B. M. Yu, A. Afshar, S. I. Ryu, T. H. Meng, and K. V. Shenoy, “Detecting neural-state transitions using hidden Markov models for motor cortical prostheses.,” J. Neurophysiol., vol. 100, no. 4, pp. 2441–52, Oct. 2008.

[23] M. Abeles, H. Bergman, I. Gat, I. Meilijson, E. Seidemann, N. Tishby, and E. Vaadia, “Cortical activity flips among quasi-stationary states.,” Proc Natl Acad Sci U S A, vol. 92, no. 19, pp. 8616–20, Sep. 1995.

[24] E. Seidemann, I. Meilijson, M. Abeles, H. Bergman, and E. Vaadia, “Simultaneously recorded single units in the frontal cortex go through sequences of discrete and stable states in monkeys performing a delayed localization task.,” J. Neurosci., vol. 16, no. 2, pp. 752–68, Jan. 1996.

[25] M. Mollazadeh, V. Aggarwal, A. G. Davidson, A. J. Law, N. V. Thakor, and M. H. Schieber, “Spatiotemporal variation of multiple neurophysiological signals in the primary motor cortex during dexterous reach-to-grasp movements.,” J. Neurosci., vol. 31, no. 43, pp. 15531–43, Oct. 2011.

[26] L. R. Rabiner, “A tutorial on HMM and selected applications in speech recognition.pdf,” Proc. IEEE, vol. 77, no. 2, pp. 257–286, 1989.

[27] A. J. Viterbi, “Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm,” IEEE Trans Inf Theory, vol. IT-13, pp. 260–269, 1967.

[28] D. J. Forney, “The Viterbi algorithm,” Proc. IEEE, vol. 61, pp. 268–278, 1973.

[29] S. Rastislav, “The on-line Viterbi algorithm,” Thesis, Dept. Compu. Sci., Comenius Univ., p. 51, 2007.

[30] Z. Ghahramani and G. E. Hinton, “Parameter Estimation for Linear Dynamical Systems,” Technical Report , Dept. Comp. Sci., Univ. Toronto, vol. CRG-TR-96, pp. 1–6, 1996.

[31] G. Bishop and N. Carolina, “An Introduction to the Kalman Filter,” In Practice, vol. 7, no. 1, pp. 1–16, 2006.

[32] D. J. Crammond and J. F. Kalaska, “Prior Information in Motor and Premotor Cortex : Activity During the Delay Period and Effect on Pre-Movement Activity,” J. Neurophysiol., vol. 84, pp. 986–1005, 2000.

[33] M. M. Churchland, G. Santhanam, and K. V. Shenoy, “Preparatory activity in premotor and motor cortex reflects the speed of the upcoming reach.,” J. Neurophysiol., vol. 96, no. 6, pp. 3130–46, Dec. 2006.

[34] V. Aggarwal, F. Tenore, A. Soumyadipta, M. H. Schieber, and N. V. Thakor, “Cortical decoding of individual finger and wrist kinematics for an upper-limb neuroprosthetics,” Conf Proc IEEE Eng Med Biol Soc., pp. 4535–4538, 2009.

[35] C. A. Chestek, A. P. Batista, G. Santhanam, B. M. Yu, A. Afshar, J. P. Cunningham, V. Gilja, S. I. Ryu, M. M. Churchland, and K. V. Shenoy, “Single-neuron stability during repeated reaching in macaque premotor cortex.,” J. Neurosci., vol. 27, no. 40, pp. 10742–50, Oct. 2007.

Chapter 5 Online Cognitive State Transition Detection

5.1 Introduction

5.2 Methods

5.2.1 Experimental Set-up

5.2.2 Hidden Markov Model

5.2.3 Point Process Observation Models

5.2.4 Model Goodness-of-fit and Model Selection

5.2.5 Neural State Transition Detection

5.2.6 Detection Latency Measurements

5.2.7 Detection Accuracy Measurements

5.3 Results

5.3.1 TD vs. TI models

5.3.2 Online vs. Offline

5.3.3 Models with Kinematics vs. Without Kinematics

5.4 Discussion and Conclusions

5.5 Bibliography

5.1 Introduction

Prior work has focused on detecting cognitive state transitions using task-dependent (TD) models [5, 13-15]. That is, in order to detect a cognitive state, the task must be either known *a priori* or determined in real-time. If the neural prosthetics needs to work for subjects performing a large variety of tasks, a new TD model would be needed for each novel movement task, thus making the TD detection paradigm rapidly unmanageable. Furthermore, even if the computation would not pose a hurdle, one needs data from each movement task to estimate the TD models, despite the fact that cognitive states (e.g., idle, holding, etc.) are invariant to the specific movement task being performed [11].

Therefore, we propose a real-time TI framework to detect cognitive state transitions from spike trains and kinematic measurements, where (1) a TI model is used with neuronal data alone to detect cognitive state transitions when the subject does not move; and, (2) when movements are detected, a TI model is used with both neuronal and kinematic data to detect cognitive state transitions [14-16]. We hypothesize that TI models will allow for a more accurate detection of cognitive state transitions than TD models.

We assume that cognitive states transition according to a hidden Markov model (HMM) and constructed our TI cognitive state transition detection framework using a HMM, The HMM's outputs are cortical spike trains and, only when the subject is moving – kinematic measurements. The HMM's outputs are modeled with generalized linear point process models [17] with history dependencies. Similar state-space techniques have also been used [13, 17-21] in neural signal modeling. Related models have also been developed and applied in the fields of neural information processing and neural data analysis [22-28].

We constructed the real-time TI cognitive state transition detection framework using 226 single-unit recordings collected via multi-electrode arrays in premotor dorsal and ventral (PMd and PMv, respectively) cortical areas of two non-human primates performing 3D multi-object reach-to-grasp tasks [12]. We compared the latency-accuracy performance [29] of the TI vs. TD models both in offline and online detection modes.

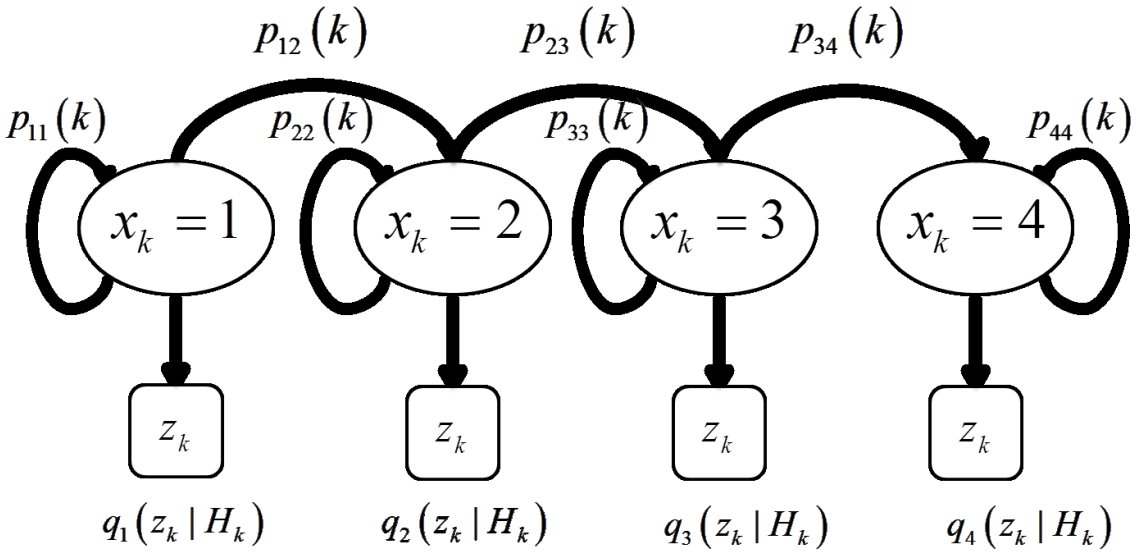
5.2 Methods

5.2.1 Experimental Set-up

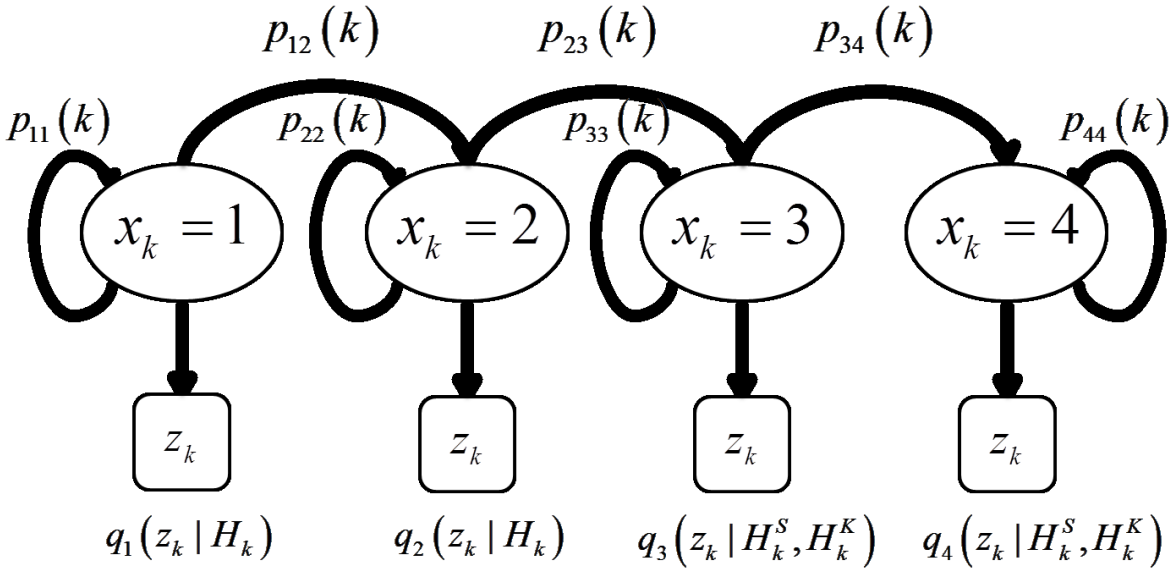
All studies were carried out by Dr. Marc Schieber's group under a protocol approved by the University of Rochester Institutional Animal Care and Use Committee. Two male rhesus monkeys (Macaca mulatta) were trained to sit in a polycarbonate primate chair that restrained the neck, torso and legs, and to follow visual cues to manipulate, grasp, and reach towards one of four target objects. The four objects include sphere, perpendicular mounted cylinder (mallet), pushbutton, or peripheral coaxial cylinder (pull handle), arranged in a planar circle at 60º intervals around the center home cylindrical object (home object) Each object was mounted on the end of a 2-3', 0.5" diameter aluminum rod, and manipulation of the object operated a micro-switch mounted at the opposite end of the rod. Each monkey used its right hand to perform the movements and the left upper extremity was restrained within the primate chair.

Each trial included four cognitive states marked by three visual cues during the experiments (Fig. 2). The baseline began with each monkey holding the home object, which was aligned with the monkey's right shoulder, for a random period between 230 and 1130 milliseconds. At the beginning of planning was a visual cue indicated by a blue LED next to one of the four peripheral objects, which instructed the monkey to release the home object and reach towards the instructed peripheral object. The onset of movement (OM) was marked as the moment the monkey released the home object. During the movement state, as soon as a monkey touched the instructed peripheral object, the monkey was required to grasp and manipulate the instructed object: either rotate the sphere object 45º, pull the mallet object, depress the pushbutton object, or to pull a peripheral mounted cylinder object against its own small spring load [12]. The monkey was required to hold the instructed target object for 1000 milliseconds during the holding state.

Single-unit neuronal activity was recorded using a Plexon (Dallas, TX) data acquisition system from 2 floating microelectrode arrays (FMAs) in Monkey #1: one in PMd and one in PMv, respectively, as well as 4 FMAs in Monkey #2: two in PMd, and two in PMv, respectively. There were two days of neuronal recordings from each monkey. There were totally 148 PMv neurons, and 304 PMd respectively. Each FMA included 16 electrodes and up to four single-units could be discriminated per electrode.



(A)



(B)

Fig. 5-1 Model schematics for the TI models with neuronal data alone (A) and with the combination of neuronal and kinematics data (B). HMM schematic with four hidden states ($x_k = 1$, $x_k = 2$, $x_k = 3$, or $x_k = 4$) to represent four cognitive states and observable outputs z_k, $k = 0,1,2,\ldots$. $p_{ij}(k)$ and $q_x(z_k \mid H_k)$ are the inhomogeneous probabilities of state transition and emission probability functions of neuronal data z_k in state $x \in \{1,2,3,4\}$ conditioned on the neuronal spiking H_k, respectively. $q_x(z_k \mid H_k^S, H_k^K)$ is the emission probability of neuronal data z_k in state $x \in \{2,3,4\}$, respectively, using both neuronal spiking history H_k and kinematics history measurements H_k^K.

5.2.2 Hidden Markov Model

For each of the four movement tasks, namely, pushing a button (#trials=62137, considering each neuron's trial), pulling a cylinder (#trials=58658), pulling a mallet (#trials=60423), grasping a sphere (#trials=60538), mutually exclusive feature sets were used for training (80%) and testing (20%) for each model. All the results here were averaged via 5-fold cross-validation.

We first constructed a TI HMM using neuronal data alone, as shown in Fig. 3A. There are 4 hidden states in the HMM, representing the baseline ($x_k = 1$), planning ($x_k = 2$), movement ($x_k = 3$), and holding ($x_k = 4$) cognitive states in the experiments (Fig. 2). Since all experimental trials start with the baseline, the initial probability of being in the baseline $\pi_0 = P(x_0 = 1) = 1$, and 0 for all the other three cognitive states.

We estimated the transition probabilities $p_{ij}(k) = P(x_k = j \mid x_{k-1} = i)$ for all time steps k, cognitive states $i, j \in \{1,2,3,4\}$, and emission probability density functions $q_i(z_k \mid H_k)$ separately. Here, a time step k = 1 millisecond. We relaxed the typical homogeneity assumption of transition probabilities to allow for the transition probabilities $p_{ij}(k)$ to be time-dependent. We used Baum-Welch algorithm [28-30] to obtain the maximum likelihood estimate of the transition probabilities $p_{ij}(k)$.

Similarly, a typical HMM framework assumes that the emission probability distributions, $q_i(z_k \mid H_k)$, are homogeneous and independent of time. We relaxed these two assumptions and allowed for dependencies on neuronal spike history and time. The observations here are $T \times 1$ vectors of spike trains, z_k composed of 1s and 0s, for discrete time steps $k = 0,1,2...$. T is the total number of time steps in an experimental trial. The emission probability not only depends on the cognitive state, but also the previous neural spiking activity history, i.e.,

$$p(z_k \mid x_k = i, z_{[0:t-1]}) = q_i(z_k \mid H_k^S) \tag{5.1}$$

where $H_k^S \equiv \{z_0, z_1, ..., z_{k-1}\}$ represents the neural spiking activity history.

We also constructed TI models using both neuronal and kinematics data (Fig. 3B). In contrast to the TI models using neuronal data alone, the emission probability distribution here depends on not only the previous neural spiking activity history, but also the previous kinematics measurement history, i.e.,

$$p(z_k \mid x_k = i, z_{[0:t-1]}, m_{[0:t-15]}) = q_i(z_k \mid H_k^S, H_k^K) \tag{5.2}$$

where $H_k^S \equiv \{z_0, z_1, ..., z_{k-1}\}$ and $H_k^K \equiv \{m_0, m_1, ..., m_{k-15}\}$ represent the neural spiking activity and kinematics history, respectively. The kinematics observations here are $T \times 1$ vectors of velocity values, m_k composed of continuous real values for discrete time $k = 0,1,2...$. T is the total number of time steps in an experimental trial.

5.2.3 Point Process Observation Models

When spike trains were the observations, the output probability densities were characterized using point process models by conditional intensity functions (CIF). To define a point-process model of neural spiking activity, in this analysis we consider an observation interval (0, *T*] and let *N*(*k*) be the number of steps counted in interval (0, *k*] for *k* ϵ (0, *T*]. A point-process model of a neural spike train can be completely characterized by its CIF, $\lambda^i\left(k \mid H_k^S\right)$, defined as follows:

$$\lambda^i\left(k \mid H_k^S\right) = \lim_{\Delta \to 0} \frac{\Pr\left(N^i\left(k+\Delta\right) - N^i\left(k\right) = 1 \mid H_k^S\right)}{\Delta} \tag{5.3}$$

where H_k^S denotes the history of neural spiking activity up to the time *k*. It follows from equation (3) that the probability of a single spike in a small interval $\left(k, k+\Delta\right]$ is approximately

$$Pr(\text{spike in } \left(k, k+\Delta\right] \mid H_k^S) = \lambda^i\left(k \mid H_k^S\right)\Delta \tag{5.4}$$

Details of point process models with history dependency can be found in [22, 23]. When Δ is small, equation (4) approximates the spiking propensity at the time step *k*. With a given CIF, the emission probability of the HMM is:

$$q_i\left(z_k \mid H_k^S\right) = \prod_{j=1}^{N}\left[1 - \lambda_j^i\left(k \mid H_k^S\right)\Delta\right]^{1-z_k(j)}\left[\lambda_j^i\left(k \mid H_k^S\right)\Delta\right]^{z_k(j)} \tag{5.5}$$

For our analyses, we used generalized linear models (GLMs) to define the CIF by expressing, for each neuron, the log of its CIF in terms of the neural spiking activity history [17]. Specifically, the GLM for each CIF has the following structure:

$$\log \lambda^i\left(k \mid H_k^S\right) = \gamma_0^i + \sum_{j=1}^{M} \gamma_j^i n_{k-j}^i \tag{5.6}$$

where γ_0 relates to the background level of neural spiking activity, M is the order of autoregressive process, γ_j represents the autoregressive coefficients, and n_j represents the neural spiking history of the neuron of interest at each selected history window j.

We also formulate a point-process model to relate the spiking propensity of each neuron to factors associated with both the neuronal spiking history *and* kinematics measurement. The CIF is defined as follows:

$$\lambda^i\left(k \mid H_k^S, H_{k-t_k}^K\right) = \lim_{\Delta \to 0} \frac{\Pr\left(N^i\left(k+\Delta\right) - N^i\left(k\right) = 1 \mid H_k^S, H_{k-t_k}^K\right)}{\Delta} \tag{5.7}$$

where H_k^S is the neural spiking activity history, $H_{k-t_k}^K$ is the corresponding kinematics history, and t_k is the built-in delay mimicking the delay in collecting kinematics information in neural prosthetics (t_k = 15 milliseconds).

We used the kinematics data for the cognitive state transition from movement to holding while the subjects were moving. Furthermore, from the recorded experimental videos and kinematics data, we found that the experiment subjects had slight hand rotation movements before releasing the home object, which is mechanically registered as OM. So, we also used kinematics data for the cognitive state transition from planning to movement.

Similarly, we also used a GLM to define our CIF, defined as follows:

$$\log \lambda^i\left(k \mid H_k^S, H_{k-t_K}^K\right) = \gamma_0^i + \sum_{j=1}^{J_S} \gamma_j^i n_{k-j}^{S(i)} + \sum_{j=1+J_S}^{J_K} \gamma_j^i n_{k-j+J_S}^{K(i)} \tag{5.8}$$

where γ_0 relates to the background level of neural spiking activity, J_s and J_k are the orders of autoregressive process, γ_j represents the autoregressive coefficients, n_j^S represents the spiking history of the neuron of interest at each selected history window j, and n_j^K represents the kinematics history at selected history window j.

5.2.4 Model Goodness-of-fit and Model Selection

Establishing the degree of agreement between a point-process model and observations of spike trains and associated experimental variables is a prerequisite for using the point-process analysis to make scientific

inferences. We used Kolmogorov–Smirnov (KS) plots based on the time-rescaling theorem to assess the model goodness-of-fit. The time-rescaling theorem is a well-known result in probability theory, which states that any point process with an integrable CIF may be transformed into a Poisson process with unit rate [30]. A KS plot, which outlines the empirical cumulative distribution function (CDF) of the transformed spike times versus the CDF of a unit rate exponential, was used to visualize the goodness-of-fit for each model. The model is better if its corresponding KS plot lies near the 45º line. We computed 95% confidence bounds for the degree of agreement using the distribution of the KS statistic [30].

In order to compare and select models (i.e. select the order of autoregressive process J_S) for each neuron, we calculated Akaike's standard information criterion (AIC) [31]. The model with the smallest ACI value and KS plots within 95% confidence bound of the 45º line was chosen to represent each neuron.

5.2.5 Neural State Transition Decoding

For each new neural spike train $Z = z_1 z_2 z_3 ... z_T$, we would like to know the optimal cognitive state sequence

$$X = x_1 x_2 x_3 ... x_T \tag{5.9}$$

which generated the observation sequence. The optimality criterion we employed was to maximize $P(X \mid Z, \lambda)$, the probability of a cognitive state sequence $X = x_1 x_2 x_3 ... x_T$ given the model λ and the neural spike trains $Z = z_1 z_2 z_3 ... z_T$. According to the Bayes Rule, we have

$$P(X \mid Z, \lambda) = \frac{P(X, Z \mid \lambda)}{P(Z \mid \lambda)} \propto P(X, Z \mid \lambda) \tag{5.10}$$

So equivalently, the optimality criterion is to maximize

$$P(X, Z \mid \lambda) = P(x_1 x_2 x_3 ... x_T, z_1 z_2 z_3 ... z_T \mid \lambda) \tag{5.11}$$

In the offline mode, we employed the Viterbi algorithm [32, 33] using dynamic programming techniques to find the single best hidden discrete state sequence $\bar{X} = x_1 x_2 x_3 ... x_T$.

In order to further understand performance limits of our proposed framework, we implemented a real-time (online) cognitive state transition detection framework by using the Bayesian information state variable estimation [34]

$$\pi_k^i = P\left(x_k = i \mid z_k, H_k\right) \tag{5.12}$$

which is the *posterior* probability of being in the state $x = i$ at the time *k* given the measurements up to and including the time *k*. Here, the detection threshold was set as 0.5. That is, we detected a state transition when $\pi_k > 0.5$. The detailed deviation of the Bayesian information state variable can be found in [35, 36].

5.2.6 Detection Latency Measurements

In order to quantify the detection performance of both the TI and TD models, we measured the detection latencies of each model. Detection latencies were measured for cognitive state transitions from baseline to planning, from planning to movement, and from movement to holding. Here we define the detection latency (in milliseconds) for each cognitive state transition of interest as the difference $T_E - T_A$ between the estimated transition time T_E and the actual transition time T_A recorded from experiments [34].

5.2.7 Detection Accuracy Measurements

In order to further quantify the cognitive state transition detection accuracy, we define a window of width 2*W (W= 50 milliseconds) centered on each actual cognitive state transition [35]. When the estimated cognitive state transitions T_E were within the detection window of the actual transitions T_A, we counted the corresponding estimated state transitions as correct estimations, and vice versa. We calculated the detection accuracy as the percentage of correct detections on the testing dataset.

5.3 Results

5.3.1 TD vs. TI models

In both online and offline modes, the TI models have significantly better detection latency-accuracy performance than corresponding TD models during the transition from baseline to planning ($p < 0.05$) (Table I, Fig. 4A). During the transition from planning to movement, TI models with the neuronal data alone have significantly better detection latency-accuracy performance than corresponding TD models ($p < 0.05$) (Table 5-1, Fig. 4B).

During the transition from movement to holding, TI models with the combined neuronal data and kinematics data have significantly better detection latency-accuracy performance than the corresponding TD models, in both online and offline modes ($p < 0.05$) (Table 5-1).

In summary, our results suggest that the TI models achieve better performance than the TD models in detecting cognitive state transitions in both online and offline modes.

Table 5-1 Detection accuracies (percentage) of TI and TD models in online and offline modes.

		TI (N)	TI (N&K)	Mallet	Pull	Push	Sphere
CUE	Online	93%	--	83%	74%	77%	71%
	Offline	91%	--	80%	78%	81%	79%
OM	Online	96%	94%	61%	63%	66%	65%
	Offline	98%	92%	59%	51%	62%	60%
SC	Online	95%	97%	88%	90%	91%	85%
	Offline	92%	96%	81%	85%	87%	80%

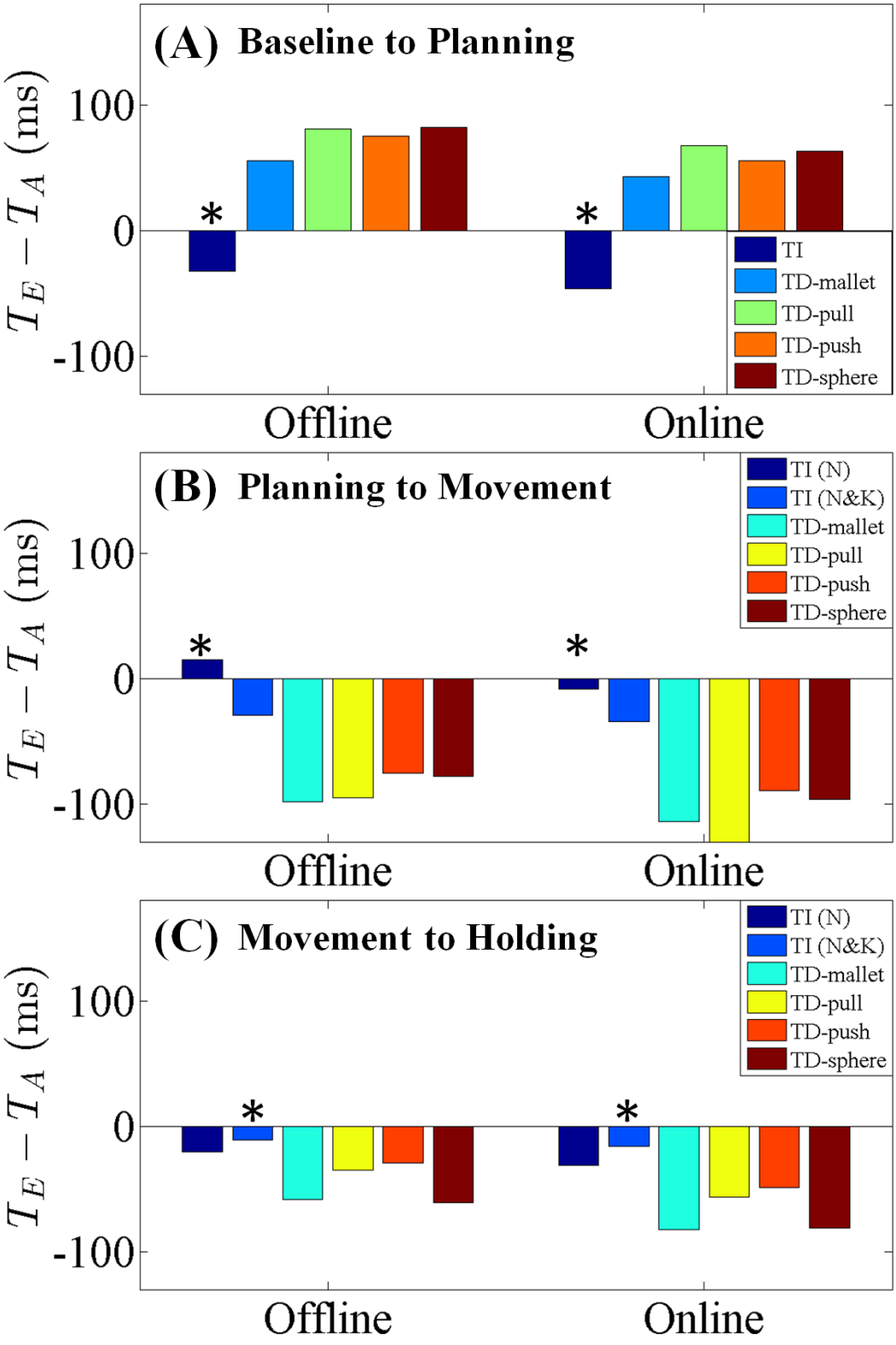


Fig. 5-2 Detection delays in TI and TD models in online and offline modes. (A) Detection latencies for the transition from baseline to planning. (B) Detection latencies for the transition from planning to movement. (C) Detection latencies for the transition from movement to holding.

5.3.2 Online vs. Offline

When a model has a negative detection latency, it indicates that the corresponding model may be predictive of the cognitive state transition. The corresponding online model is found to have a larger negative detection latency compared to the offline TI model with a negative detection latency for the transition from baseline to planning (Fig. 5-2A), indicating that the online model is more predictive. However, the detection accuracy of the online model is slightly lower than the corresponding offline TI model, because the predicted transitions from the offline TI models are closer to the actual transition times (Table 5-1). Similar trends have been observed for the TI models using both neuronal and kinematics data during the transitions from planning to movement (Fig. 5-2B) and from movement to holding (Fig. 5-2C), as well as the TI models using neuronal data alone for the transition from movement to holding (Fig. 5-2C).

When a model has a positive detection latency, it means that the corresponding model has delays in detecting cognitive state transitions. For the TI model using neuronal data alone, the offline TI model has a positive detection latency (Fig. 5-2B). The corresponding online TI model, however, has a negative detection latency (Fig. 5-2B) and an improved detection accuracy (Table 5-1).

Despite that the online and offline TI models have differences in detection latencies and accuracies, our statistical test results show that online TI models have comparable detection latency-accuracy performance to offline TI models.

5.3.3 Models With Kinematics vs. Without Kinematics

During the transition from movement to holding, TI models with the combined neuronal and kinematics data have significantly better detection latency-accuracy performance than TI models with neuronal data alone ($p < 0.05$) in both the offline and online modes (Table 5-1, Fig. 5-2C). In contrast, the TI models with neuronal data alone have better latency-accuracy performance than the TI models with both neuronal data and kinematics data for the transition from planning to movement in both the offline and online modes (Table 5-1, Fig. 5-2B). Note that the kinematics data in the planning state were available only for a very short period of time (the average duration < 10 ms) - right before the recorded transition from planning to movement; in contrast, the kinematics data were available throughout the movement state (the average duration = 1290

ms). Our results may suggest that only when there is sufficient kinematics data available, the addition of kinematics data may improve the detection performance of TI models.

5.4 Discussions and Conclusions

Cognitive state transition detection is a popular topic in neuroscience and medicine [21, 38, 39]. However, these applications detected the transition points offline and also chose the detection threshold based on the knowledge of the entire neuronal activity. Yu recently applied the quickest detection framework [21] to detect changes in the neuronal spiking rates in the online mode. But the model in [21] assumed that the transition times followed a geometric distribution and the observations were independent and identically distributed. Our proposed task-independent cognitive state transition detection framework generalizes the online cognitive state transition detection problem because it (i) allows temporal dependency among neighboring observations, making it more applicable for practical neuroscience applications [35, 40], (ii) does not make any assumptions or requirements of the cognitive state transition times, and (iii) is independent of movements and easily scalable, which is necessary for the application in a clinical viable neural prosthetics.

The proposed TI cognitive state transition detection framework is different from the other approaches [21, 38, 39]; it is adaptive, i.e., the number of tasks does not need to be known as *a priori* nor chosen *a posteriori* based on the modulation of some statistics. Furthermore, our proposed framework automatically adapts to different patterns of sequential neuronal observations without changing model structure. Our proposed framework uses point process models with history dependencies to model each neuron. Even when there is no clear difference in the spiking activities between the neighboring states for each neuron across different trials, the point process models can capture the hidden neuronal dynamics and provide one set of parameters for each neuron for each state across different trials. The point process models may also be predicative of temporal dynamics when there are changes in the neuronal activity before the transition times. For example, during the transition from planning to movement, there is an increase in neuronal activity for some neurons (Fig. 5-3A). The Bayesian information state variable π_k thus increases above the threshold of 0.5 before the state transition time (Fig. 5-3B), predicting the transition before it actually occurs.

We finally note that, our framework adapts to different applications and observations using a time-varying, history-dependent HMMs. The HMMs in our proposed framework can be estimated offline using training data. Previous approaches often set the number of hidden states in modeling each cognitive state as *a priori* [13, 36, 37]. Here the proposed framework was able use only one hidden state for each cognitive state with history dependencies to customize to the observations, simplifying model structures and parameters. Our framework could pave the way for a TI control of prosthesis from cortical neurons.

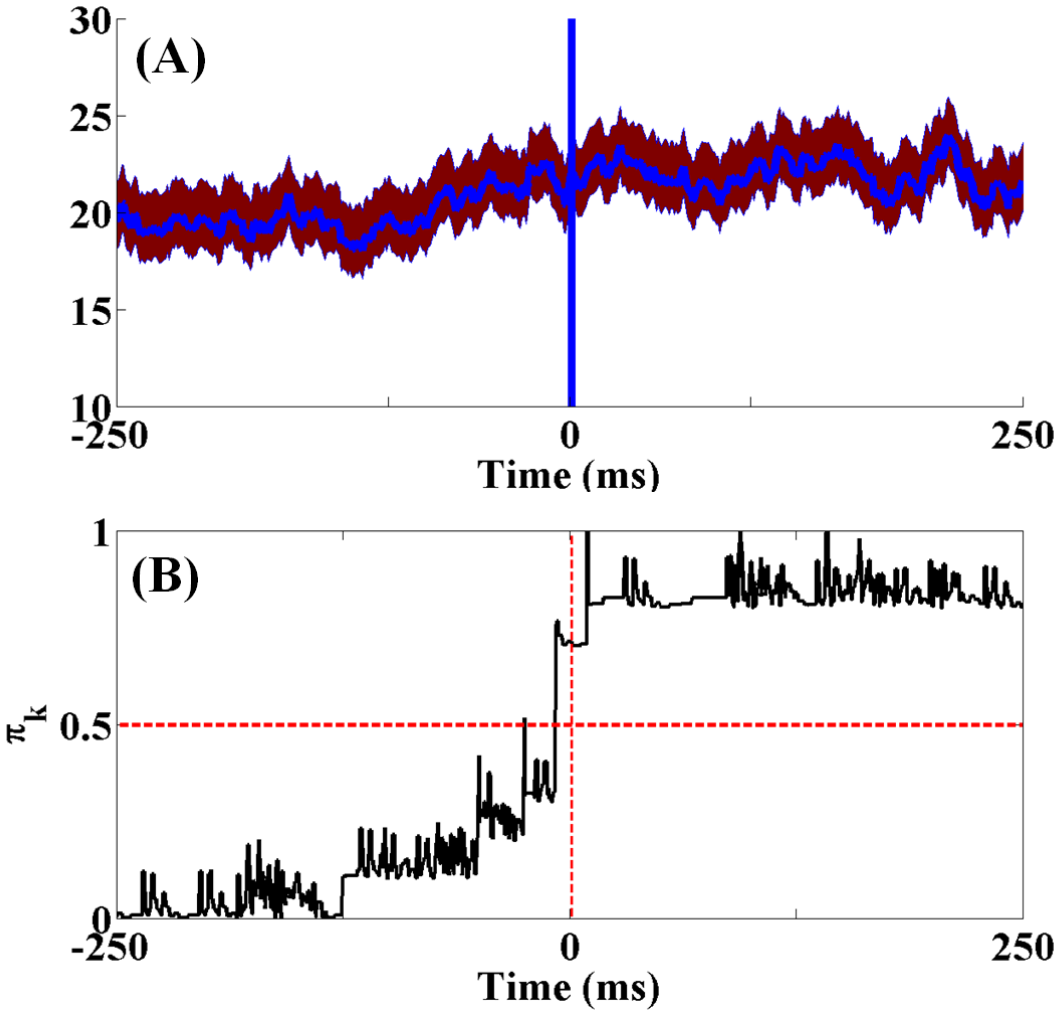


Fig. 5-3 (A) Example of a PM neuron histogram with 95% confidence bound (red) during movement tasks aligned such that 0 denotes the movement onset (blue). (B) The Bayesian information state variable (black) for a single trial from (A). Time scale in (B) also applies to (A).

5.5 Bibliography

[1] E. E. Fetz, "Real-time control of a robotic arm by neuronal ensemble," *Nature,* vol. 2, pp. 583-584, 1999.

[2] J. M. Carmena, M. A. Lebedev, R. E. Crist, E. O. Joseph, D. M. Santucci, D. F. Demitrov, P. G. Patil, C. S. Henriquez and M. A. L Nicolelis, "Learning to control a brain-computer interface for reaching and grasping by primates," *PLoS BioL,* vol. 1, pp. 193-208, 2003.

[3] J. P. Donoghue, "Connecting cortex to machines: recent advances in brain interface," *Nature Neurosci,* vol. 5, pp. 19085-1088, 2002.

[4] M. A. Nicolelis, "Actions from thoughts," *Nature,* vol. 409, pp. 403-407, 2001.

[5] G. Santhenam, S. I. Ryu, B. M. Yu, A. Afshar and K. V. Shenoy, "A high-performance brain-computer interface," *Nature,* vol. 442, pp. 195-198, 2006.

[6] A. B. Schwartz, "Cortical neuronal prosthetics," *Annual. Rev. Neurosci.,* vol. 27, pp. 487-505, 2004.

[7] S. H. Scott, "Neuroscience: converting thoughts into actions," *Nature,* vol. 416, pp. 141-142, 2006.

[8] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows and J. Donoghue, "Instant neural control of a movement signal," *Nature,* vol. 416, pp. 141-142, 2002.

[9] D. M. Taylor, S. I. H. Tillery and A. B. Schwarts, "Direct cortical control of 3D neuroprosthetic devices," *Science,* vol. 296, pp. 1829-1832, 2002.

[10] S. Musallam, B. D. Corneil, H. Scherberger and R. A. Andersen, "Cognitive control signals for neural prosthetics," *Science,* vol. 305, pp. 258-262, 2004.

[11] T. M. Michell, R. Huchinson, R. S. Niculescu, F. Pepeira and X. Wang, "Learning to Decode Cognitive States From Brain Images," *Mach Learn,* vol. 57, pp. 145-175, 2004.

[12] M. Mollazadeh, V. Aggrawal, A. G. Davidson, A. J. Law, N. V. Thakor and M. H. Schieber, "Spatiotemporal variation of multiple neurophysiological signals in the primary motor cortex during dexterous reach-to-grasp movements," *J. Neurosci.,* vol. 31, pp. 15531-15543, 2011.

[13] C. Kemere, G. Santhanam, B. M. Yu, S. I. Ryu, T. H. Meng and K. V. Shenoy, "Model-based decoding of reaching movements for prosthetic systems," *Proc IEEE Eng. Med. Biol. Soc.,* pp. 4524-4528, 2004.

[14] C. Kemere, K. V. Shenoy and T. H. Meng, "Model-based neural decoding of reaching movements: a maximum likelihood approach," *IEEE Trans. Biomed. Eng.,* vol. 51, pp. 925-932, 2004.

[15] B. M. Yu, C. Kemere, G. Santhanam, A. Afshar, S. I. Ryu, T. H. Meng, M. Sahani and K. V. Shenoy, "Mixture of trajectory models for neural decoding of goal directed movements," *J Neurophysiol,* vol. 97, pp. 3763-3780, 2007.

[16] G. Radons, J. D. Becker, B. Dulfer and J. Kruger, "Analysis, classification, and coding of multielectrode spike trains with hidden Markov models," *Biol. Cybern.,* vol. 71, pp. 359-373, 1994.

[17] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue and E. N. Brown, "A point process framework for relating neural spiking acitivity to spiking history, neural ensemble, and extrinsic covariate effects," *J. Neurophysiol.,* vol. 93, pp. 1074-1089, 2005.

[18] A.-C. Camproux, F. Saunier, G. Chouvet, J.-C. Thalabard and G. Thomas, "A hidden Markov approach to neuron firing patterns," *Biophys. J.,* vol. 71, pp. 2404-2412, 1996.

[19] W. Wu, Y. Gao, E. Bienenstock, J. P. Donoghue and M. J. Black, "Bayesian population coding of motor cortical activity using a Kalman filter," *Neural Comput.,* vol. 18, pp. 80-118, 2006.

[20] W. Wu, J. Kulkarni, N. Hatsopoulos and L. Paninski, "Neural decoding of goal-directed movements using a linear statespace model with hidden states," *IEEE Trans. Biomed. Eng.,* vol. 17, pp. 1-9, 2009.

[21] A. J. Yu, "Optimal change-detection and spiking neurons," *Proc. 20th Annu. Conf. Neural Inf. Process. Syst,* pp. 1545-1552, 2006.

[22] D. R. Cox and V. Isham, Point processes, CRC Press, 1980.

[23] D. L. Snyder and M. I. Miller, Random point processes in time and space, Springer-Verlag, 1991.

[24] J. Moeller and R. Waagepetersen, Statistical inference and simulation for spatial point processes, Chapman Hall, 2004.

[25] J. Moeller, A. Syversveen and R. Waagepetersen, "Log-Gaussian Cox processes," *Scand. J. Stat.,* vol. 25, pp. 451-482, 1998.

[26] J. E. Kulkarni and L. Paninski, “Common-input models for multiple neural spike-train data,” *Network,* vol. 18, pp. 375-407, 2007.

[27] S. Santaniello, E. B. Montgomery, J. T. Gale and S. V. Sarma, "Non-stationary discharge patterns in motor cortex under subthalamic nucleus deep brain stimulation," *Front Integr Neurosci,* vol. 6, pp. 1-13, 2012.

[28] S. V. Sarma, M. L. Cheng, U. Eden, Z. Williams, E. N. Brown and E. Eskandar, "The effects of cues on neurons in the basal ganglia in Parkingson's disease," *Front Integr Neurosci,* vol. 6, pp. 1-12, 2012.

[29] J. Bloit and X. Rodet, "Short-term Viterbi for Online HMM Decoding: Evaluation on a Real-time Phone Recognition Task," in *ICASSP*, Las Vegas, NV, 2008.

[30] N. L. Johnson and S. Kotz, Distribution in Statistics-Continuous Univariate Distributions, NY: John Wiley & Sons, 1970.

[31] H. Akaike, “Factor analysis and AIC,” *Psychometrika,* vol. 52, pp. 317-332, 1987.

[32] A. J. Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decodign algorithm,” *IEEE Trans. Inform. Theory,* vol. 13, pp. 260-269, 1967.

[33] G. D. Forney, “The Viterbi algorithm,” *Proc of IEEE,* vol. 61, pp. 268-278, 1973.

[34] S. Santaniello, D. L. Sherman, N. V. Thakor, E. N. Eskandar and S. V. Sarma, "Optimal Control-Based Bayesian Detection of Clinical and Behavioral State Transitions," *IEEE Trans. Neural Syst. Rehab. Eng.,* vol. 20, pp. 708-719, 2012.

[35] S. Faul, G. Boylan, S. Connolly, L. Marnane and G. Lightbody, “An evaluation of neonatal seizure detection methods,” *J. Clin. Neurophysiol.,* vol. 116, pp. 1533-`541, 2005.

[36] V. Aggarwal, S. Acharya, F. Tenore, S. Hyun-chool, R. Etienne-Cummings, M. H. Schieber and N. V. Thakor, “Asynchronous decoding of dexterous finger using M1 neurons,” *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 16, pp. 3-14, 2008.

[37] S. Acharya, F. Tenore, V. Aggarwal, R. Etienne-Cummings, M. H. Schieber and N. V. Thakor, “Decoding individual finger movements using volume constrained neuronal ensembles in the M1 hand area,” *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 16, pp. 15-23, 2008.

[38] L. E. Baum and J. A. Eagon, “An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology,” *Bull. Amer. Math. Soc.,* vol. 3, pp. 360-363, 1967.

[39] L. E. Baum and T. Petrie, “Statistical inference for probabilistic functions of finite state Markov chains,” *Ann. Math. Stat.,* vol. 6, pp. 1554-1563, 1966.

[40] L. E. Baum, T. Petrie, G. Soules and N. Weiss, “A maximization technique occuring in the statistical analysis of probabilitic functions of Markov chains,” *Ann. Math. Stat.,* vol. 1, pp. 164-171, 1970.

[41] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue and E. N. Brown, “A point process framework fo relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects,” *J. Neurophysiol.,* vol. 93, pp. 1074-1089, 2005.

[42] J. Nelder and R. Wedderburn, "Generalized Linear Models," *Journal of the Royal Statistical Society,* vol. 135, no. 3, p. 370–384, 1972.

[43] A. Dempster, N. Laird and D. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J.R.Stat.Soc.,* vol. 39, pp. 1-38, 1977.

[44] A. C. Smith and E. N. Brown, “Estimating a state-space model from point process observations,” *Neural Computation,* vol. 15, pp. 965-993, 2003.

[45] A. C. Smith, L. M. Frank, S. Wirth, M. Yanike, D. Hu, Y. Kubota, A. M. Graybiel, W. Suzuki and E. N. Brown, “Dynamic analysis of learning in behavioral experiments,” *J. Neurosci. ,* vol. 24, pp. 447-461, 2004.

[46] M. Basseville and I. V. Nikiforov, Detection of abrupt changes: theory and changes, NJ: Prentice Hall, 1993.

[47] U. Eden, L. Frank, R. Barbieri, V. Solo and E. Brown, "Dynamic Analysis of Neural Encoding by Point Process Adaptive Filtering," *Neural Computation,* vol. 16, pp. 971-998, 2004.

[48] M. Basseville and I. V. Nikiforov, Detection of Abrupt Changes: Theory and Changes, NJ: Prentice Hall, 1993.

[49] J. W. Pillow, Y. Ahmadian and L. Paninski, "Model-based decoding, information estimation, and change-point detection techniques for multineuron spike trains," *Neural Comput.,* vol. 23, pp. 1-45, 2011.

[50] S. V. Sarma, U. T. Eden, M. L. Cheng, Z. M. Williams, R. Hu, E. Eskansar and E. N. Brown, "Using point process models to compare neural spiking activity in the subthalamic nucleus of Parkinson's patients and a healthy primate.," *IEEE Trans Biomed Eng,* vol. 57, pp. 1297-1305, 2010.

[51] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Stat.,* vol. 6, pp. 1554-1563, 1966.

[52] L. E. Baum and J. A. Eagon, "An ineqality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bull. Amer. Math. Soc.,* vol. 3, pp. 360-363, 1967.

[53] L. E. Baum, T. Petrie, G. Soules and N. Weiss, "A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.,* vol. 1, pp. 164-171, 1970.

[54] J. Ushiba, Y. Tomita, Y. Masakado and Y. Komune, "A cumulative sum test for a peri-stimulus time histogram using the Monte Carlo method," *J. Neurosci. Meth.,* vol. 118, pp. 207-214, 2002.

[55] J. Zhuang, W. Truccolo, C. Vargas-Irwin and J. P. Donoghue, "Decoding 3D reach and grasp kinematics from high-frequency local field potentials in primate primary motor cortex," *IEEE Trans Biomed Eng.,* vol. 57, no. 7, pp. 1774-1784, 2010.

[56] C. Mehring, J. Rickert, E. Vaadia, S. Oliveira, A. Aertsen and S. Rotter, "Inference of hand movements from local field potentials in monkey motor cortex," *Nat Neurosci,* vol. 6, no. 12, pp. 1253-1254, 2003.

[57] G. Schalk, J. Kubanek, K. J. Miller, N. R. Anderson, E. C. Leuthardt, J. G. Ojemann, D. Limbrick, D. Moran, L. A. Gerhardt and J. R. Wolpaw, "Decoding two-dimensional movement trajectories using electrocorticographic signals in humans," *J. Neural Eng.,* vol. 4, pp. 264-275, 2007.

[58] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows and J. P. Donoghue, "Brain-machine interface: instant neural control of a movement signal," *Nature,* vol. 416, no. 6877, pp. 141-142, 2002.

[59] J. Wessberg, C. R. Stambaugh, J. D. Kralik, P. D. Beck, M. Laubach, J. K. Chapin, J. Kim, S. J. Biggs, M. A. Srinivasan and M. A. Nicolelis, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature,* vol. 408, no. 6810, pp. 361-365, 2000.

[60] E. Stark and M. Abeles, "Predicting movement from multiunit activity," *J Neurosci,* vol. 27, no. 31, pp. 8387-8394, 2007.

[61] J. R. Wolpaw and D. J. McFarland, "Control of of a two-dimensional movement signal by a noninvasive brain-computer interface in humans," *Proc. Natl Acad. Sci.,* vol. 101, pp. 17849-17854, 2004.

[62] A. P. Georgopoulos, A. B. Schwartz and R. E. Kettner, "Neuronal population coding of movement direction," *Science,* vol. 233, pp. 1416-1419, 1986.

[63] D. M. Taylor, S. I. Tillery and A. B. Schwartz, "Direct cortical control of 3D neuroprosthetic devices," *Science,* vol. 296, pp. 1829-1832, 2002.

[64] M. A. Lebedev, J. M. Carmena, J. E. O'Doherty, M. Zacksenhouse, C. S. Henriquez, J. C. Principe and M. A. Nicolelis, "Cortical ensemble adaptation to represent velocity of an artificial actuator controlled by a brain-machine interface," *J. Neurosci.,* vol. 25, pp. 4681-4693, 2005.

[65] J. P. Donoghue, A. Nurmikko, G. Friehs and M. Black, "Development of neuromotor prostheses for humans," *Suppl. Clin. Neurophysiol.,* vol. 57, pp. 592-606, 2004.

[66] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. A. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn and J. P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature,* vol. 442, pp. 164-171, 2006.

[67] W. Wu, Y. Gao, E. Bienenstock, J. Donoghue and M. Black, "Bayesian Population Decoding of Motor Cortical Activity Using a Kalman Filter," *Neural Computation,* vol. 18, pp. 80-118, 2006.

[68] X. Kang, S. Sarma, S. Santaniello, M. Schiber and N. V. Thakor, "Task independent cognitive state transition detection from cortical neurons," *IEEE Trans. on Neural Syst. and Rehab. Engi.,* 2014.

Chapter 6 Task Independent Kinematics Decoding

6.1 Introduction

6.2 Methods

6.2.1 Experimental Set-up

6.2.2 Encoding Models

6.2.2.1 Point Process Models of Neuronal Observations

6.2.2.2 Autoregressive Models of Kinematics States (State Evolution)

6.2.3 Cognitive State Detection

6.2.4 Task-Independent Decoding Algorithm

6.2.5 Task-Dependent Decoding Algorithm

6.2.6 Model Goodness-of-fit and Model Selection

6.3 Results

6.4 Discussion and Conclusions

6.5 Bibliography

6.1 Introduction

There are many decoding algorithms that have been developed for different types of neural data: local field potentials (LFPs) [1, 2], Ecog [3], EEG [4], and spike trains [5-11]. Despite the development of numerous algorithms to decode kinematics from neural activity, current approaches have two main drawbacks. Firstly, current decoding systems use task-dependent (TD) encoding models. That is, a different encoding model is constructed for each task observed during the training session. Consequently, the parameters of the decoding algorithms are different depending on the movement task, and grow very large in numbers in a more general setting with multiple tasks. Furthermore, these TD decoding systems may only perform well on the training movement tasks, and not on unobserved movement tasks. Clinically practical prosthetics need to handle hundreds if not thousands of natural movements, and decoding algorithms must be able to operate accurately even if movements are not included in the model training sets.

The second drawback with current decoding algorithm is that the kinematics are usually decoded without taking into account cognitive states and their transitions, even though the neural dynamics (and thus encoding models) may differ across cognitive states. Cognitive states are latent neural states that occur between the beginning and completion of a movement task, such as planning or moving. Cognitive states are often independent of the detailed movements in a specific task. For example, in 3D reach-to-grasp movements with multiple target objects, the cognitive state sequences for movements with different target objects are the same, which include baseline (BASE), planning (PLAN), moving (MOVE), and holding (HOLD). The differences between the movement tasks with different objects lie in the kinematic *execution* of the movements.

We hypothesize that kinematics can be decoded in a task-independent (TI) manner, with equivalent if not better, decoding performance as compared to TD decoding algorithms. Here, our proposed TI encoding model is trained in three out of four movement tasks, leaving one movement task out for testing the proposed decoding algorithm, while TD encoding models are each trained for each of the movement tasks. Furthermore, we hypothesize that the addition of cognitive state transition information in our encoding models will improve decoding performance. In order to test our hypotheses, we use a decoding algorithm

based on point process encoding models of neuronal spiking activity and an autoregressive model to represent the subjects' kinematics. The point process models represent the neuronal spiking activity of each neuron as a generalized Poisson process with dependency on the spiking history of the neuron as well as kinematics. The linear state-space model of kinematics states is represented as a first-order autoregressive process with cognitive-state dependent state matrices. We assess the decoding accuracy by comparing the estimated kinematics with the actual kinematics and calculating the average mean squared errors, and also by comparing its performance with TD decoding algorithms. The TD decoding algorithms are using with the same decoding algorithm based on point process models and linear state-space models, except all TD encoding models do not depend on cognitive states.

6.2 Methods

6.2.1 Experimental Set-up

The experimental set-up and neural data recording are the same as Chapter 5 Section 5.1. The corresponding kinematics were recorded using 30 3D positional sensors mounted on the non-human subjects' fingers, wrist, and hand at 200Hz (Fig. 3) [12].

In the following subsections, we present our encoding models and decoding algorithm. First, we introduce the encoding models, which consist of a state (kinematics) evolution equation and an observation equation. The state evolution equation is an autoregressive model of the kinematics. The observations are modeled as point processes as they represent the observed neuronal spike trains. Thus, the output equation is the probability of the observed spikes, which can be computed from the point process models. Finally, we describe the decoding algorithm used to estimate the kinematics states based on the neuronal spike train observations.

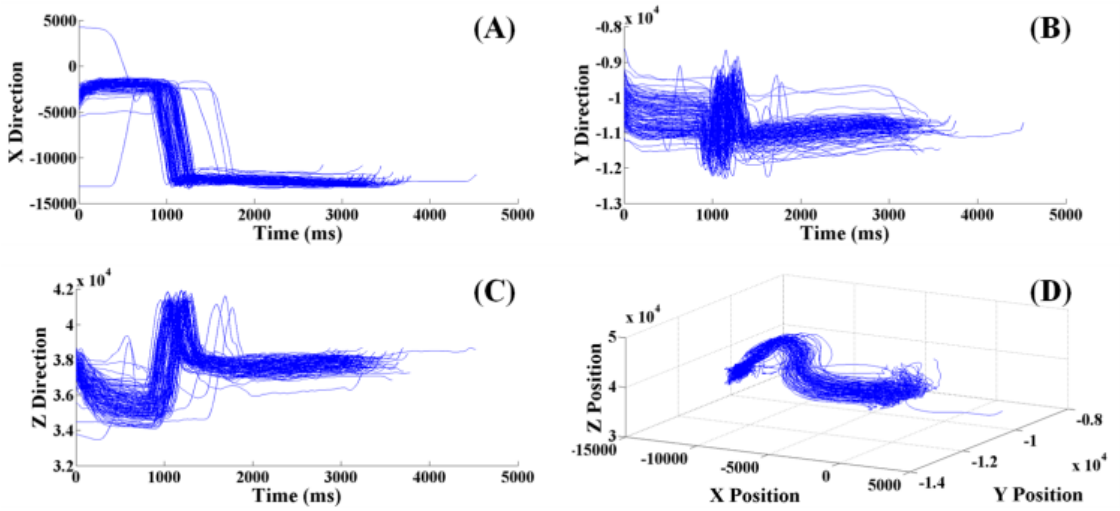


Fig. 6-1 Recorded positional kinematics in the experiments. (A) X-directional positional data; (B) Y-directional positional data; (C) Z-directional position data; (D) 3D positional kinematics.

6.2.2 Encoding Models

There are four movement tasks including pushing a button (#trials=62137), pulling a cylinder (#trials=58658), pulling a mallet (#trials=60423), and grasping a sphere (#trials=60538) from the same two monkeys mentioned in Chapter 5. When constructing the TI models, mutually exclusive feature sets including neuronal spike train observations and kinematics states for three of the four movement tasks were used for training and the fourth movement task was used for testing. We repeated leaving-one-out experiment four times and calculated the average mean squared errors between actual and decoded kinematics.

6.2.2.1 Point Process Models of Neuronal Observations

We assume that the neuronal spike trains are observations of a hidden Markov model (HMM), whose underlying states represent the hidden cognitive states during a movement task. The HMM is also used for real-time cognitive state transition detection during decoding as described [35] (Fig. 4). The HMM has four hidden states to represent the four cognitive states in our experiments, including BASE ($z_k = 1$), PLAN ($z_k = 2$), MOVE ($z_k = 3$), and HOLD ($z_k = 4$). All trials began with BASE, so we set the initial probability of being in BASE $\Gamma_0 = P(z_0 = 1) = 1$, and 0 for all the other cognitive states. The initial state does not need to start with BASE. Our proposed framework can be easily extended for more natural scenarios. For example,

if the subject does not start in the idle state, then the priors Γ_i for $i = 1,2,3,4$, can be set so that the subject has some probability of starting in any of the given state. Then one can implement state detection in the same manner as described in [35].

Here, a time step k = 1 millisecond. When estimating the transition probabilities $p_{ij}(k) = P(z_k = j \mid z_{k-1} = i)$ for all time steps k and cognitive states $i, j \in \{1,2,3,4\}$, we relaxed the typical homogeneity assumption and allowed for the transition probabilities $p_{ij}(k)$ to be time-dependent. The Baum-Welch algorithm [38, 39, 40] was used to obtain the maximum likelihood estimate of the transition probabilities $p_{ij}(k)$.

When estimating the emission probability distributions $q(y_k \mid \theta, H_k)$, we used a stochastic point process model, which incorporated the effects of spike history and kinematics states [22, 23, 41, 30].

A stochastic point process can be fully defined by its conditional intensity function (CIF), $\lambda_c\left(k \mid \theta, H_k^c\right)$, as follows:

$$\lambda_c\left(k \mid \theta, H_k^c\right) = \lim_{\Delta->0} \frac{P\left(N_c(k+\Delta) - N_c(k) = 1 \mid \theta, H_k^c\right)}{\Delta} \tag{6.1}$$

where c denotes the neuron index, for $c = 1,2,3,...,C$, where C is the total number of neurons. As mentioned previously, k represents the time step, where $k = 1,2,3,...,T$ and T is the total number of time steps observed in a trial. $N(t)$ denotes the number of spikes counted in the time interval $(0,t]$ for $t \in (0,T]$, and θ denotes the unknown model parameters to be estimated from data. H_k^c entails the neuron's spiking history to up time t in addition to the kinematic covariates described below.

The joint probability mass function of a discretized spike train can be expressed here as a product of conditional independent Bernoulli events as follows:

$$q(y_k \mid \theta, H_k) = \prod_{c=1}^{C}\left[1 - \lambda_c\left(k \mid \theta, H_k^c\right)\Delta\right]^{1-y_k(c)}\left[\lambda_c\left(k \mid \theta, H_k^c\right)\Delta\right]^{y_k(c)} \tag{6.2}$$

where y_k is a spike train vector with the dimension of $C \times 1$ at the time step k. The equation can be further expressed as follows:

$$q(y_k \mid \theta, H_k) = \exp\left\{\sum_{c=1}^{C} \log\left[\lambda_c\left(k \mid \theta, H_k^c\right)\Delta\right] y_k(c) - \sum_{c=1}^{C} \lambda_c\left(k \mid \theta, H_k^c\right)\Delta\right\} \tag{6.3}$$

We then formulate a generalized linear model (GLM) [42] for the CIF as a linear combination of the parameters that incorporate the influence of spiking history and kinematics states for each neuron as follows:

$$\log \lambda\left(k \mid H_k\right) = \gamma_0 + \sum_{j=1}^{J} \gamma_j n_{k-j} + m x_{k+\tau} \tag{6.4}$$

where γ_0 relates to the background level of neural spiking activity, J ($J = 5$) is the number of spike history covariates computed from the 80 msec time interval $\left[k-80,\quad k\right]$, γ_j represents the coefficients for the spike history covariates, n_{k-j} represents the number of spikes in the time interval $\left[k-\frac{80j}{J},\quad k-\frac{80\left(j-1\right)}{J}\right]$ for the neuron of interest at each selected spike history interval j, m represents a 6×1 vector of coefficients for each kinematic state variable, and $x_{k+\tau}$ $\left(x_{m+\tau} \in R^{6\times 1}\right)$ represents the kinematics state vector at time step $t+\tau$. $\tau = 15$ ms is the built-in delay between the kinematics and neuronal spiking activity we chose based on the experiments. We did not try other delays and the performance of the proposed framework may improve if a different delay is chosen.

Finally, all parameters of the CIFs for all neurons (i.e., point process models) were estimated via maximum likelihood estimation, which is efficiently computed for GLMs [43, 44, 45].

6.2.3 Autoregressive Models of Kinematics States (State Evolution)

We used an AR(1) process to model and characterize the evolution of kinematics (Fig. 6-2) as follows:

$$x_k = \hat{A^{z_k}} x_{k-1} + \varepsilon_k \tag{6.5}$$

where x_k represents the kinematics states at time step k $x_k = \left[\begin{array}{cccccc} p_x & p_y & p_z & v_x & v_y & v_z \end{array}\right]_k^T$ which are the positions and velocities in the three-dimensional Cartesian coordinate space. $\hat{z_k}$ represents the estimated cognitive state from the HMM. $\hat{A^{z_k}}$ denotes the cognitive-state-dependent state matrix, and $\hat{A^{z_k}} \in R^{6\times 6}$. $\varepsilon_k \sim$ $W_\varepsilon\right)$ is the noise term given by a zero mean, 6-dimension white Gaussian vector with 6×6 covariance matrix W_ε. Both the state matrix $\hat{A^{z_k}}$ and the covariance matrix W_ε were fitted with maximum likelihood method [43, 44, 45].

6.2.4 Cognitive State Detection

In the TI decoding algorithm, the cognitive states have to be detected in real-time from neuronal spike train observations. We performed a real-time (online) implementation of the cognitive state transition detection using the Bayesian information state variable estimation [34] which computes

$$\pi_k^i = P\left(z_k = i \mid H_k\right) \tag{6.6}$$

from the HMM and spike train observations, where π_k^i is the a *posterior* probability of being in state $z_k = i$ ($i = 1,2,3,4$) at time k. The detection threshold was set as 0.5. That is, we detected a state transition to state i when $\pi_k^i > 0.5$. The detailed derivation of the Bayesian information state variable can be found in [34, 46].

6.2.5 Task-Independent Decoding Algorithm

The goal of decoding is to estimate the kinematics states with neuronal spike train observations. We use the following recursive point process filter to estimate the kinematics or state vector for each time step in the TI decoding algorithm.

1) Prediction Step:

$$x_{k|k-1} = \hat{A^{z_k}} x_{k-1|k-1} \tag{6.7}$$

$$W_{k|k-1} = \hat{A^{z_k}} W_{k-1|k-1} \hat{A^{z_k}} + W_\varepsilon \tag{6.8}$$

2) Update Step:

$$\begin{aligned} W_{k|k} = & \left[W_{k|k-1}^{-1} + \sum_{c=1}^{C} \left[\nabla \log \lambda_c \left(k \mid \theta, H_{k-\tau}^c(y), x_{k|k-1} \right) \right] \right. \\ & \lambda_c \left(k \mid \theta, H_{k-\tau}^c(y), x_{k|k-1} \right) \Delta \left[\nabla \log \lambda_c \left(k \mid \theta, H_{k-\tau}^c(y), x_{k|k-1} \right) \right]' \\ & - \sum_{c=1}^{C} \nabla^2 \log \lambda_c \left(k \mid \theta, H_{k-\tau}^c(y), x_{k|k-1} \right) \\ & \left. \left[n_{1:k}^c - \lambda_c \left(k \mid \theta, H_{k-\tau}^c(y), x_{k|k-1} \right) \Delta \right] \right]^{-1} \end{aligned} \tag{6.9}$$

$$\begin{aligned} x_{k|k} = & x_{k|k-1} + W_{k|k} \\ & \times \sum_{c=1}^{C} \left(\nabla \log \lambda_c \left(k \mid \theta, H_{k-\tau}^c(y), x_{k|k-1} \right) \right. \\ & \left. \left[n_{1:k}^c - \lambda_c \left(k \mid \theta, H_{k-\tau}^c(y), x_{k|k-1} \right) \Delta \right] \right) \end{aligned} \tag{6.10}$$

The term $\nabla\left(\nabla^2\right)$ denotes the 6-dimensional vector (6×6 matrix) of first (second) partial derivatives with respect to x_k. $W_{k|k}$ is the posterior covariance matrix of x_k. The detailed derivation of the recursive point process filter can be found in [47]. τ is the built-in delay between neuronal spike train observations and kinematics states ($\tau = 15$ ms) as mentioned above.

6.2.6 Task-Dependent Decoding Algorithm

In order to evaluate our proposed TI decoding algorithm, we also constructed a TD decoding algorithm. We used mutually exclusive feature sets of neuronal spike trains and kinematics for each movement for training (80%) and testing (20%). We then constructed point process encoding models for neuronal spike train observations that were NOT dependent on cognitive states but were different for each movement task. When constructing the AR(1) encoding model of kinematic states, the state matrix A ($A \in R^{6\times 6}$) is not dependent on cognitive states and is denoted as follows:

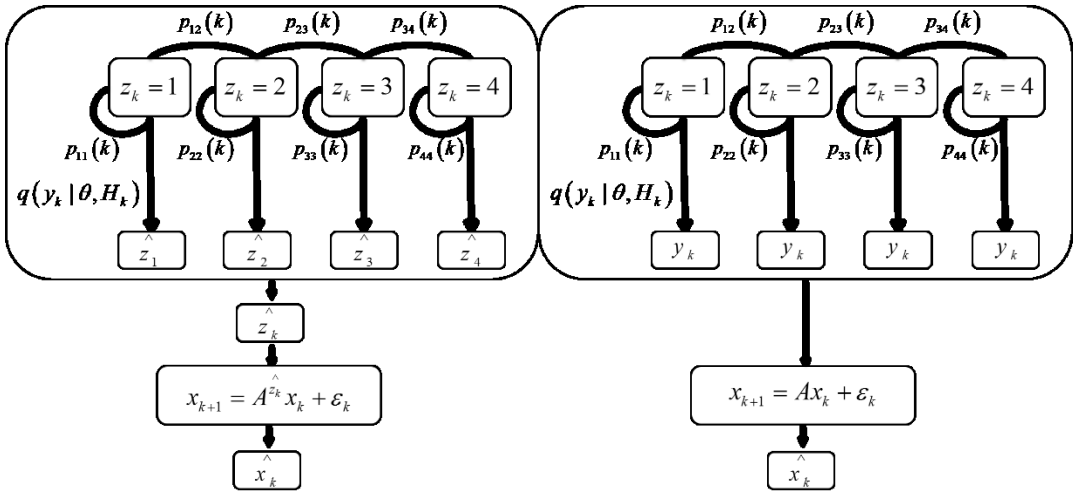
$$x_k = Ax_{k-1} + \varepsilon_k \tag{6.11}$$

$\varepsilon_k \sim$ $W_\varepsilon\right)$ is the noise term given by a zero mean, 6-dimension white Gaussian vector with 6×6 covariance matrix W_ε. Both the state matrix A and the covariance matrix W_ε are constructed via maximum likelihood estimation [43, 44, 45]. The decoding algorithm is the same as in our TI framework (*Eqs. 8-11*), except the state matrix is not cognitive-state dependent, but the point process models are task dependent.

6.2.7 Model Goodness-of-Fit and Model Selection

Establishing the degree of agreement between a point-process model and observations of spike trains and associated experimental variables is a prerequisite for using the point-process analysis to make scientific inferences. We used Kolmogorov–Smirnov (KS) plots based on the time-rescaling theorem to assess the model goodness-of-fit of each point process model. The time-rescaling theorem is a well-known result in probability theory, which states that any point process with an integrable CIF may be transformed into a Poisson process with unit rate [30]. A KS plot, which outlines the empirical cumulative distribution function (CDF) of the transformed spike times versus the CDF of a unit rate exponential, was used to visualize the

goodness-of-fit for each model. The model is better if its corresponding KS plot lies near the 45º line. We computed 95% confidence bounds for the degree of agreement using the distribution of the KS statistic [30]. In order to compare and select model classes (i.e., select the number of spike history covariates J in the GLM of the point process models) for each neuron in the point process encoding models of neuronal spike trains, we calculated Akaike's standard information criterion (AIC) [31]. The model class with the smallest AIC value across all neurons was chosen to represent each neuron.



(a) (b)

Fig. 6-2 Overview of the proposed TI framework and TD framework. (a) The TI framework including the encoding models and the decoding algorithm. The first box represents the point process encoding model of the neuronal spike trains. The first stage outputs the estimated cognitive state transitions; these are then input to the kinematics decoding algorithm. (b) The TD framework whose autoregressive models for kinematics states are not dependent on cognitive states.

6.3 Results

In order to quantify and compare our proposed TI decoding framework to a traditional TD decoding framework, we calculated the average mean squared error rate (%) between the actual and estimated kinematics (Fig. 6-4). For the TI decoding framework, the average mean squared error rate was 9.8%±1.5%. For the TD decoding framework, the average mean squared error rates for each task respectively were 14.3%, 15.8%, 13.9%, and 16.2% as shown in (Fig. 6-4). We calculated *p* values by comparing the TI algorithm with each TD algorithm, and the maximum p value is $p=0.03$. The estimated trajectories of the TI decoding framework had uniformly smaller decoding errors than those of TD decoding framework.

We also compared the estimated trajectories from our proposed TI decoding framework and TD models (Fig. 6-5). As can be seen in Fig. 6-5, TD decoding is less accurate than TI decoding, with larger delays between the estimated and actual trajectories.

The TI estimated trajectories also incur a delay of 80 milliseconds because of the fact that our point process models depend on spiking history going back 80 milliseconds.

Our TI framework uses cognitive state transition information to improve the kinematic decoding accuracy. Fig. 7 shows the transitions of the Bayesian state information variable $\pi_i = 1,2,3,4$ over time in an experimental trial.

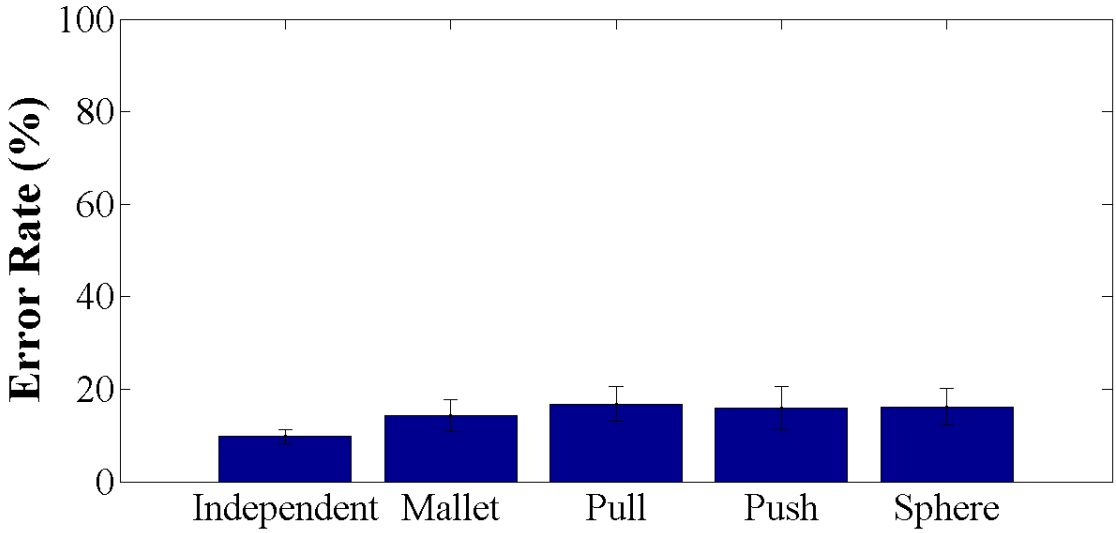
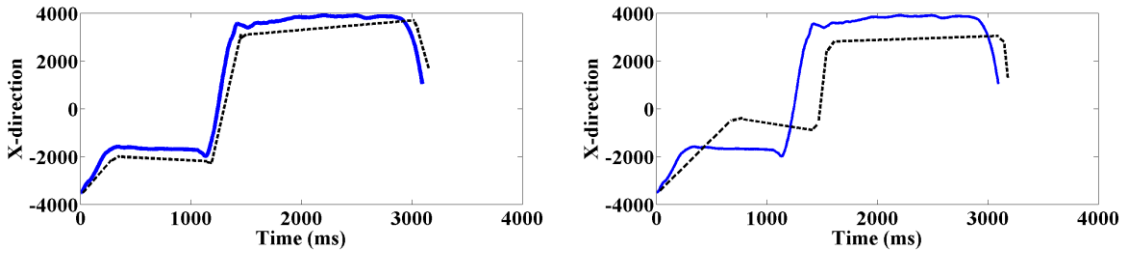
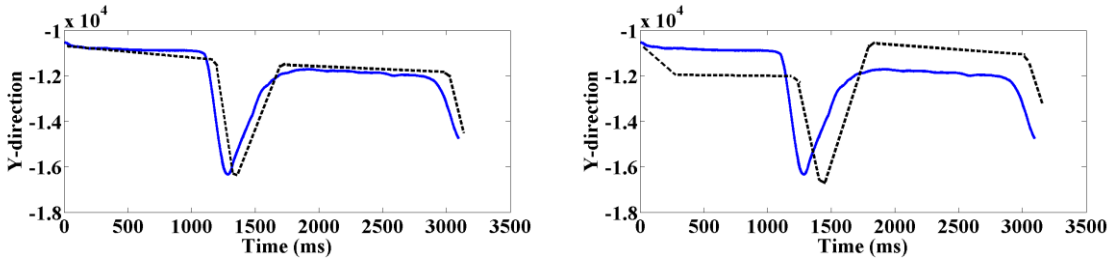


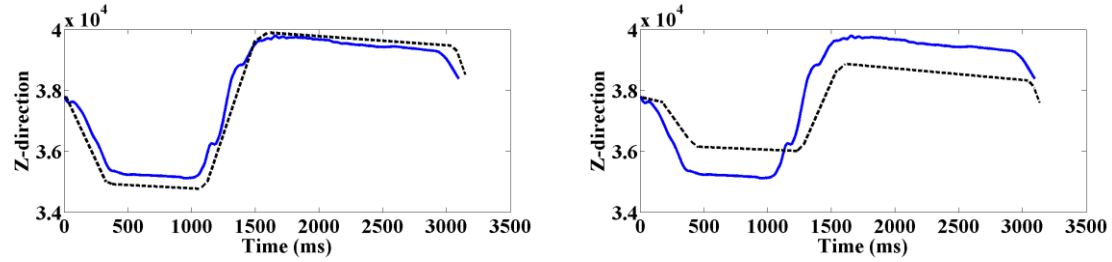
Fig. 6-3 Average mean squared decoding error rates of our proposed TI framework and four TD decoding framework. The error rate of the proposed TI decoding framework is significantly lower than TD models (maximum p=0.03).



(a)



(b)



(c)

Fig. 6-4 Example comparison of estimated trajectories using the TI decoding framework (left) and TD decoding framework (right) with the actual kinematics trajectories. (a) x-direction estimated and actual trajectories (left: TI framework; right: TD framework); (b) y-direction estimated and actual trajectories (left: TI framework; right: TD framework); (c) z-direction estimated and actual trajectories (left: TI framework; right: TD framework).

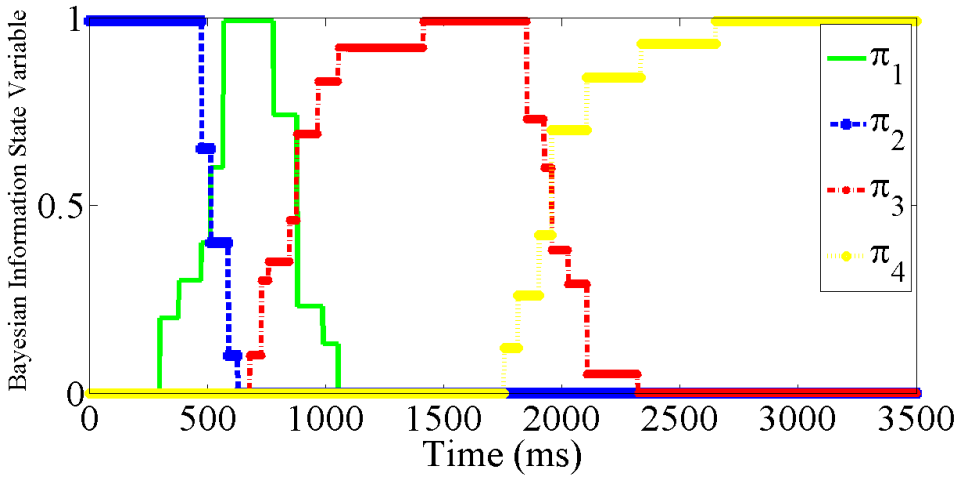


Fig. 6-5 Bayesian information state variable π_i (i=1, 2, 3, 4) transitions over time in an experimental trial. The horizontal line is the threshold 0.5. When π_i is over the threshold 0.5, the cognitive state transitions into the next one.

6.4 Discussion and Conclusions

Studies have shown that neuronal data encode rich information about kinematics [1-11, 28-31], and most decoding algorithms operate in a task-dependent fashion. These systems are trained on a limited number of movement tasks, and do not generalize well over movement tasks not included in the training datasets. Our proposed task-independent decoding framework generalizes the decoding problem, as demonstrated by the framework's performance on tasks not included in the training datasets. The number of movement tasks and the trajectories of the movement tasks do not need to be known a priori nor chosen a posteriori based on the modulation of some statistics. Because our proposed TI decoding framework is independent of movement tasks, it is scalable, which is necessary for clinically viable neural prosthetics.

Furthermore, our proposed TI decoding framework includes cognitive state transition information [35, 21] and uses cognitive-state-dependent parameters instead of movement-dependent parameters. The cognitive-state-dependent parameters characterize the kinematics in each cognitive state specifically with one set of model parameters for each cognitive state, and thus have better decoding performance than a kinematics decoding model with only one set of parameters for all cognitive states in a given movement task. When detecting the cognitive state transitions, we used point process models with history dependency to model the neuronal spiking activity and have one hidden state to represent one cognitive state, instead of choosing the number of hidden states for each cognitive state based on hysteresis. In other words, the cognitive state transition detection model is data-driven, rather than fitting data into a pre-fixed model.

These findings suggest that TI models with cognitive-state-dependent parameters may be used to more accurately decode kinematics data, and our proposed decoding system could pave the way for a clinical viable neural prosthetics.

6.5 Bibliography

[1] E. E. Fetz, "Real-time control of a robotic arm by neuronal ensemble," *Nature,* vol. 2, pp. 583-584, 1999.

[2] J. M. Carmena, M. A. Lebedev, R. E. Crist, E. O. Joseph, D. M. Santucci, D. F. Demitrov, P. G. Patil, C. S. Henriquez and M. A. L Nicolelis, "Learning to control a brain-computer interface for reaching and grasping by primates," *PLoS BioL,* vol. 1, pp. 193-208, 2003.

[3] J. P. Donoghue, "Connecting cortex to machines: recent advances in brain interface," *Nature Neurosci,* vol. 5, pp. 19085-1088, 2002.

[4] M. A. Nicolelis, "Actions from thoughts," *Nature,* vol. 409, pp. 403-407, 2001.

[5] G. Santhenam, S. I. Ryu, B. M. Yu, A. Afshar and K. V. Shenoy, “A high-performance brain-computer interface,” *Nature,* vol. 442, pp. 195-198, 2006.

[6] A. B. Schwartz, “Cortical neuronal prosthetics,” *Annual. Rev. Neurosci.,* vol. 27, pp. 487-505, 2004.

[7] S. H. Scott, “Neuroscience: converting thoughts into actions,” *Nature,* vol. 416, pp. 141-142, 2006.

[8] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows and J. Donoghue, “Instant neural control of a movement signal,” *Nature,* vol. 416, pp. 141-142, 2002.

[9] D. M. Taylor, S. I. H. Tillery and A. B. Schwarts, “Direct cortical control of 3D neuroprosthetic devices,” *Science,* vol. 296, pp. 1829-1832, 2002.

[10] S. Musallam, B. D. Corneil, H. Scherberger and R. A. Andersen, “Cognitive control signals for neural prosthetics,” *Science,* vol. 305, pp. 258-262, 2004.

[11] T. M. Michell, R. Huchinson, R. S. Niculescu, F. Pepeira and X. Wang, "Learning to Decode Cognitive States From Brain Images," *Mach Learn,* vol. 57, pp. 145-175, 2004.

[12] M. Mollazadeh, V. Aggrawal, A. G. Davidson, A. J. Law, N. V. Thakor and M. H. Schieber, “Spatiotemporal variation of multiple neurophysiological signals in the primary motor cortex during dexterous reach-to-grasp movements,” *J. Neurosci.,* vol. 31, pp. 15531-15543, 2011.

[13] C. Kemere, G. Santhanam, B. M. Yu, S. I. Ryu, T. H. Meng and K. V. Shenoy, “Model-based decoding of reaching movements for prosthetic systems,” *Proc IEEE Eng. Med. Biol. Soc.,* pp. 4524-4528, 2004.

[14] C. Kemere, K. V. Shenoy and T. H. Meng, “Model-based neural decoding of reaching movements: a maximum likelihood approach,” *IEEE Trans. Biomed. Eng.,* vol. 51, pp. 925-932, 2004.

[15] B. M. Yu, C. Kemere, G. Santhanam, A. Afshar, S. I. Ryu, T. H. Meng, M. Sahani and K. V. Shenoy, “Mixture of trajectory models for neural decoding of goal directed movements,” *J Neurophysiol,* vol. 97, pp. 3763-3780, 2007.

[16] G. Radons, J. D. Becker, B. Dulfer and J. Kruger, “Analysis, classification, and coding of multielectrode spike trains with hidden Markov models,” *Biol. Cybern.,* vol. 71, pp. 359-373, 1994.

[17] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue and E. N. Brown, “A point process framework for relating neural spiking acitivity to spiking history, neural ensemble, and extrinsic covariate effects,” *J. Neurophysiol.,* vol. 93, pp. 1074-1089, 2005.

[18] A.-C. Camproux, F. Saunier, G. Chouvet, J.-C. Thalabard and G. Thomas, “A hidden Markov approach to neuron firing patterns,” *Biophys. J.,* vol. 71, pp. 2404-2412, 1996.

[19] W. Wu, Y. Gao, E. Bienenstock, J. P. Donoghue and M. J. Black, “Bayesian population coding of motor cortical activity using a Kalman filter,” *Neural Comput.,* vol. 18, pp. 80-118, 2006.

[20] W. Wu, J. Kulkarni, N. Hatsopoulos and L. Paninski, “Neural decoding of goal-directed movements using a linear statespace model with hidden states,” *IEEE Trans. Biomed. Eng.,* vol. 17, pp. 1-9, 2009.

[21] A. J. Yu, “Optimal change-detection and spiking neurons,” *Proc. 20th Annu. Conf. Neural Inf. Process. Syst,* pp. 1545-1552, 2006.

[22] D. R. Cox and V. Isham, Point processes, CRC Press, 1980.

[23] D. L. Snyder and M. I. Miller, Random point processes in time and space, Springer-Verlag, 1991.

[24] J. Moeller and R. Waagepetersen, Statistical inference and simulation for spatial point processes, Chapman Hall, 2004.

[25] J. Moeller, A. Syversveen and R. Waagepetersen, “Log-Gaussian Cox processes,” *Scand. J. Stat.,* vol. 25, pp. 451-482, 1998.

[26] J. E. Kulkarni and L. Paninski, “Common-input models for multiple neural spike-train data,” *Network,* vol. 18, pp. 375-407, 2007.

[27] S. Santaniello, E. B. Montgomery, J. T. Gale and S. V. Sarma, "Non-stationary discharge patterns in motor cortex under subthalamic nucleus deep brain stimulation," *Front Integr Neurosci,* vol. 6, pp. 1-13, 2012.

[28] S. V. Sarma, M. L. Cheng, U. Eden, Z. Williams, E. N. Brown and E. Eskandar, "The effects of cues on neurons in the basal ganglia in Parkingson's disease," *Front Integr Neurosci,* vol. 6, pp. 1-12, 2012.

[29] J. Bloit and X. Rodet, "Short-term Viterbi for Online HMM Decoding: Evaluation on a Real-time Phone Recognition Task," in *ICASSP*, Las Vegas, NV, 2008.

[30] N. L. Johnson and S. Kotz, Distribution in Statistics-Continuous Univariate Distributions, NY: John Wiley & Sons, 1970.

[31] H. Akaike, "Factor analysis and AIC," *Psychometrika,* vol. 52, pp. 317-332, 1987.

[32] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decodign algorithm," *IEEE Trans. Inform. Theory,* vol. 13, pp. 260-269, 1967.

[33] G. D. Forney, "The Viterbi algorithm," *Proc of IEEE,* vol. 61, pp. 268-278, 1973.

[34] S. Santaniello, D. L. Sherman, N. V. Thakor, E. N. Eskandar and S. V. Sarma, "Optimal Control-Based Bayesian Detection of Clinical and Behavioral State Transitions," *IEEE Trans. Neural Syst. Rehab. Eng.,* vol. 20, pp. 708-719, 2012.

[35] S. Faul, G. Boylan, S. Connolly, L. Marnane and G. Lightbody, "An evaluation of neonatal seizure detection methods," *J. Clin. Neurophysiol.,* vol. 116, pp. 1533-`541, 2005.

[36] V. Aggarwal, S. Acharya, F. Tenore, S. Hyun-chool, R. Etienne-Cummings, M. H. Schieber and N. V. Thakor, "Asynchronous decoding of dexterous finger using M1 neurons," *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 16, pp. 3-14, 2008.

[37] S. Acharya, F. Tenore, V. Aggarwal, R. Etienne-Cummings, M. H. Schieber and N. V. Thakor, "Decoding individual finger movements using volume constrained neuronal ensembles in the M1 hand area," *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 16, pp. 15-23, 2008.

[38] L. E. Baum and J. A. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bull. Amer. Math. Soc.,* vol. 3, pp. 360-363, 1967.

[39] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Stat.,* vol. 6, pp. 1554-1563, 1966.

[40] L. E. Baum, T. Petrie, G. Soules and N. Weiss, "A maximization technique occuring in the statistical analysis of probabilitic functions of Markov chains," *Ann. Math. Stat.,* vol. 1, pp. 164-171, 1970.

[41] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue and E. N. Brown, "A point process framework fo relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects," *J. Neurophysiol.,* vol. 93, pp. 1074-1089, 2005.

[42] J. Nelder and R. Wedderburn, "Generalized Linear Models," *Journal of the Royal Statistical Society,* vol. 135, no. 3, p. 370–384, 1972.

[43] A. Dempster, N. Laird and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J.R.Stat.Soc.,* vol. 39, pp. 1-38, 1977.

[44] A. C. Smith and E. N. Brown, "Estimating a state-space model from point process observations," *Neural Computation,* vol. 15, pp. 965-993, 2003.

[45] A. C. Smith, L. M. Frank, S. Wirth, M. Yanike, D. Hu, Y. Kubota, A. M. Graybiel, W. Suzuki and E. N. Brown, "Dynamic analysis of learning in behavioral experiments," *J. Neurosci. ,* vol. 24, pp. 447-461, 2004.

[46] M. Basseville and I. V. Nikiforov, Detection of abrupt changes: theory and changes, NJ: Prentice Hall, 1993.

[47] U. Eden, L. Frank, R. Barbieri, V. Solo and E. Brown, "Dynamic Analysis of Neural Encoding by Point Process Adaptive Filtering," *Neural Computation,* vol. 16, pp. 971-998, 2004.

[48] M. Basseville and I. V. Nikiforov, Detection of Abrupt Changes: Theory and Changes, NJ: Prentice Hall, 1993.

[49] J. W. Pillow, Y. Ahmadian and L. Paninski, "Model-based decoding, information estimation, and change-point detection techniques for multineuron spike trains," *Neural Comput.,* vol. 23, pp. 1-45, 2011.

[50] S. V. Sarma, U. T. Eden, M. L. Cheng, Z. M. Williams, R. Hu, E. Eskansar and E. N. Brown, "Using point process models to compare neural spiking activity in the subthalamic nucleus of Parkinson's patients and a healthy primate.," *IEEE Trans Biomed Eng,* vol. 57, pp. 1297-1305, 2010.

[51] L. E. Baum and T. Petrie, “Statistical inference for probabilistic functions of finite state Markov chains,” *Ann. Math. Stat.,* vol. 6, pp. 1554-1563, 1966.

[52] L. E. Baum and J. A. Eagon, “An ineqality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology,” *Bull. Amer. Math. Soc.,* vol. 3, pp. 360-363, 1967.

[53] L. E. Baum, T. Petrie, G. Soules and N. Weiss, “A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains,” *Ann. Math. Stat.,* vol. 1, pp. 164-171, 1970.

[54] J. Ushiba, Y. Tomita, Y. Masakado and Y. Komune, "A cumulative sum test for a peri-stimulus time histogram using the Monte Carlo method," *J. Neurosci. Meth.,* vol. 118, pp. 207-214, 2002.

[55] J. Zhuang, W. Truccolo, C. Vargas-Irwin and J. P. Donoghue, “Decoding 3D reach and grasp kinematics from high-frequency local field potentials in primate primary motor cortex,” *IEEE Trans Biomed Eng.,* vol. 57, no. 7, pp. 1774-1784, 2010.

[56] C. Mehring, J. Rickert, E. Vaadia, S. Oliveira, A. Aertsen and S. Rotter, “Inference of hand movements from local field potentials in monkey motor cortex,” *Nat Neurosci,* vol. 6, no. 12, pp. 1253-1254, 2003.

[57] G. Schalk, J. Kubanek, K. J. Miller, N. R. Anderson, E. C. Leuthardt, J. G. Ojemann, D. Limbrick, D. Moran, L. A. Gerhardt and J. R. Wolpaw, “Decoding two-dimensional movement trajectories using electrocorticographic signals in humans,” *J. Neural Eng.,* vol. 4, pp. 264-275, 2007.

[58] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows and J. P. Donoghue, “Brain-machine interface: instant neural control of a movement signal,” *Nature,* vol. 416, no. 6877, pp. 141-142, 2002.

[59] J. Wessberg, C. R. Stambaugh, J. D. Kralik, P. D. Beck, M. Laubach, J. K. Chapin, J. Kim, S. J. Biggs, M. A. Srinivasan and M. A. Nicolelis, “Real-time prediction of hand trajectory by ensembles of cortical neurons in primates,” *Nature,* vol. 408, no. 6810, pp. 361-365, 2000.

[60] E. Stark and M. Abeles, “Predicting movement from multiunit activity,” *J Neurosci,* vol. 27, no. 31, pp. 8387-8394, 2007.

[61] J. R. Wolpaw and D. J. McFarland, “Control of of a two-dimensional movement signal by a noninvasive brain-computer interface in humans,” *Proc. Natl Acad. Sci.,* vol. 101, pp. 17849-17854, 2004.

[62] A. P. Georgopoulos, A. B. Schwartz and R. E. Kettner, “Neuronal population coding of movement direction,” *Science,* vol. 233, pp. 1416-1419, 1986.

[63] D. M. Taylor, S. I. Tillery and A. B. Schwartz, “Direct cortical control of 3D neuroprosthetic devices,” *Science,* vol. 296, pp. 1829-1832, 2002.

[64] M. A. Lebedev, J. M. Carmena, J. E. O'Doherty, M. Zacksenhouse, C. S. Henriquez, J. C. Principe and M. A. Nicolelis, “Cortical ensemble adaptation to represent velocity of an artificial actuator controlled by a brain-machine interface,” *J. Neurosci.,* vol. 25, pp. 4681-4693, 2005.

[65] J. P. Donoghue, A. Nurmikko, G. Friehs and M. Black, “Development of neuromotor prostheses for humans,” *Suppl. Clin. Neurophysiol.,* vol. 57, pp. 592-606, 2004.

[66] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. A. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn and J. P. Donoghue, “Neuronal ensemble control of prosthetic devices by a human with tetraplegia,” *Nature,* vol. 442, pp. 164-171, 2006.

[67] W. Wu, Y. Gao, E. Bienenstock, J. Donoghue and M. Black, "Bayesian Population Decoding of Motor Cortical Activity Using a Kalman Filter," *Neural Computation,* vol. 18, pp. 80-118, 2006.

[68] X. Kang, S. Sarma, S. Santaniello, M. Schiber and N. V. Thakor, "Task independent cognitive state transition detection from cortical neurons," *IEEE Trans. on Neural Syst. and Rehab. Engi.,* 2014.

Chapter 7 Conclusions and Summary

7.1 Conclusion

7.2 Limitations and Future Work

7.1 Conclusion

This dissertation explored the relationship of cognitive states, kinematics, and 3D reach-to-grasp tasks to enable clinically practical decoding of kinematics in complex 3D reach-to-grasp tasks with cognitive states. This dissertation also demonstrated the methods of online decoding cognitive states and their applications in clinical practice. The approach involved designing a novel system framework using hidden Markov models and point process models (cognitive state transition detector) to detect cognitive state transitions in real time from cortical / neuronal spike trains. It further involved designing a novel system framework using state models in coupled with the cognitive state transition detector to decode kinematics in real time from cortical / neuronal spike trains. This dissertation demonstrated that (1) the proposed task-independent framework for cognitive state transition detection has equivalent, or better, performance than task-dependent framework; (2) the proposed task-independent framework for kinematics decoding has equivalent, or better, performance than task-dependent framework; (3) the incorporation of cognitive state transition information can improve the decoding performance of kinematics decoding models.

7.2 Future Work and Limitations

The task-independent framework for detecting cognitive state transitions and decoding kinematics presented in the dissertation generalize well across 3D reach-to-grasp movements. These concepts not only apply to the four movement targets within the current 3D reach-to-grasp movement set-up, but also to other reach-to-grasp tasks to include N targets.

Although we only tested the framework in reach-to-grasp movements, the framework may be able to perform well for other experiments and movement patterns. Future work may study further the application of this proposed framework for other movement task decoding.

Furthermore, current work in this dissertation only studied the usage of neuronal spike trains as neural inputs to detect cognitive state transitions. Future work may explore further the possibilities and performance of other neuronal signals, such as local field potentials, EEG, ECog etc.

Last but not least, the proposed work can also further be applied in other medical research, such as epilepsy patients. Future work can explore using our framework for other medical applidcations.

7.3 Reference

[1] M. Hallbach, U. Egert, J. Hescheler and K. Banach, "Estimation of action potential changes from field potential recordings in multicellular mouse cardiac myocyte cultures," *Cell Physiol Biochem,* vol. 13, no. 5, pp. 271-284, 2003.

[2] S. Romani and M. Tsodyks, "Short-term plasticity based network model of place cells dynamics," *Hippocampus,* 2014.

[3] C. Salvatico, C. Specht and A. Triller, "Synaptic receptor dynamics: from theorectical concepts to deep quantification and chemistry in cellulo," *Neuropharmacology,* 2014.

[4] T. Ball, M. Kern, I. Mutschler, A. Aertsen and A. Schulze-Bonhage, "Signal quality of simultaneously recorded invasive and non-invasive EEG," *NeuroImage,* vol. 46, pp. 708-716, 2009.

[5] G. Buzsaki, C. A. Anastassiou and C. Koch, "The origin of extracellular fields and currents - EEG, ECog, LFP and spikes," *Nature Review,* vol. 13, pp. 407-420, 2012.

[6] M. A. Tansey, "EEG sensorimotor rhythm biofeedback training: some effects on the neurologic precursors of learning disabilities," *International Journal of Psychophysiology,* pp. 163-177, 1984.

[7] S. H. Patel and P. N. Azzam, "Characterization of N200 and P300: selected studies of the event-related potential," *Int J Med Sci,* vol. 2, pp. 147-154, 2005.

[8] N. K. Logothetis, "What we can do and what we cannot do with fMRI," *Nature,* vol. 453, pp. 869-878, 2008.

[9] M. N. Kim, T. Durduran, S. Frangos, B. L. Edlow, E. M. Buckley, H. E. Moss, C. Zhou, G. Yu, R. Choe, E. Maloney-Wileysky, R. L. Wolf, M. S. Grady, J. H. Greenberg, J. M. Levine, A. G. Yodh, J. A. Detre and W. A. Kofke, "Noninvasive measurement of cerebral blood flow and blood oxygenation using near-infrared and diffuse correlation spectroscopies in critically brain-injuried adults," *Neurocrit Care,* 2011.

[10] W. G. Walter, "Contigent negative variation: an electrical sign of sensorimotor association and expectancy in the human brain," *Nature,* vol. 203, pp. 380-384, 1964.

[11] E. Fetz, "Operant conditioning of cortical unit activity," *Science,* vol. 163, pp. 955-958, 1969.

[12] J. Donoghue, "Bridging the brain to the world: a perspective on neural interface systems," *Neuron ,* pp. 511-521, 2008.

[13] J. M. Carmena, M. A. Lebedev, R. E. Crist, E. O. Joseph, D. M. Santucci, D. F. Demitrov, P. G. Patil, C. S. Henriquez and M. A. L Nicolelis, "Learning to control a brain-computer interface for reaching and grasping by primates," *PLoS BioL,* vol. 1, pp. 193-208, 2003.

[14] A. B. Schwartz, "Cortical neuronal prosthetics," *Annual. Rev. Neurosci.,* vol. 27, pp. 487-505, 2004.

[15] A. Riehle and E. Vaadia, Motor cortex in voluntary movements, CRC press, 2005.

[16] S. H. Scott, "Population vectors and motor cortex: neural coding or epiphenonmenon?," *Nature Neurosicence,* vol. 3, pp. 307-308, 2000.

[17] R. E. Chapin, K. T. Morgan and J. S. Bus, "The morphogenesis of testilar degeneration induced in rats by orally administered 2,5-hexanedione," *Experimental and Molecular Pathology,* vol. 38, no. 2, pp. 149-169, 1983.

[18] R. L. Doty, "Studies of olfactory dysfunction in major neurological disorders," *Advanced in the Biosciences,* vol. 93, pp. 593-602, 1994.

[19] N. Birbaumer, "Breaking the silence: brain-computer interfaces for communication and motor control," *Psychophysiology,* vol. 43, pp. 517-532, 2006.

[20] E. C. Leuthardt, G. Schalk, J. R. Wolpaw, J. G. Ojemann and D. W. Moran, "A brain-computer interface using electrocorticographic signals in humans," *J. Neural Eng.,* vol. 1, pp. 63-71, 2004.

[21] J. P. Donoghue, A. Nurmikko, G. Friehs and M. Black, "Development of neuromotor prostheses for humans," *Suppl. Clin. Neurophysiol.,* vol. 57, pp. 592-606, 2004.

[22] M. A. Nicolelis, "Actions from thoughts," *Nature,* vol. 409, pp. 403-407, 2001.

[23] S. Musallam, B. D. Corneil, H. Scherberger and R. A. Andersen, “Cognitive control signals for neural prosthetics,” *Science,* vol. 305, pp. 258-262, 2004.

[24] D. M. Taylor, S. I. Tillery and A. B. Schwartz, “Direct cortical control of 3D neuroprosthetic devices,” *Science,* vol. 296, pp. 1829-1832, 2002.

[25] R. J. Seitz, T. A. Matyas and L. M. Carey, "Neural plasticity as a basis for motor learning and neurorehabilitation," *Brain Impairment,* vol. 9, no. 2, pp. 103-113, 2008.

[26] E. E. Fetz, “Real-time control of a robotic arm by neuronal ensemble,” *Nature,* vol. 2, pp. 583-584, 1999.

[27] J. P. Donoghue, “Connecting cortex to machines: recent advances in brain interface,” *Nature Neurosci,* vol. 5, pp. 19085-1088, 2002.

[28] G. Santhenam, S. I. Ryu, B. M. Yu, A. Afshar and K. V. Shenoy, “A high-performance brain-computer interface,” *Nature,* vol. 442, pp. 195-198, 2006.

[29] S. H. Scott, “Neuroscience: converting thoughts into actions,” *Nature,* vol. 416, pp. 141-142, 2006.

[30] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows and J. Donoghue, “Instant neural control of a movement signal,” *Nature,* vol. 416, pp. 141-142, 2002.

[31] D. M. Taylor, S. I. H. Tillery and A. B. Schwarts, “Direct cortical control of 3D neuroprosthetic devices,” *Science,* vol. 296, pp. 1829-1832, 2002.

[32] T. M. Michell, R. Huchinson, R. S. Niculescu, F. Pepeira and X. Wang, "Learning to Decode Cognitive States From Brain Images," *Mach Learn,* vol. 57, pp. 145-175, 2004.

[33] M. Mollazadeh, V. Aggrawal, A. G. Davidson, A. J. Law, N. V. Thakor and M. H. Schieber, “Spatiotemporal variation of multiple neurophysiological signals in the primary motor cortex during dexterous reach-to-grasp movements,” *J. Neurosci.,* vol. 31, pp. 15531-15543, 2011.

[34] C. Kemere, G. Santhanam, B. M. Yu, S. I. Ryu, T. H. Meng and K. V. Shenoy, “Model-based decoding of reaching movements for prosthetic systems,” *Proc IEEE Eng. Med. Biol. Soc.,* pp. 4524-4528, 2004.

[35] C. Kemere, K. V. Shenoy and T. H. Meng, “Model-based neural decoding of reaching movements: a maximum likelihood approach,” *IEEE Trans. Biomed. Eng.,* vol. 51, pp. 925-932, 2004.

[36] B. M. Yu, C. Kemere, G. Santhanam, A. Afshar, S. I. Ryu, T. H. Meng, M. Sahani and K. V. Shenoy, “Mixture of trajectory models for neural decoding of goal directed movements,” *J Neurophysiol,* vol. 97, pp. 3763-3780, 2007.

[37] G. Radons, J. D. Becker, B. Dulfer and J. Kruger, “Analysis, classification, and coding of multielectrode spike trains with hidden Markov models,” *Biol. Cybern.,* vol. 71, pp. 359-373, 1994.

[38] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue and E. N. Brown, “A point process framework for relating neural spiking acitivity to spiking history, neural ensemble, and extrinsic covariate effects,” *J. Neurophysiol.,* vol. 93, pp. 1074-1089, 2005.

[39] A.-C. Camproux, F. Saunier, G. Chouvet, J.-C. Thalabard and G. Thomas, “A hidden Markov approach to neuron firing patterns,” *Biophys. J.,* vol. 71, pp. 2404-2412, 1996.

[40] W. Wu, Y. Gao, E. Bienenstock, J. P. Donoghue and M. J. Black, “Bayesian population coding of motor cortical activity using a Kalman filter,” *Neural Comput.,* vol. 18, pp. 80-118, 2006.

[41] W. Wu, J. Kulkarni, N. Hatsopoulos and L. Paninski, “Neural decoding of goal-directed movements using a linear statespace model with hidden states,” *IEEE Trans. Biomed. Eng.,* vol. 17, pp. 1-9, 2009.

[42] A. J. Yu, “Optimal change-detection and spiking neurons,” *Proc. 20th Annu. Conf. Neural Inf. Process. Syst,* pp. 1545-1552, 2006.

[43] D. R. Cox and V. Isham, Point processes, CRC Press, 1980.

[44] D. L. Snyder and M. I. Miller, Random point processes in time and space, Springer-Verlag, 1991.

[45] J. Moeller and R. Waagepetersen, Statistical inference and simulation for spatial point processes, Chapman Hall, 2004.

[46] J. Moeller, A. Syversveen and R. Waagepetersen, “Log-Gaussian Cox processes,” *Scand. J. Stat.,* vol. 25, pp. 451-482, 1998.

[47] J. E. Kulkarni and L. Paninski, “Common-input models for multiple neural spike-train data,” *Network,* vol. 18, pp. 375-407, 2007.

[48] S. Santaniello, E. B. Montgomery, J. T. Gale and S. V. Sarma, "Non-stationary discharge patterns in motor cortex under subthalamic nucleus deep brain stimulation," *Front Integr Neurosci,* vol. 6, pp. 1-13, 2012.

[49] S. V. Sarma, M. L. Cheng, U. Eden, Z. Williams, E. N. Brown and E. Eskandar, "The effects of cues on neurons in the basal ganglia in Parkingson's disease," *Front Integr Neurosci,* vol. 6, pp. 1-12, 2012.

[50] J. Bloit and X. Rodet, "Short-term Viterbi for Online HMM Decoding: Evaluation on a Real-time Phone Recognition Task," in *ICASSP*, Las Vegas, NV, 2008.

[51] N. L. Johnson and S. Kotz, Distribution in Statistics-Continuous Univariate Distributions, NY: John Wiley & Sons, 1970.

[52] H. Akaike, “Factor analysis and AIC,” *Psychometrika,* vol. 52, pp. 317-332, 1987.

[53] A. J. Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decodign algorithm,” *IEEE Trans. Inform. Theory,* vol. 13, pp. 260-269, 1967.

[54] G. D. Forney, “The Viterbi algorithm,” *Proc of IEEE,* vol. 61, pp. 268-278, 1973.

[55] S. Santaniello, D. L. Sherman, N. V. Thakor, E. N. Eskandar and S. V. Sarma, "Optimal Control-Based Bayesian Detection of Clinical and Behavioral State Transitions," *IEEE Trans. Neural Syst. Rehab. Eng.,* vol. 20, pp. 708-719, 2012.

[56] S. Faul, G. Boylan, S. Connolly, L. Marnane and G. Lightbody, “An evaluation of neonatal seizure detection methods,” *J. Clin. Neurophysiol.,* vol. 116, pp. 1533-`541, 2005.

[57] V. Aggarwal, S. Acharya, F. Tenore, S. Hyun-chool, R. Etienne-Cummings, M. H. Schieber and N. V. Thakor, “Asynchronous decoding of dexterous finger using M1 neurons,” *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 16, pp. 3-14, 2008.

[58] S. Acharya, F. Tenore, V. Aggarwal, R. Etienne-Cummings, M. H. Schieber and N. V. Thakor, “Decoding individual finger movements using volume constrained neuronal ensembles in the M1 hand area,” *IEEE Trans. Neural Syst. Rehabil. Eng.,* vol. 16, pp. 15-23, 2008.

[59] L. E. Baum and J. A. Eagon, “An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology,” *Bull. Amer. Math. Soc.,* vol. 3, pp. 360-363, 1967.

[60] L. E. Baum and T. Petrie, “Statistical inference for probabilistic functions of finite state Markov chains,” *Ann. Math. Stat.,* vol. 6, pp. 1554-1563, 1966.

[61] L. E. Baum, T. Petrie, G. Soules and N. Weiss, “A maximization technique occuring in the statistical analysis of probabilitic functions of Markov chains,” *Ann. Math. Stat.,* vol. 1, pp. 164-171, 1970.

[62] W. Truccolo, U. T. Eden, M. R. Fellows, J. P. Donoghue and E. N. Brown, “A point process framework fo relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects,” *J. Neurophysiol.,* vol. 93, pp. 1074-1089, 2005.

[63] J. Nelder and R. Wedderburn, "Generalized Linear Models," *Journal of the Royal Statistical Society,* vol. 135, no. 3, p. 370–384, 1972.

[64] A. Dempster, N. Laird and D. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *J.R.Stat.Soc.,* vol. 39, pp. 1-38, 1977.

[65] A. C. Smith and E. N. Brown, “Estimating a state-space model from point process observations,” *Neural Computation,* vol. 15, pp. 965-993, 2003.

[66] A. C. Smith, L. M. Frank, S. Wirth, M. Yanike, D. Hu, Y. Kubota, A. M. Graybiel, W. Suzuki and E. N. Brown, “Dynamic analysis of learning in behavioral experiments,” *J. Neurosci. ,* vol. 24, pp. 447-461, 2004.

[67] M. Basseville and I. V. Nikiforov, Detection of abrupt changes: theory and changes, NJ: Prentice Hall, 1993.

[68] U. Eden, L. Frank, R. Barbieri, V. Solo and E. Brown, "Dynamic Analysis of Neural Encoding by Point Process Adaptive Filtering," *Neural Computation,* vol. 16, pp. 971-998, 2004.

[69] M. Basseville and I. V. Nikiforov, Detection of Abrupt Changes: Theory and Changes, NJ: Prentice Hall, 1993.

[70] J. W. Pillow, Y. Ahmadian and L. Paninski, "Model-based decoding, information estimation, and change-point detection techniques for multineuron spike trains," *Neural Comput.,* vol. 23, pp. 1-45, 2011.

[71] S. V. Sarma, U. T. Eden, M. L. Cheng, Z. M. Williams, R. Hu, E. Eskansar and E. N. Brown, "Using point process models to compare neural spiking activity in the subthalamic nucleus of Parkinson's patients and a healthy primate.," *IEEE Trans Biomed Eng,* vol. 57, pp. 1297-1305, 2010.

[72] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Stat.,* vol. 6, pp. 1554-1563, 1966.

[73] L. E. Baum and J. A. Eagon, "An ineqality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bull. Amer. Math. Soc.,* vol. 3, pp. 360-363, 1967.

[74] L. E. Baum, T. Petrie, G. Soules and N. Weiss, "A maximization technique occuring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Stat.,* vol. 1, pp. 164-171, 1970.

[75] J. Ushiba, Y. Tomita, Y. Masakado and Y. Komune, "A cumulative sum test for a peri-stimulus time histogram using the Monte Carlo method," *J. Neurosci. Meth.,* vol. 118, pp. 207-214, 2002.

[76] J. Zhuang, W. Truccolo, C. Vargas-Irwin and J. P. Donoghue, "Decoding 3D reach and grasp kinematics from high-frequency local field potentials in primate primary motor cortex," *IEEE Trans Biomed Eng.,* vol. 57, no. 7, pp. 1774-1784, 2010.

[77] C. Mehring, J. Rickert, E. Vaadia, S. Oliveira, A. Aertsen and S. Rotter, "Inference of hand movements from local field potentials in monkey motor cortex," *Nat Neurosci,* vol. 6, no. 12, pp. 1253-1254, 2003.

[78] G. Schalk, J. Kubanek, K. J. Miller, N. R. Anderson, E. C. Leuthardt, J. G. Ojemann, D. Limbrick, D. Moran, L. A. Gerhardt and J. R. Wolpaw, "Decoding two-dimensional movement trajectories using electrocorticographic signals in humans," *J. Neural Eng.,* vol. 4, pp. 264-275, 2007.

[79] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows and J. P. Donoghue, "Brain-machine interface: instant neural control of a movement signal," *Nature,* vol. 416, no. 6877, pp. 141-142, 2002.

[80] J. Wessberg, C. R. Stambaugh, J. D. Kralik, P. D. Beck, M. Laubach, J. K. Chapin, J. Kim, S. J. Biggs, M. A. Srinivasan and M. A. Nicolelis, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature,* vol. 408, no. 6810, pp. 361-365, 2000.

[81] E. Stark and M. Abeles, "Predicting movement from multiunit activity," *J Neurosci,* vol. 27, no. 31, pp. 8387-8394, 2007.

[82] J. R. Wolpaw and D. J. McFarland, "Control of of a two-dimensional movement signal by a noninvasive brain-computer interface in humans," *Proc. Natl Acad. Sci.,* vol. 101, pp. 17849-17854, 2004.

[83] A. P. Georgopoulos, A. B. Schwartz and R. E. Kettner, "Neuronal population coding of movement direction," *Science,* vol. 233, pp. 1416-1419, 1986.

[84] M. A. Lebedev, J. M. Carmena, J. E. O'Doherty, M. Zacksenhouse, C. S. Henriquez, J. C. Principe and M. A. Nicolelis, "Cortical ensemble adaptation to represent velocity of an artificial actuator controlled by a brain-machine interface," *J. Neurosci.,* vol. 25, pp. 4681-4693, 2005.

[85] L. R. Hochberg, M. D. Serruya, G. M. Friehs, J. A. Mukand, M. Saleh, A. H. Caplan, A. Branner, D. Chen, R. D. Penn and J. P. Donoghue, "Neuronal ensemble control of prosthetic devices by a human with tetraplegia," *Nature,* vol. 442, pp. 164-171, 2006.

[86] W. Wu, Y. Gao, E. Bienenstock, J. Donoghue and M. Black, "Bayesian Population Decoding of Motor Cortical Activity Using a Kalman Filter," *Neural Computation,* vol. 18, pp. 80-118, 2006.

[87] X. Kang, S. Sarma, S. Santaniello, M. Schiber and N. V. Thakor, "Task independent cognitive state transition detection from cortical neurons," *IEEE Trans. on Neural Syst. and Rehab. Engi.,* 2014.

CV for Ph.D. Candidate

The Johns Hopkins University School of Medicine

Xiaoxu Kang 09/28/2014

Educational History

Ph.D.	2014	Program in Biomedical Engineering	Johns Hopkins University
		Mentors: Dr. Nitish Thakor and Dr. Sridevi Sarma	
M.S.E.	2009	Program in Biomedical Engineering	Johns Hopkins University
B.S.	2007	Program in Biomedical Engineering	Beihang University

EMPLOYMENT

Johns Hopkins University, Baltimore, MD. Jul, 2009-Sep, 2014

National Institute of Health (NIH) NeuroEngineering Training Program

Advisor: Nitish V. Thakor (Biomedical Engineering & Electrical Engineering)

- Doctoral research part of DARPA Revolutionizing Prosthetics program to develop a neurally controlled upper-limb prosthetic device for use by amputees and paralyzed humans.
- Developed advanced signal processing algorithms for Brain-Machine Interfaces (BMI) that decode information from the brain for dexterous control of a prosthetic upper limb.
- Introduced a novel state-based kinematics prediction framework using switching linear dynamical systems to control a virtual prosthetic arm in real time using neural recordings in premotor cortex of primates.
- Presented a novel approach based on hidden Markov models for detecting neural state transitions during reach-to-grasp tasks performed by primates.

Johns Hopkins University, Baltimore, MD. Sep, 2007-2009

Department of Biomedical Engineering

Advisor: Nitish V. Thakor (Biomedical Engineering & Electrical Engineering)

- Master research part of $3M National Institute of Health (NIH) Grant to develop methods to improve post-cardiac-arrest neurological recovery;
- Designed several complete experimental protocols approved by Johns Hopkins School of Medicine;
- Developed a novel quantitative index which predicts the severity of neurological injuries in post-cardiac-arrest patients;
- Developed advanced signal processing algorithms to systematically study the effects of temperature modulation on the neurological injuries of post-cardiac-arrest patients;
- Designed a closed-loop temperature modulation device for small animal experiments;
- Optimized the hypothermia temperature and intervention timings to maximize the neurological recovery of post-cardiac-arrest patients.

Fellowships, Awards and Honors

- Diversity Recognition Award *presented by President Ronald Daniels* for outstanding accomplishments fostering greater appreciation, advancement and celebration of diversity in the Johns Hopkins culture, March, 16th, 2012
- Women in Bio Robbie Melton Scholarship ($1000) for outstanding entrepreneurship and leadership in life science, 12/03, 2011
- Student Representative of Johns Hopkins University for Emerging Women Leaders Summits, 2012
- Student Representative of Johns Hopkins University for Emerging Women Leaders Summits, 2011
- Leadership Award presented by Center of Leadership Program Director, Johns Hopkins University, 2010
- Nominee for Howard Hughes Medical Institute International (HHMI) Student Research Fellowship (**6 in total**), Johns Hopkins University, 2011
- Faculty for the Future Fellowship ($30,000) (**33 in total worldwide**), Schlumberger Foundation, 2010
- Chiang Chen Scholarship ($50,000) as best engineering students overseas, China, 2007 (**7 in total nationally**)
- The Academic Star in the School of Electrical Engineering, Beihang University, 2006 (**5 in 3000**)
- The Second President Scholarship in School of Electrical Engineering, Beihang University, 2006 (**3 in 3000**)
- Excellence in Neural Engineering Travel Award ($800) in the 4th International IEEE EMBS on Neural Engineering Meeting, 2009
- JHU Creative Use of Technology Grant ($1500), Nov, 2012
- Macklin Mini-Fellowship (16 in total from Johns Hopkins University), March 3rd, 2012
- National Institute of Health (NIH) Neuroengineering Training Fellowship, 2011.9-2012.5
- Johns Hopkins Technology Fellowship Grant ($5000), Center for Educational Resources, 2011
- Advanced Computer Integrated Surgery Project Award, NSF Engineering Research Center, 2009
- The Whiting School of Engineering Graduate Student Fellowship, JHU, 2007-2009
- The Outstanding Undergraduate Scholarship, Beihang University, 2007
- The Third Prize in National College Student Electronics Design Competition, China, 2005
- The Second Prize in Feng Ru Academic and Technology Contest, Beihang University, 2005
- The Top Prize of the National College-Student English Contest, China, 2005
- The Second Prize for Excellent Students, Beihang University, 2004-2005
- The Top Prize of Academic Contest, Beihang University, 2004-2005
- The Third Prize of the Scientific Contest, Beihang University, 2004-2005
- The Second Prize of Academic Contest, Beihang University, 2003-2004
- The Top Prize for Excellent Students, Beihang University, 2003-2004
- The Second Prize in the National College-Student English Contest, China, 2004
- The Third Prize in the Fresh Students Topic Lecture Contest, Beihang University, 2003
- The Top Prize of Excellent Freshman Scholarship, Beihang University, 2003

PUBLICATIONS

JOURNALS

- **Xiaoxu Kang**, Sabato Santaniello, Sridevi V. Sarma, Marc H. Schieber, Nitish V. Thakor. "Task independent model for cognitive state transition detection in 3D reach-to-grasp movements". IEEE Trans on Neural Systems and Rehabilitation Engineering. (in review) 2014
- **Xiaoxu Kang**, Sabato Santaniello, Sridevi V. Sarma, Marc H. Schieber, Nitish V. Thakor. "Novel Framework of Online, Task-Independent kinematics decoding in 3D reach-to-grasp movements". IEEE Trans on Neural Systems and Rehabilitation Engineering. (in review) 2014

- **Xiaoxu Kang**, Anil Maybhate, Xiaofeng Jia, Romergryko G. Geocadin, Nitish V. Thakor, et al. "Multiscale entropy analysis of EEG for assessment of post-cardiac-arrest neurological recovery under hypothermia in rats". IEEE Transactions on Biomedical Engineering. 2009, 56 (4): 1023-1031.
- Matthew A. Koenig, Xiaofeng Jia, **Xiaoxu Kang**, Romergryko G. Geocadin, Nitish V. Thakor. "Intraventricular Orexin-A Improves Early Arousal and EEG Entropy in Rats after Cardiac Arrest". Brain Res. 2009 Feb 19; 1255:153-61.

THESES

- **Xiaoxu Kang**. Master Thesis: "Quantitative Analysis of EEG for Assessment of Long-Term Post-Cardiac-Arrest Neurological Outcomes under Temperature Modulation". 376 pages. Department of Biomedical Engineering, Johns Hopkins University. (2009)
- **Xiaoxu Kang**. Undergraduate Thesis: "A Novel Speech Processing Algorithm for Cochlear Implant based on wavelet transforms". Department of Biomedical Engineering, Advanced Engineering School, Beihang University, China. (2007)

CONFERENCES & ABSTRACTS

- **Xiaoxu Kang**, Marc H. Schieber, Nitish V. Thakor. " Decoding Cognitive States From Neural Activities of Somatosensory Cortex". 19th International Conference on Neural Information Processing (ICONIP). 2012
- **Xiaoxu Kang**, Marc H. Schieber, Nitish V. Thakor. "Switching Linear Dynamical System Decoding of Finger, Hand and Arm Kinematics Using Pre-Motor Cortical Ensembles". 34th Annual International Conference of the IEEE Engineering in Medicine and Biology. 2012
- **Xiaoxu Kang**, Nitish Thakor. "Factor Analyzed Hidden Markov Models for Estimating Prosthetic Limb Motions Using Premotor Cortical Ensembles". The Fourth IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob) June 24-27, 2012, Roma, Italy.
- **Xiaoxu Kang**, Sridevi V. Sarma, Marc H. Schieber, Nitish V. Thakor. "Hidden Markov Model Based State Decoding in Primate Reach-to-Grasp Tasks Using Premotor Cortical Ensembles". 16th International Conference on Cognitive and Neural Systems. May 30th-June 1st, 2012, Boston, USA.
- **Xiaoxu Kang**, Sanjeev Khudanpur."IVisual – Multi-modal surgical data visualization and synchronization tool for robotic minimally invasive surgery". Augmented Reality for Computer Assisted Surgery workshop. Medical Imaging Computing and Computer Assisted Intervention Conference, Toronto, Canada, Sep 18th-22nd, 2011.
- Wei Xiong, Matthew Koenig, **Xiaoxu Kang**, Xiaofeng Jia, Adrian Puttgen, Nitish Thakor, Romeryko Geocadin. "Evolution of Somatosensory Evoked Potentials in Post Cardiac Arrest Rats". Neurocritical Care Society Annual Meeting. 11:S1–S224. (2009) (Poster 34)
- **Xiaoxu Kang**, Xiaofeng Jia, Matthew Koenig, Romergryko Geocadin, Nitish V. Thakor. "Long-Term Assessment of Post-Cardiac-Arrest Neurological Outcomes with Somatosensory Evoked Potential in Rats". The 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2196 - 2199. Sep 3rd-6th. (2009)
- **Xiaoxu Kang**, Koenig, M., Orukari, I., Geocadin, R.G., Nitish V. Thakor. "Assessment of Post-Cardiac-Arrest Somatosensory Evoked Potential in Rats". The 4th International IEEE EMBS Conference on Neural Engineering, 609 - 613. April 29th-May 2nd. (2009)
- **Xiaoxu Kang**, Anil Maybhate, Xiaofeng Jia, Romergryko G. Geocadin, Nitish V. Thakor. "EEG signal analysis during post-cardiac arrest recovery under therapeutic hypothermia". The Society for Neuroscience Annual Meeting, Nov 15-19th. (2008)
- **Xiaoxu Kang**, Anil Maybhate, Xiaofeng Jia, Romergryko G. Geocadin, Nitish V. Thakor, et al. "Multi-Scale Entropy analysis of EEG in post-cardiac arrest recovery with therapeutic hypothermia". The Biomedical Engineering Society 2008 Annual Fall Meeting.
- **Xiaoxu Kang**, Xiaofeng Jia, Anil Maybhate, Romergryko G. Geocadin, Nitish V. Thakor. "Neuro-protective effect of therapeutic hypothermia on neurological recovery post-cardiac arrest with multi-scale entropy analysis". The 38th Neural Interface Conference. Cleveland, OH, June 16-18th. (2008).
- Matthew A. Koenig, Xiaofeng Jia, **Xiaoxu Kang**, Romergryko G. Geocadin, Nitish V. Thakor. "Intraventricular Injection of Orexin-A Elicits Earlier Arousal in Rats after Cardiac Arrest". American Heart Association Resuscitation Science Symposium. Circulation, Oct 2008; Vol: 118 Iss: 18 Supplement: S_1468 - S_1469. (2008)

- **Xiaoxu Kang**. "Research progress in the growth, mechanisms, properties and applications of zinc oxide nanostructures". 5th China International NanoScience and Technology Xi'an Symposium (2006)

PROFESSIONAL SERVICES

- **Reviewer**: Journal of Computational and Mathematical Methods in Medicine; IEEE Journal of Translational Engineering in Health and Medicine
- **Teaching Assistant**, "Natural Language Processing" by Dr. Jason Eisner (Department of Computer Science), Johns Hopkins University, Fall, 2011
- **Teaching Assistant**, "System Bioengineering II" (Department of Biomedical Engineering), Johns Hopkins University, Spring, 2011
- **Teaching Assistant**, "Random Signal Analysis" by Dr. Sanjeev Khudanpur (Department of Electrical and Engineering Department), Johns Hopkins University, Fall, 2010
- **Teaching Assistant**, "Biomedical Instrumentation" by Dr. Nitish V. Thakor (Department of Biomedical Engineering), Johns Hopkins University, Spring, 2010
- **B'More Ambassador/Instructor**, B'more Freshmen Academic and Personal Enrichment Program, Johns Hopkins University, 2010
- **Teaching Assistant**, "Medical Device and Sensors" by Dr. Nitish Thakor, Johns Hopkins Applied Physics Laboratory (APL) Fall, 2008-2009