MODELING COMPLEX BIOLOGICAL AND MECHANICAL MOVEMENTS: APPLICATIONS TO ANIMAL LOCOMOTION AND GESTURE CLASSIFICATION IN ROBOTIC SURGERY

by

Shahin Sefati

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

Baltimore, Maryland

August, 2014

Abstract

Mutual interaction between biology and robots can significantly benefit both fields. The richness and diversity in animal locomotion and movement provides an extensive resource for inspiration in engineering design of robots. On the other hand, biomimetic and bio-inspired robots play a critical role in testing hypotheses in biology and neuromechanics. Modeling complex biological and mechanical movements is at the core of this mutual interaction. Models and analytical tools are required for decoding and analysis of behavior in biological and mechanical systems, both at low level (sensory systems and control) and high level (activity recognition). This dissertation is focused on modeling approaches for biological and mechanical movements. We first primarily focus on physics-based template modeling to answer a long-standing question in animal locomotion: why do animals often produce substantial forces in directions that do not directly contribute to movement? We examine the weakly electric knifefish, a well-suited model system to investigate the relationship between mutually opposing forces and locomotor control. We use slow-motion videography to study the ribbon-fin motion and develop a physics-based template model at the task-level for

ABSTRACT

tracking behavior. Using the developed physics-based model integrated with experiments with a biomimetic robot, we demonstrate that the production and differential control of mutually opposing forces is a strategy that generates passive stabilization while simultaneously enhancing maneuverability, thereby simplifies neural control.

The second part of this work aims to propose a more general data-driven system-theoretic framework for decoding complex behaviors. Specifically we introduce a new class of linear time-invariant dynamical systems with sparse inputs (LDS-SI). In the proposed framework, at each time instant, the input to the system is sparse with respect to a dictionary of inputs. In the context of complex behaviors, the dictionary may represent the dictionary of inputs for all possible simple behaviors. We propose a convex optimization formulation for the state estimation with unknown inputs in LDS-SI. We derive sufficient conditions for the perfect joint recovery and explore the results with simulation. We demonstrate the power of the proposed framework in the analysis of complex gestures in robotic surgery. Results are better than state-of-the-art methods in joint segmentation and classification of surgical gestures in a dataset of suturing task trials performed by different surgeons.

Primary Reader: Noah J. Cowan

Secondary Reader: René Vidal, Marin Kobilarov

Acknowledgments

First and foremost, I want to thank my adviser, mentor, teacher, and colleague Dr. Noah Cowan. I enjoyed every single conversation with him from the first phone interview throughout the years at Hopkins. His brilliant thoughts, extreme enthusiasm of science and engineering, and highest standards have always amazed me and motivated me. I would also like to thank my thesis committee Dr. René Vidal and Dr. Marin Kobilarov for their valuable comments and discussions.

This work would have not been possible without the support and close collaboration of many people. I would like to thank our collaborators Dr. René Vidal of Johns Hopkins University, Dr. Eric. S. Fortune of New Jersey Institute of Technology, and Dr. Malcolm I. MacIver of Northwestern University. I would also like to thank Izaak Neveln, Dr. Bijan Afsari and Narges Ahmidi for fruitful discussions and providing me help with experiments, data collection.

I am very thankful to my labmates at LIMBS lab, past and present: Manu Madhav, Mert Ankarali, Alican Demir, Erin Sutton, Robert Nickl, Ravikrishnan Jayakumar, Sarah Stamper, Eric Tytell, Erion Plaku, Jusuk Lee, John Swensen, Avik De

and Eatai Roth. I would also like to thank many great friends, staff, and faculties at JHU. Thank you the administrative staff: Mike Bernard in ME department, Alison Morrow and Jamie Meehan in LCSR. Thank you Michelle and David for keeping me caffeinated.

I had the privilege to meet great friends in ME department and LCSR during my six year journey at Johns Hopkins, friends who made this journey memorable: Ehsan Basafa, Pezhman Foroughi, Soroosh Solhjou, Kevin Wolfe, Giancarlo Troni, Baptiste Coudrillier, Tian Xia, Tricia Gibo, Tom Wedlick, Min Yang Jung, Yixin Gao, Xingchi He, Yan Yan, Topher Mcfarland, Mike Kutzer, Berk Gonenc, Tutkun Sen, Jonathan Bohren and many others. I would like to extend my thanks to all of my wonderful non-JHU friends whose list of their names takes pages. I would also like to thank Farhad Pashakhanloo and Naeem Masnadi with whom I have spent a lot of memorable moments during the past few years.

Last and most important, family. I thank my beloved parents for their endless unconditional support, love, and sacrifice. Certainly, words can not express my gratitude. I thank my brother, Shahriar, for being my best friend in my life. And finally, I thank Sadaf for her encouragement, support, and love.

Dedication

To my beloved parents, Beji and Ali.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Animals and human produce extremely rich and robust behaviors in often complex environment. The richness and diversity in animal locomotion and movement provides an extensive resource for inspiration in engineering design of robots. On the other hand, while even the most advanced bio-mimetic and bio-inspired robots still are far behind their biological counterparts (in terms of robustness, sensing etc.), robots also play a critical role in testing hypotheses in biology and neuromechanics. Mutual interaction between biology and robots can significantly benefit both fields. Modeling complex biological and mechanical movements is at the core of this mutual interaction. Models and analytical tools are required for decoding and analysis of behavior in biological and mechanical systems, both at low level (sensory systems and control) and high level (activity recognition).

In the following sections, I discuss two common frameworks for modeling biological

and mechanical movements: physics-based and data-driven approaches. At one side of the spectrum, physics-based models derived, as the name suggests, from fundamental laws of physics, are referred to as white box modeling approach. At the other side of the spectrum, data-driven models are referred to as black box modeling approach. I also briefly review how *template* and *anchor* models are utilized in physics-based modeling of animal locomotion.

1.1 Physics-based and Data-driven Modeling of Complex Movements

Developing a model from first principles, e.g. Newton's laws, is sometimes a very effective modeling approach for describing the dynamics of a physical system. For example, a system of ordinary differential equations derived from fundamental physics laws can accurately describe the dynamics of a set of connected masses, springs, and dampers. Low-dimensional physics-based models have a long history in modeling animal locomotion [1–4] (see Section 1.2 for more details). For instance, one of the widely used so-called models in legged locomotion is the spring-loaded inverted pendulum (SLIP) model for describing human walking dynamics in the sagittal plane [2, 5]. Such a simple model, consisting of only a point mass and a spring, is certainly not rich enough to encompass the whole range of walking dynamics. For instance, the model excludes muscles and sensing; nevertheless, it provides accurate predictions for

center-of-mass dynamics during steady-state locomotion [5].

Examining locomotion at a lower level reveals that muscles create forces in animals (actuators create forces and torques in robots), the body has inertia and damping, and the environment exerts reactions forces on the body. Thus the observed motion of an animal is the result of these complex and often nonlinear interactions. Therefore, at this level, locomotion should be investigated using more detailed high-dimensional models. Closed-loop analytical solutions to physics-based models generally do not even exist at such complexity. However, with recent technological advances in computational power, high-fidelity computational models developed from laws of physics can provide accurate predictions of locomotion dynamics [6].

While physics-based models have proven to be successful in modeling the dynamics of biological and mechanical movements, there are limitations. Physics-based approaches for modeling complex behaviors at lower levels (e.g., the spiking activity of all motor neurons) may lead to a very complex model that does not accurately capture high-level behavior. This might be in part due to the lack of knowledge of reliable physical models of the components, and / or due to lack of knowledge of the topology of interacting components.

Alternatively, data-driven system identification approaches aim to directly identify a dynamical model based on empirical data. In general, data-driven system identification may take a *black box* approach in which only a general model structure is assumed (say, an ODE or frequency response function). Alternatively, in the

so-called *grey box* approach, certain physics-based models might be integrated with data-driven system identification. In this case, prior knowledge about the underlying dynamical model informs and constrains system identification.

Most of classical system identification techniques exploit models by empirically measuring both input and output data. In biology, for instance, detailed quantification of behavior can be achieved by restricting the behavior to a limited number and complexity of inputs and outputs. While input–output system identification is preferable, in many cases such as the analysis of complex biological and mechanical movements at lower levels, the input data is not accessible and more unified modeling frameworks are required. Such "blind" system identification requires the development and application of novel tools and techniques, as well as careful mathematical characterization of the available observations.

1.2 Template and Anchor Models of Locomotion

Locomotion is often a result of complex and nonlinear interaction between animal and its environment. Despite the apparent complexity, many behaviors seem to result from comparatively simple, low-dimensional patterns of movement. Low-dimensional, task-specific models for the locomotor mechanics enable the application of control systems analysis to decode the neural mechanisms for sensorimotor processing [1–4].

These simple *descriptive* mechanical models, sometimes termed "templates" [2, 5], are essential for understanding stability and control in biological systems and bio-inspired robots [1, 7–10]. Template models are often low-dimensional simplified *grey box* physical model that describe some of the salient features of the behavior at the task level. As mentioned above, the dynamics of center-of-mass in legged locomotion is well approximated by a simple point mass and spring in sagittal plane (SLIP model). While the dynamics of the SLIP model is governed by the fundamental laws of physics, such an oversimplified model is based on empirically refutable hypotheses [5]. On the other hand, more elaborate and realistic high-dimensional models, sometimes termed "anchors" [2,5] can facilitate the exploration of more detailed questions about closed-loop control.

1.3 Thesis Organization

This dissertation is presented in two parts. In Part I we primarily focus on *physics-based template* models and develop a task-level template model of the biomechanics of the ribbon-fin of weakly electric knifefish during tracking behavior to understand the role of mutually opposing forces during locomotion. Animals often produce substantial forces in directions that do not directly contribute to movement. For example, running and flying insects produce side-to-side forces as they travel forward. These forces generally cancel out, and so their role was a mystery. To investigate

the relationship between antagonistic forces and locomotor control, we examined the weakly electric glass knifefish *Eigenmannia virescens*, a well-suited model system that produces mutually opposing forces using a single elongated fin. We develop a *physics-based template* model of the biomechanics of the ribbon-fin of weakly electric knifefish during tracking behavior. The mechanics-based model is integrated with biological experiments and force measurements from a biomimetic robot in a multidisciplinary approach. We show that mutually opposing forces can enhance both maneuverability and stability at the same time, although at some energetic cost. In addition to challenging the maneuverability–stability dichotomy within locomotion, our results challenge the same tradeoff within the engineering of mobile robots. This may inspire the exploration of a new set of strategies for the design and control of mobile systems.

The presented results in Part I also benefits the neuromechanical field. The mechanics of locomotion dictates the control problem confronted by the nervous system. The glass knifefish perform a behavior—refuge tracking—that can be modeled as a single degree of freedom behavior, greatly facilitating neuromechanical control systems modeling. Toward the end of Part I of this thesis we revisit a control-theoretic framework for making neural control predictions in glass knifefish, and provide a discussion based on the validated plant model developed in this work for tracking behavior.

In Part II of this dissertation we aim to propose a general framework with implications in the *data-driven* analysis of complex biological and mechanical movements.

Specifically, we introduce a new class of linear time-invariant dynamical systems with sparse inputs. Linear dynamical systems are widely used to model time-series data including simple human movements, dynamic textures, and surgical video data. In all mentioned examples the output is modeled as a stationary ARMA process. We argue that most biological and mechanical movements are categorically non-stationary. On the other hand, sparse representation theory has a long history in signal processing community. Sparsity embodies the notion that quite often very complex (and high-bandwidth) signals can be represented as a combination of surprisingly few basis vectors. While notion of sparsity has been incorporated in the context if linear dynamical systems, most of the prior works assume that the states, initial conditions, or innovations are sparse. In our view, we believe these notions of sparsity are fundamentally limited. We formally introduce the linear dynamical systems with sparse inputs (LDS-SI) in chapter 7. We then study a fundamental problem of state estimation with *unknown* (and non-stationary) inputs by proposing a convex optimization problem. We derive theoretical sufficient conditions for perfect joint recovery of initial state and unknown inputs. We also present the simulation results for perfect recovery.

Finally, we demonstrate the power of the proposed framework in the analysis of complex gestures in robotic surgery. More specifically, we show that a classification algorithm based on a very simple linear dynamical system with sparse inputs improve the state-of-the-art methods in segmentation and gesture classification of a data set including suturing task trials recorded by da Vince surgical robot system.

Part I

Task-Level Dynamical Model For Tracking Behavior in *Eigenmannia virescens*

Chapter 2

Motivation

2.1 Why task-level mechanics-based template models? Why biomimetic robots?

The nervous system processes the sensory information for closed-loop control of task-level locomotion, such as tracking behavior [11,12]. The mechanical plant defines the way motor signals are transformed into forces and movements, and so discovering the neural controller [13–17] of a biological system greatly benefits from a task-specific mechanical model of the underlying locomotor dynamics [3,7,11]. Low-dimensional, task-specific models for the locomotor mechanics enable the application of control systems analysis to decode the neural mechanisms for sensorimotor processing [1–4]. These simple descriptive mechanical models, sometimes termed "templates" [2,5], are

essential for understanding stability and control in biological systems [1, 7, 8].

More elaborate models, sometimes termed "anchors" [2, 5] can facilitate the exploration of more detailed questions about closed-loop control. Multidisciplinary approaches integrate computational models and experiments with biomimetic robots to study the locomotor mechanics in more details and with higher accuracy. With advances in computing, high-fidelity simulations have categorically improved our understanding of various locomotor strategies in different species [18–21]. On the other hand, biomimetic robots enable us to experimentally validate the mechanical models [7,19,22,23], and to explore the effect of parameters beyond their biological ranges, providing insight as to where the biological performance lies within the range of the wider range of possible mechanical solutions [7, 24].

Although there are many task-level modelings for flight control [25–27] and terrestrial locomotion [1, 2], to date there are remarkably few experimentally validated task-level plant models for swimming fish [7], despite the fact that the mechanics of fish locomotion has been widely studied for decades [13, 28–31].

2.2 Mutually Opposing Forces in Locomotion

Animals routinely generate substantial "antagonistic" (mutually opposing) forces during locomotion that either cancel out at each instant of time, or average to zero

over each gait cycle [26, 32–35]. This is surprising because the production of antagonistic forces requires the activation of muscles and yet, since the forces sum to zero, they do not contribute to movement of the center of mass of the animal. Such antagonistic forces are not only present during forward locomotion but also in hovering for animals such as hummingbirds, hawkmoths, and electric fish; these animals produce large antagonistic forces and exhibit extraordinary maneuverability during station-keeping [11, 16, 36, 37]. In this thesis, we demonstrate that active generation and differential control of such antagonistic forces can eliminate the tradeoff between stability and maneuverability during locomotion.

Stability is generally defined as the resistance to, and recovery from, disturbances to an intended trajectory [38]. While maneuverability can be defined in several ways [39,40], it is perhaps most generally recognized as the relative amplitude of the control signal required to change movement direction [41]. That is, if a small change in the control amplitude effects a rapid change in direction, the system would be considered highly maneuverable. The potential for a tradeoff between the resistance to changes in direction, and the ability to change direction, appears self-evident [35, 38, 41, 42], and indeed this tradeoff is considered a fundamental challenge for the engineering design of airborne, submarine, and terrestrial vehicles [42–45]. Many swimming, flying, and running animals, however, appear to use locomotor strategies that are extremely stable and yet facilitate the control of extraordinary maneuvers [26, 38, 46, 47].

To investigate the relationship between antagonistic forces and locomotor control,

we studied the glass knifefish *Eigenmannia virescens* that hovers and rapidly changes direction while producing mutually opposing forces using a single elongated fin. We developed a task-level mechanics-based model to study theses mutually opposing forces. The computational model—validated by force measurements from experiments with biomimetic robot—revealed that mutually opposing forces improve the fore–aft maneuverability and concurrently enhance the passive stability by providing a damping-like force to reject perturbations.

In addition to challenging the maneuverability–stability dichotomy within locomotion, our results challenge the same tradeoff within the engineering of mobile robots. This may inspire the exploration of a new set of strategies for the design and control of mobile systems.

2.3 Roadmap and Contributions

In Chapter 3 we examine the glass knifefish *Eigenmannia virescens*, a well-suited model system, which produces mutually opposing forces during a hovering behavior. Kinematics data of the knifefish ribbon-fin is quantified through a set of biological experiments.

Based on biological experiments explained in Chapter 3, Chapter 4 presents a task-level mechanics-based model for tracking behavior in *Eigenmannia virescens*. We also use a biomimetic knifefish robot to measure the forces generated by the fin.

Simulation results validated by experiments with the biomimetic robotic fin reveals that mutually opposing forces can eliminate the tradeoff between maneuverability and stability.

We conclude Part I in Chapter 5, and provide a discussion on the role of mechanics in decoding sensory systems.

2.4 Dissemination

During the process of completing Part I of this dissertation, portions of the work have been reported at several scientific meetings [23, 48, 49], a paper [7], and was reviewed in [50]. Some figures and text in this dissertation appeared in these publications.

Chapter 3

Counter-propagating Waves in *Eigenmannia virescens*

3.1 Why weakly electric knifefish?

To investigate the relationship between antagonistic forces and locomotor control, we studied the glass knifefish *Eigenmannia virescens*. These fish hover and rapidly change direction while producing mutually opposing forces using a single elongated fin (Figure 3.1(A)). Glass knifefish, like other knifefish, generate thrust force primarily through undulatory motions of an elongated anal fin [21, 30, 51]. The ribbon fin consists of 217 ± 27 ventrally pointing rays (Table 4 from [52]; all statistics are quoted as mean $\pm$ standard deviation unless otherwise noted), with each ray independently controlled by a set of muscles. These rays are oscillated in a plane transverse to

the body axis, and can be coordinated to produce a wave that travels longitudinally along the fin. In this study, we integrate biological experiments (Figure 3.2), computational modeling and experiments with a biomimetic robot (Figure 3.1(B) and Figure 4.1) in order to understand how the fish achieves both stability and maneuverability during rapid adjustments of its fore-aft position. *Eigenmannia* and other similar species of knifefish often partition their ribbon fin into two inward-counter-propagating waves [51]. The fin kinematics can be idealized as a pair of inward-traveling waves with parameters including oscillation frequency (f), wavelength (λ) and angular amplitude (θ) (Figure 3.1(C)). We term the point where these two waves meet the "nodal point". While much is understood about the kinematics and mechanics of unidirectional traveling waves in a fluid [13, 28–30, 53, 54], far less is known about counter-propagating waves [51, 55], particularly in relation to control.

3.2 Methods

3.2.1 Experimental apparatus

A schematic of the experimental setup is shown in Figure 3.2(A). An electric pump circulates water in the flow tunnel. A refuge machined from a 15 cm segment of 2-inch diameter PVC pipe was mounted parallel with the flow in the middle of the test section. The bottom half of the pipe was removed to allow the fish to be video recorded through a window on the bottom of the test section. The refuge was

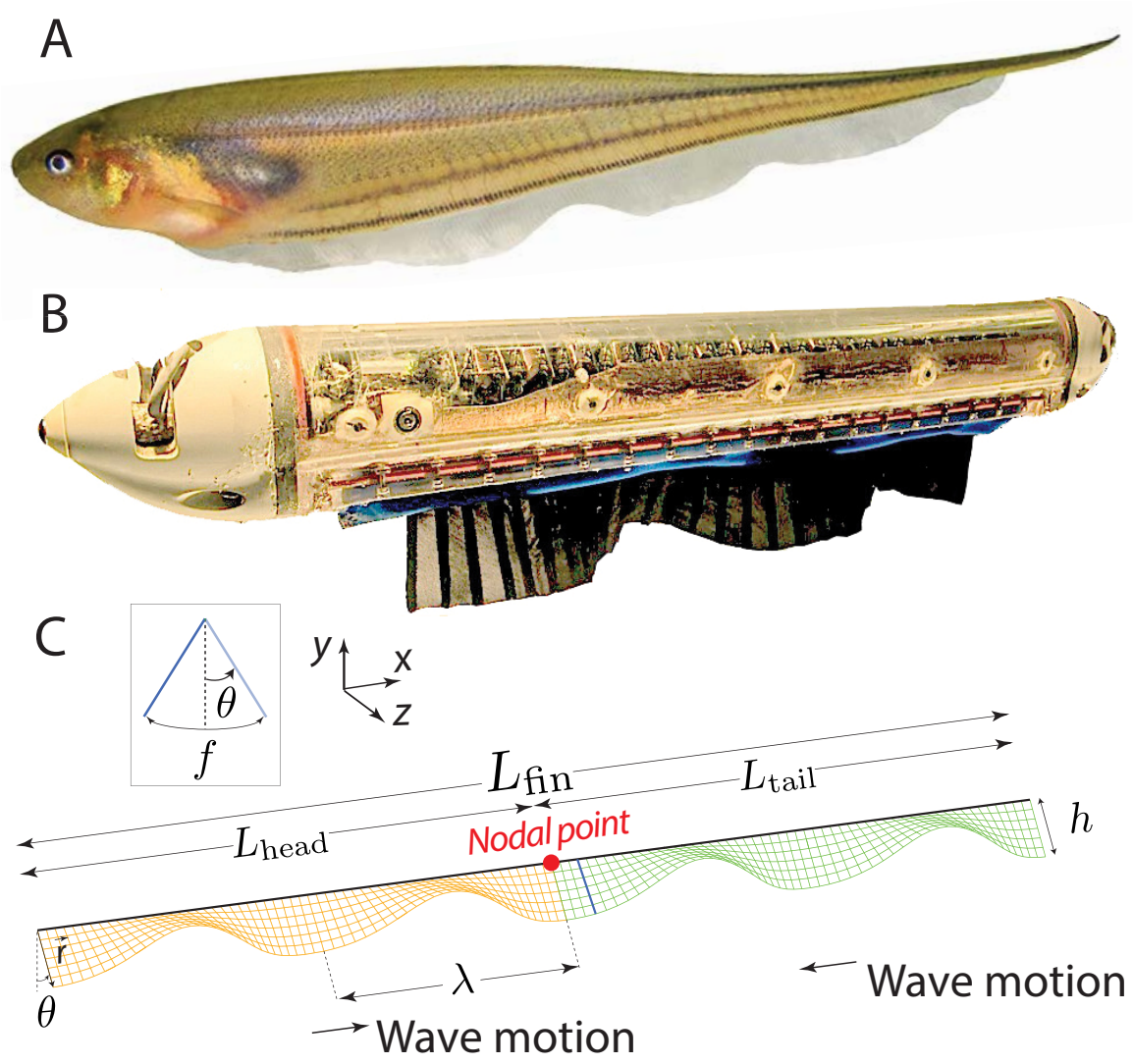

Figure 3.1: Three testbeds considered in this paper include the glass knifefish, a biomimetic robot, and a model of the swimming dynamics. (**A**) The glass knifefish *Eigenmannia virescens*. Experiments with a biomimetic robot match force measurements predicted by a computational model of ribbon-fin propulsion. (**B**) The biomimetic robot has a ventral ribbon-fin to emulate the fin of knifefish. The biomimetic robotic fin consists of 32 independently controlled rays, allowing for a wide range of fin kinematics such as counter-propagating waves. (**C**) The fin is modeled as a pair of inward-traveling waves. Directions of head and tail waves, and kinematics of the ribbon-fin are shown in this schematic: angular deflection (θ), wavelength (λ), lengths of the two waves (L_{head} and L_{tail}), length of whole fin (L_{fin}), temporal frequency (f), and nodal point (red circle).

positioned far enough away from the bottom of the tank to avoid boundary layer effects. A high-speed camera captured video from below.

The experimental test section of the flow tunnel (Figure 3.2 (A)) is approximately 90 cm long, 25 cm wide and 30 cm deep. Steady-state flow speed through the tunnel can be adjusted using a frequency controller connected to the electric pump. Flow speed in the test section was calibrated, as a function of pump frequency, by timing small drops of colored die as they traversed a known distance through the test section, at pump frequencies from 0 to 60 Hz in increments of 3 Hz that resulted in flow speeds from 0 to 15 cm/s (flow speed = 0.25×pump frequency, $R^2 = 0.996$).

A pco.1200s high-speed camera (Cooke Corp, Romulus, MI) with a Micro-Nikkor 60 mm f/2.8D lens (Nikon Inc., Melville, NY) captured video from below. The video was captured at 100 frames per second for all trials.

3.2.2 Biological experiments

Adult *Eigenmannia virescens*, obtained through commercial vendors, were housed in community tanks. Experiments were performed in the custom flow facility described above. In both the flow facility and housing tanks, water temperature was maintained at approximately 25-27°C, and conductivity was approximately 150-250 μS/cm. All experimental procedures were reviewed and approved by the Johns Hopkins University animal care and use committee and follow guidelines established by the National Research Council, the Society for Neuroscience, and previously estab-

lished methodologies [56].

Individual fish (N=5) were placed in the test section of the flow tunnel. Without training, the fish tend to swim into and stay inside the PVC tube [11, 57]. When we varied the steady-state flow speed, the fish typically remained stationary relative to the refuge. A single trial consisted of a fish remaining stationary in the tube by swimming forward (into the flow) at the flow speed. Trials were conducted at flow speeds from 0 to 12 cm/s in 1.5 cm/s increments. The order of these nine trials was pseudo-randomized, and three replicates (sets) of trials were collected for each individual, totaling 27 experiments per fish. Note that we only examined forward swimming for experimental convenience, since the fish often tend to reorient themselves into the flow. However, the fish readily swim both forward and backward when tracking a refuge ([11, 16]) and when they do swim backward, the nodal point shifts rostral to its 0 position as expected.

For each trial, several seconds of data were collected. Using open source code [58] written for MATLAB (The Mathworks Inc., Natick, MA, USA), the overall fore-aft position of the fish was tracked from the video. One second of data (100 frames) of steady-state swimming was selected by inspection of the position plotted as a function of time. This one second of data was used to quantify the kinematic parameters of both the rostral and caudal traveling waves. The nodal point, positions of both ends of the fin, and the peaks and troughs of the fin were manually digitized for each trial (Figure 3.2(B)). The fin height profile, $h(x)$, was digitized for each indi-

vidual fish (Figure 3.6) for use in the computational fluid model, below; fish were lightly anesthetized in buffered MS222 (Tricaine-S, Western Chemical, Inc., 0.2g/L) for photography.

These data were post-processed using a custom MATLAB script to compute the rostrocaudal nodal shift, wavelength, frequency, and amplitude of angular deflection of the two waves. For each trial, amplitude of angular deflection was fitted for each wave assuming it remains constant for all rays along each half of the fin.

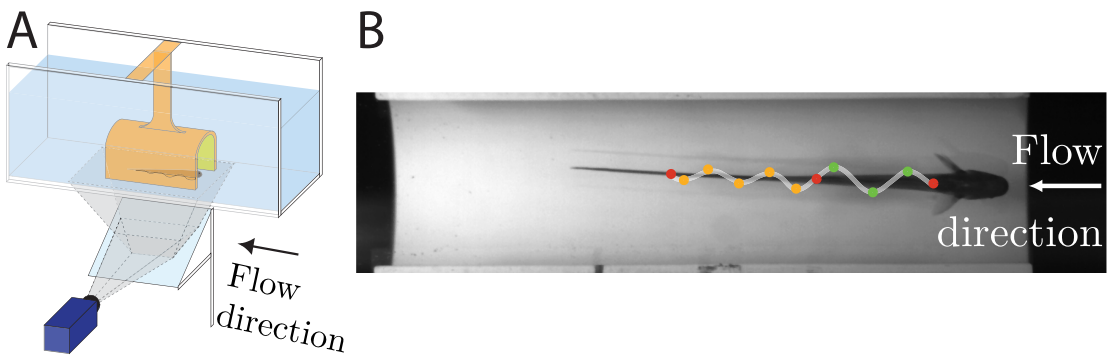

Figure 3.2: Experimental apparatus. (**A**) Steady state flow (0-12 cm/s) direction is shown. The fish keeps itself stationary relative to the PVC tube and kinematics of the ribbon-fin are recorded from below through an angled mirror. (**B**) One annotated frame recorded from the experiment is shown. Both ends of the fin and nodal point are shown in red. All peaks and troughs of head and tail waves are shown with green and orange dots, respectively.

3.2.3 Amplitude of angular deflection

In a 2D snapshot of ribbon-fin captured from bottom view, digitized peaks and troughs correspond to the fin rays that are oscillating with the amplitude of angular

deflection at that instant (orange and green circles in Figure 3.2 (B)). Using all digitized peaks and troughs in 100 video frames, an envelope curve was calculated for each wave. In Figure 3.6, envelopes for tail and head waves of a representative fish are shown in orange and green respectively. At each tested flow speed, the amplitude of angular deflections for all fin rays along the tail and head waves, namely θ_t and θ_h, were calculated by minimizing a sum of squared differences between the 2D projection of fin rays and the envelopes of digitized data from fin motion.

3.3 Results

3.3.1 Nodal point shift

Using high-speed videography at 100 frames per second, the kinematics of the ribbon fin of five fish were digitized during station keeping (Figure 3.2 and Figure 3.6). Individual fish were placed in the test section of the flow tunnel. When we varied the steady-state flow speed, the fish typically remained stationary relative to a refuge mounted in the flow tunnel. A single trial consisted of a fish remaining stationary in the refuge by swimming forward (into the flow) at the flow speed. For each trial and flow speed, we analyzed one-second intervals (100 video frames) while the fish maintained position. Trials were conducted at 9 flow speeds (flow moving from head to tail in all cases) between 0 and 12 cm/s in increments of 1.5 cm/s. The order of these nine trials was pseudo-randomized, and three sets (replicates) of trials were

collected for each individual, totaling 27 experiments per fish. The masses of the individuals averaged 2.80 ± 0.72 g. The fin and body lengths were 7.36 ± 0.57 cm and 11.59 ± 0.71 cm respectively. As shown in Figure 3.3(A), the ribbon fin typically organized itself into two inward-counter-propagating waves. In four trials at the highest speed tested (12 cm/s) the ribbon fin had transitioned into a single wave traveling from head-to-tail.

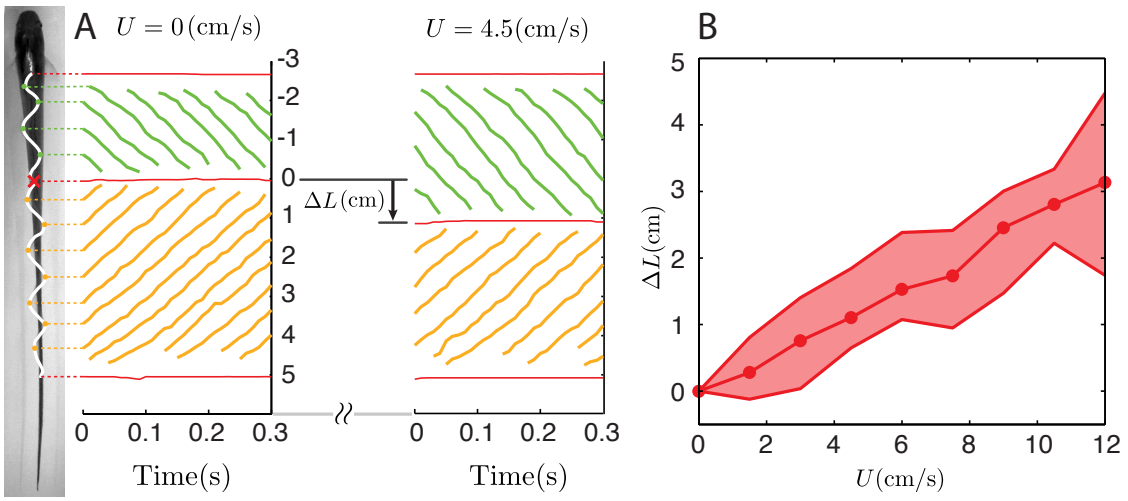

Figure 3.3: *Eigenmannia virescens* partitions its fin into two inward counter-propagating waves that produce antagonistic thrust forces. (**A**) Both ends of the fin and the nodal point (red cross), all peaks and troughs of the head wave (green circles), and all peaks and troughs of the tail wave (orange circles) were tracked during station keeping at different swimming speeds. The nodal position at $t = 0$ was taken as the reference for rostro-caudal position. Nodal point shift, ΔL, from 0 cm/s flow speed (no ambient flow) to 4.5 cm/s flow speed of a representative data set is shown in (A). (**B**) The nodal point shifts caudally as a function of flow speed approximately linearly. At each tested flow speed, the average over all replicates of data is shown with a filled circle. Shaded regions indicate the full range of nodal point shifts for all trials and all fish.

We found that the nodal point moved toward the tail as a function of increased head-on flow speed (Figure 3.3(B)). The nodal point shift, $\Delta L = L_{\text{flow}} - L_{\text{hov}}$, was

measured for each trial; here, L_{hov} corresponds to the nodal point position during hovering ($U = 0$) and L_{flow} corresponds to the test condition ($U > 0$). Other kinematic parameters varied less substantially with flow speed (Figure 3.4). The nodal point shift of one replicate from one fish was an outlier quantitatively and therefore was removed from statistical analyses (see section 3.3.4 and Figure 3.7). All other replicates from all fish were quantitatively similar within and across individuals.

3.3.2 Other wave parameters varied minimally with flow speed

For each trial, wavelengths of the tail and head waves, λ_t and λ_h, were computed by averaging the rostro-caudal distances between all adjacent pairs of peaks and troughs in each wave, over one hundred video frames; see Figure 3.4 (A). The tail and head wavelengths varied minimally as a function of flow speed, trending downward and upward slightly for tail and head waves respectively at the highest swimming speeds. Similarly, the maximum angular deflection of both waves varied minimally as a function of steady state flow speed; see Figure 3.4 (B). The temporal frequency of tail and head waves (f_T and f_H) was calculated for all trials. For the 4 trials at $U = 12$ cm/s there was only one single traveling wave from head to tail. Temporal frequencies averaged over all trials as a function of steady state swimming speeds are shown in Figure 3.4 (C). Using the data shown in Figure 3.4 (A) and (C), wave speed

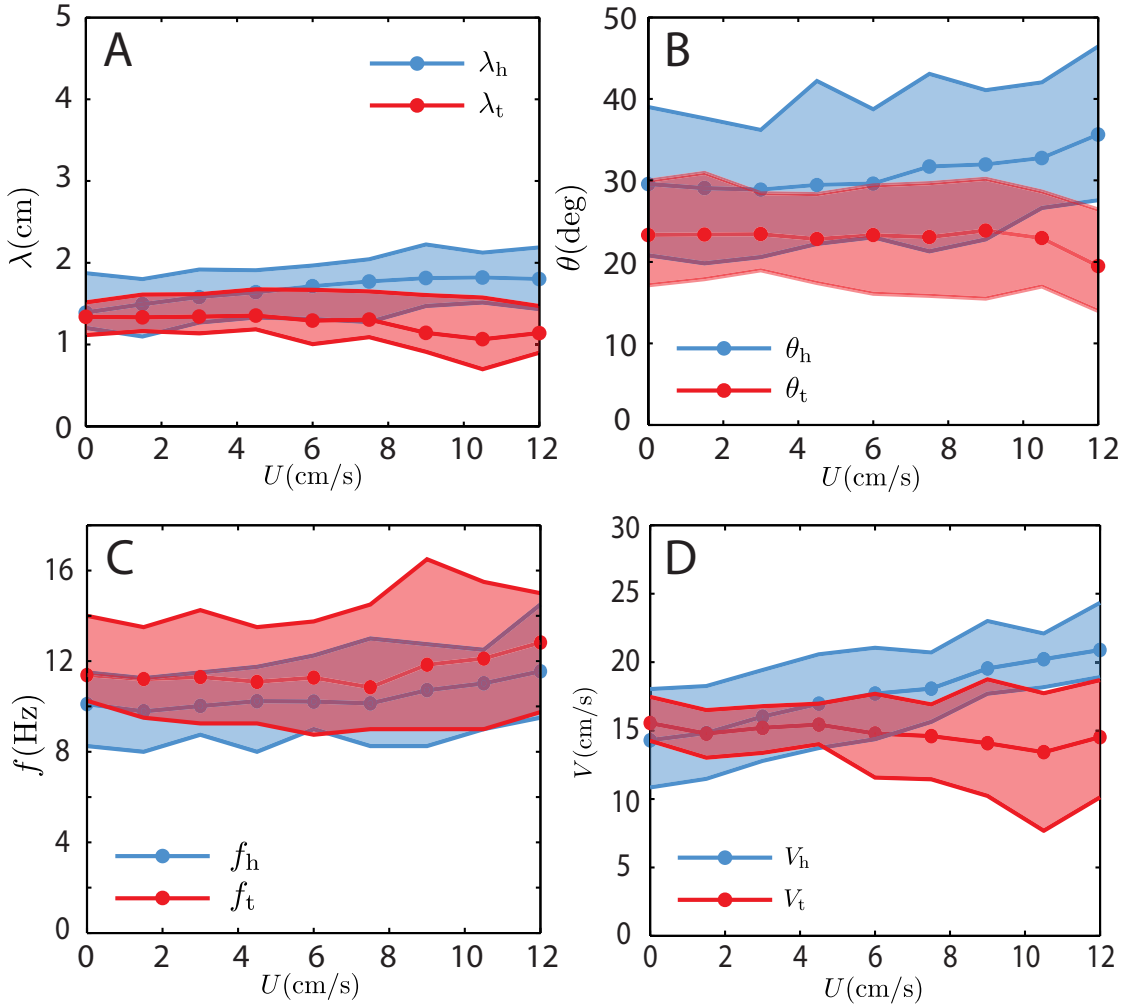

Figure 3.4: Ribbon-fin kinematics as a function of steady-state flow speed. At each tested flow speed, the average over all replicates of data is shown with a filled circle. Shaded regions indicate the full range of a given kinematic parameter for all trials and all fish. (**A**) Wavelength of the tail (red) and head waves (blue) remain nearly constant across flow speeds. (**B**) The angular amplitude of the tail wave (red), and head wave (blue) also remain nearly consistent across flow speeds, although there is a small trend, particularly for the tail wave. (**C**) Similar to wavelength and angular amplitude, the temporal frequency of the tail wave and head wave also remains nearly constant, particularly for lower swimming speeds. (**D**) Wave speed ($V = \lambda f$) of the tail (red) and head (blue) waves are roughly equal at lower swimming speeds.

($V = \lambda f$) of the tail and head waves are shown in Figure 3.4 (D). At lower swimming speeds, where we believe the counter-propagating waves strategy is the dominant mechanism for control, the wave speeds of the two waves (product of wavelength and temporal frequency) are very similar. Note that despite differences in frequency and wavelength, these differences result in roughly equal wave speed in the two waves, i.e. the two waves travel at approximately the same speed during slow swimming (i.e. near hovering). The difference becomes more significant at higher swimming speeds. While beyond our present scope, this deviation could possibly be explained by the transition from counter-propagating waves strategy to single traveling wave strategy. In other words although nodal shift serves as the dominant strategy for modulating the thrust force at low speed swimming, the role of other kinematics such as frequency and wavelength may become important at higher swimming speeds. Lastly the tail wave becomes very short (shorter than one complete wavelength) at the highest swimming speeds tested. As a result there may be subtle artifacts associated with estimating the tail wavelength in the digitization process.

In Figure 3.4 the shaded regions depict the entire range of variation across all trials and all individuals. While there was moderate variability across individuals, each individual was extremely consistent. At each tested flow speed, the standard deviation of the angular deflection for each of the five individual fish was between 0.1 to 7.2 deg for the tail wave (θ_t), and 0.1 to 4.7 deg for the head wave (θ_h). Similarly, the standard deviation of the wavelength for each of the five individual fish

was between 0.1 to 0.26 cm for the tail wave (λ_t), and 0.1 to 0.3 cm for the head wave (λ_h). The standard deviation of frequency for each of five individual fish was between 0.1 to 2.3 Hz for the tail wave (f_t), and 0.1 to 1.6 Hz for the head wave (f_t). The standard deviation of wave speed for each of five individual fish was between 0.02 to 4.85 cm/s for the tail wave (V_t), and 0.01 to 1.84 cm/s for the head wave (V_h).

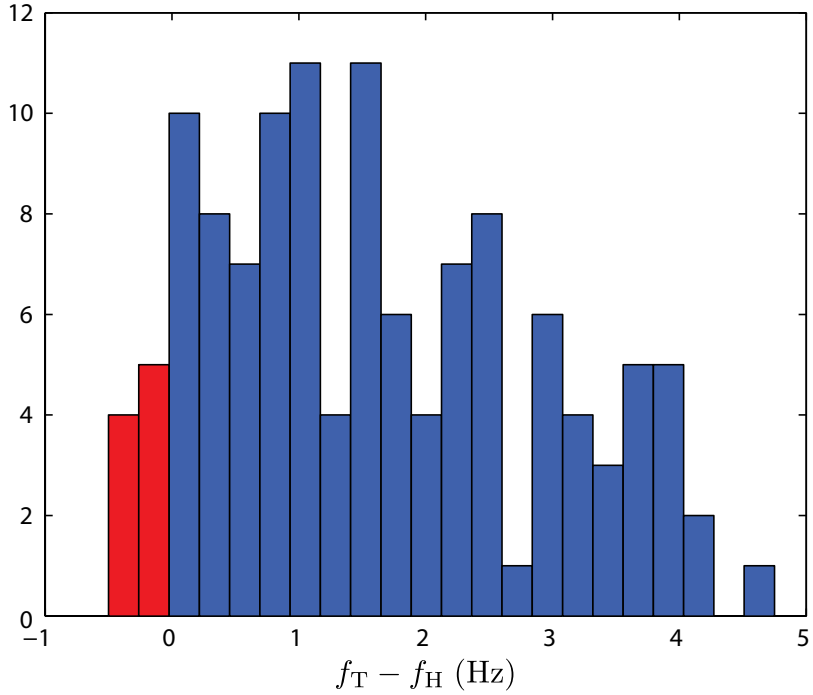

Figure 3.5: Histogram of temporal frequency difference between tail and head waves. Blue and red bins correspond to trials where the tail wave has a higher and lower temporal frequencies respectively.

An unexpected and interesting finding was that the two distinct waves often oscillated at different frequencies, causing the two distinct waves to roll in and out of phase with one another at the nodal point. The temporal frequency difference between tail and head waves ($f_T - f_H$) was calculated for all trials except for the 4 trials

at $U = 12$ cm/s for which there was only one single traveling wave. Trials from the outlier replicate described above were excluded from the analysis. In 84.4% of trials (shown in blue) the tail wave was oscillating at a higher frequency than the head wave. While beyond the scope of the current study, this finding might be of interest in studying the central pattern generators (CPG), the neuronal circuits that produce multiple oscillatory patterns of muscle activity and rhythmic movements.

3.3.3 Ribbon-fin tapers at both ends

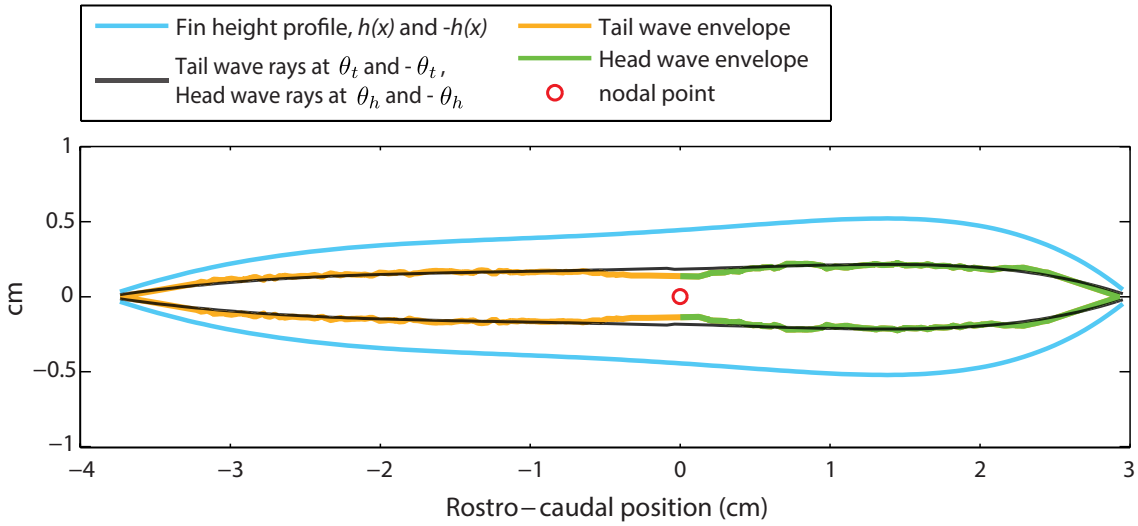

Figure 3.6: 2D bottom view: Ribbon-fin is tapered at both ends. The fin height profile was digitized for each individual fish. The fin height profile for a representative fish is shown in blue. Envelope of all digitized peaks and troughs is shown in orange (tail wave) and green (head wave). 2D visualization of the fin with fitted θ_t and θ_h is shown in black.

The height of the ribbon-fin is not constant along the body, and the fin is tapered at both ends. For use in a computational fluid model, and calculation of amplitude

of angular deflection of fin rays from the 2D motion of the ribbon-fin (fin motion was captured from the bottom view as explained in methods), the fin height profile, $h(x)$, was digitized for each individual fish. Fish were briefly anesthetized and positioned to capture a lateral image. The fin height profile of a representative trial is shown in Figure 3.6. The blue curve depicts $h(x)$ and $-h(x)$ of an individual fish. Envelope of all digitized peaks and troughs is shown in orange (tail wave) and green (head wave) during steady state swimming at $U = 3$ cm/s.

3.3.4 Outlier replicate in biological data

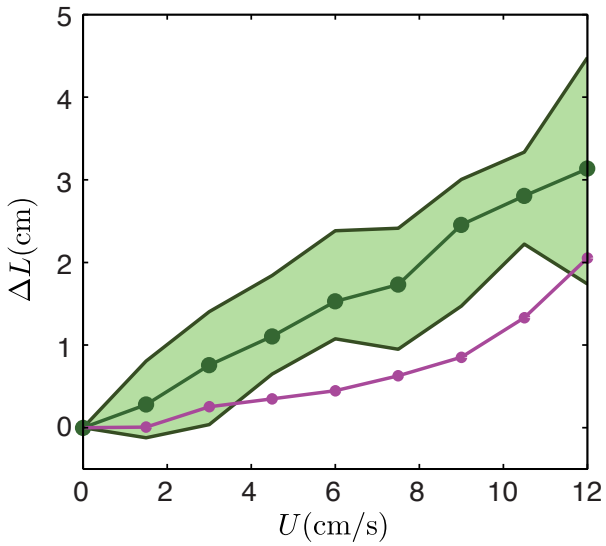

Figure 3.7: One replicate of data was removed from the statistics: Measured nodal shift for the outlier replicate is shown in purple.

As explained in 3.3.1, the kinematics of the ribbon-fin of five fish were digitized in this study and at each tested flow speed, three replicates of data were collected. The

nodal shift of the third replicate collected for Fish 3 followed the qualitative trends of other replicates, but was an outlier quantitatively. Measured nodal shift of this replicate is shown with a different color (purple) in Figure 3.7.

3.4 Discussion

Counter-propagating waves modulate fore-aft thrust for hovering: During station keeping, the net forces over the body include the antagonistic thrust forces generated by the head and tail waves, as well as the drag force over the body and pectoral fins. While nonzero net force is necessary for transient movement and unsteady swimming, net forces over the fish body must sum to zero during station keeping.

Eigenmannia modulates net thrust, generated by the two waves, mainly by moving the nodal point. When there is no ambient flow, the nodal point remains near the middle of the fin. If the ribbon fin were not tapered at its ends, and kinematic parameters of the two counter-propagating waves were identical, then in theory the nodal point would be exactly in the middle of the fin. For relatively slow flow speeds—under 12 cm / s in this study—the tail wave travels against the flow while the head wave travels along with, although faster than, the flow. Moreover, the nodal point moves caudally as the steady state swimming speed increases during upstream station keeping. This produces two competing effects in the amount of force generated by each wave: a change of length (and thus area) of each wave, and a change in the

relative velocity between the waves and the ambient flow. Although the tail wave (L_{tail}) shortens, the relative velocity between the tail wave speed ($V_t = \lambda_t f_t$) and flow speed (U) increases ($V_t - (-U) = U + V_t$). By contrast, although the head wave (L_{head}) lengthens, the relative velocity between the head wave speed ($V_h = \lambda_h f_h$) and flow speed decreases ($-V_h - (-U) = U - V_h$). As a result of these two competing effects—namely decrease/increase in fin length and increase/decrease in relative velocity between the ribbon-fin wave speeds and the flow speed—antagonistic forces generated by the two waves balance each other during station keeping.

Chapter 4

Task-level Dynamical Model and Bio-inspired Robotic Fin

4.1 Introduction

Glass knifefish *Eigenmannia virescens* hovers in place with extraordinary precision. An undulating ribbon-fin runs along the body, enabling knifefish to rapidly alternate between forward and backward swimming without changing body orientation. Knifefish routinely partition the ribbon-fin into two counter-propagating waves [7,59], recruiting the frontal portion of the fin to generate forward thrust (a wave traveling from head-to-tail) with the rear section (tail-to-head wave) generating opposing forces. In stationary hovering, these opposing forces cancel each other. These waves meet at the "nodal point". Observation from biological experiments in chapter 3 revealed

that glass knifefish modulates the net fore–aft force, primarily, by moving the nodal point.

To better understand the biomechanics of the ribbon-fin and the role of mutually opposing forces during locomotion, we developed a task-level mechanics-based model for counter-propagating waves and a lumped plant model for station keeping in *Eigenmannia virescens*. To validate our model and test our hypothesis we also used a biomimetic knifefish robot to measure the forces generated by the fin.

Task-level mechanics-based model and force measurements from experiments with biomimetic robot revealed that the net fore–aft thrust force varies linearly as a function of nodal point position, in contrast to the seemingly simpler strategy of single traveling wave in which the generated thrust force exhibits a nonlinear profile as function of temporal frequency or maximum angular deflection of the single traveling wave [7, 24]. Simulations validated by experimental results with biomimetic robot showed that the use of counter-propagating waves significantly improves the fore–aft maneuverability (by decreasing the control effort), and concurrently enhances the passive stability (stabilization without active feedback control) by providing a damping-like force to reject the perturbations, thus simplifies control.

4.2 Task-level Mechanics-based Template Model for Ribbon-fin

We approximated the fin kinematics using two sinusoidal traveling waves, as is standard for unidirectional waves [21, 54]. The angle, θ, between each fin ray and the sagittal plane oscillates, and the relative phase changes along the rostro-caudal axis producing a traveling wave, modeled as a sinusoid:

$$\begin{aligned} \theta_h(x,t) &= \theta_{h,\max} \sin\left(2\pi\Big(\frac{x}{\lambda_h} + f_h t\Big)\right), \\ \theta_t(x,t) &= \theta_{t,\max} \sin\left(2\pi\Big(\frac{x}{\lambda_t} - f_t t\Big)\right). \end{aligned} \tag{4.1}$$

Subscripts h and t stand for head and tail waves respectively, x denotes the coordinate along the rostro-caudal axis, λ_h and λ_t are the head and tail wavelengths, and f_h and f_t are the head and tail frequencies of fin oscillation. The kinematic parameters are depicted in Figure 3.1 (C).

The computational model used in this study is based on a fluid drag model. This model has been used in numerous numerical analyses [13, 53, 60], but this is the first time this model has been applied to counter-propagating waves. The model applies to flow regimes with high Reynolds number and neglects the fluid interaction. Under the conditions of the experiment, the Reynolds number ($Re = \frac{UL}{\nu}$) can be estimated in the range of 10^3 to 10^4 ($\nu_{water} = 10^{-6} m^2/s$, $L_{\text{fin}} \approx 0.1$ m, for $U \approx$ 1-10 cm/s).

Drag force applied to the propulsive infinitesimal element is given by:

$$d\vec{\mathbf{F}} = \frac{1}{2} C_D \rho dA [\vec{\mathbf{u}} \cdot \vec{\mathbf{n}_s}]^2 \vec{\mathbf{n}_s} \tag{4.2}$$

where C_D is the coefficient of the drag depending on the shape ($C_D \approx 2.5$ in this study, evaluated from robotic experiments), ρ is the density of the fluid, dA is the area of the infinitesimal element, and $\vec{\mathbf{n}_s}$ is the unit normal to the surface at the centroid of the infinitesimal element. See section 4.2.1 and 4.2.2 for the details of how this model is used to estimate the nodal point gain (Equation (4.17)) and damping constant (Equation (4.18)) for *Eigenmannia*, and leads to the plant model shown in Equation (4.15).

4.2.1 Computational simulation

Here, we describe a computational model for computing the net force produced by a single traveling wave. During the derivation below, we suppress the subscripts t and h, which indicate tail and head waves, respectively, until we compute the overall forces F_t and F_h in Equation (4.12), below.

As discussed above, drag force applied to the propulsive infinitesimal element is given by:

$$d\vec{\mathbf{F}} = \frac{1}{2} C_D \rho dA [\vec{\mathbf{u}} \cdot \vec{\mathbf{n}_s}]^2 \vec{\mathbf{n}_s} \tag{4.3}$$

where C_D is the coefficient of the drag, ρ is the density of the fluid, dA is the area of the infinitesimal element, and $\vec{\mathbf{n}_s}$ is the unit normal to the surface at the centroid of

the infinitesimal element (see below). The function $\vec{\mathbf{u}}(x, r)$ is the relative velocity of the centroid of the element on the fin and steady-state flow speed:

$$\vec{\mathbf{u}}(x, r) = u_{\text{fin}}(x, r) \pm U\vec{\mathbf{i}}. \tag{4.4}$$

The sign of the last term is negative for the rostral wave and positive for the caudal wave, and

$$u_{\text{fin}}(x, r) = r\frac{\partial\theta}{\partial t}\Big(\sin\theta\vec{\mathbf{j}} + \cos\theta\vec{\mathbf{k}}\Big) \tag{4.5}$$

where r is the radial distance from the base of the fin ray to the centroid of the infinitesimal element, x is the rostro-caudal coordinate of the element, and θ is angle of the fin for the two waves, as defined in Equation (4.1).

The two-dimensional surface of a rectangular ribbon-fin can be parameterized as a set of points in 3D:

$$\vec{\mathbf{H}} = x\vec{\mathbf{i}} - r\cos\theta\vec{\mathbf{j}} + r\sin\theta\vec{\mathbf{k}}. \tag{4.6}$$

Here, $x \in [L_{\min}, L_{\max}]$, is defined over the half-wave of interest, and the range of $r \in [0, h(x)]$ depends on the fin profile function, $h(x)$ (see section 3.3.3).

Geometric properties of the surface such as the unit normal vector, $\vec{\mathbf{n}}_s$, of the surface at each point can be derived from the metric tensor [61]:

$$\vec{\mathbf{n}}_s = \frac{1}{\sqrt{1 + r^2\theta_x^2}}[-r\theta_x\vec{\mathbf{i}} + \sin\theta\vec{\mathbf{j}} + \cos\theta\vec{\mathbf{k}}] \tag{4.7}$$

where θ_x is $\frac{\partial\theta}{\partial x}$.

Note that the normal, $\vec{\mathbf{n}}_s$, is defined relative to one side of the fin, but that at each local peak or trough of the fin, there is a switch in which side of the fin is traveling

"upstream"; this switch depends on both the sign of θ_x and the wave direction which is different for the head and tail waves. Thus, the normal vector for each infinitesimal element on the surface of the fin is either $\vec{\mathbf{n}} = \pm\vec{\mathbf{n}_s}$. Using the normal vector for each differential element, we have

$$\vec{\mathbf{u}} \cdot \vec{\mathbf{n}} = \frac{(\lambda f \pm U)}{\sqrt{1 + r^2\theta_x^2}}(r|\theta_x|) \tag{4.8}$$

$$= \frac{(V \pm U)}{\sqrt{1 + r^2\theta_x^2}}(r|\theta_x|) \tag{4.9}$$

where $V - U$ correspond to the head wave and $V + U$ correspond to the tail wave. Instantaneous net force is computed by integrating $d\vec{\mathbf{F}}$ over the half-wave of interest; the two half-wave forces can then be added to compute the total force on the fin. The time averaged force over one period of fin undulation is zero for y and z (lateral) components; this can be seen by the periodic y and z components in the unit normal vector shown in Equation (4.7). The time averaged thrust force generated by each half wave in the x direction can be computed by

$$\begin{aligned} |F_x| &= \frac{1}{T}\int_0^T \int_{L_{\min}}^{L_{\max}} \int_0^{h(x)} (d\vec{\mathbf{F}} \cdot \vec{\mathbf{n}}_x)\, dr\, dx\, dt \\ &= \frac{1}{T}\frac{\rho C_D}{2}\left(\frac{2\pi\theta_m}{\lambda}\right)^3 (V \pm U)^2 \\ &\quad \int_0^T \int_{L_{\min}}^{L_{\max}} \int_0^{h(x)} \frac{r^3|\cos 2\pi(\frac{x}{\lambda} \mp ft)|^3}{(1 + r^2\theta_x^2)^{3/2}}\, dr\, dx\, dt, \end{aligned} \tag{4.10}$$

where $\vec{\mathbf{n}}_x$ is the unit vector in the x direction, $V - U$ correspond to the head wave and $V + U$ correspond to the tail wave. In the analysis of forces generated by the fin, we use the kinematic parameters measured for each individual trial (frequency, wavelength, amplitude of angular deflection) which are assumed to remain constant

over one period of undulation. The fin height function for each fish, $h(x)$, is obtained via digitization of the fin profile (see section 3.3.3). The net thrust force, generated by the ribbon fin, is the summation of thrust forces generated by two half waves.

4.2.2 A plant model for station keeping in *Eigenmannia*

In addition to the detailed force analysis for each biological trial, in which the digitized height profile is taken into account, here we further approximate the fin in order to capture the essential structure of counter-propagating wave mechanics in a lumped-parameter model. We assume the fin has a rectangular profile, i.e. the height of fin is the same along the length of fin (note that this matches the morphology of our biomimetic robot). Since in Equation (4.10) we are averaging the force over one period of oscillation ($T = 1/f$), for a fixed r, the integrand becomes independent of variable x. So the time averaged generated thrust is $F \propto L_{\text{fin}}(V \pm U)^2$ where L_{fin} is the length of the fin, V is the wave speed along the fin and U is the steady state flow speed. Net force over the body includes the thrusts generated by the two waves, and drag force over the body:

$$F_{\text{net}} = F_t + F_h + F_{\text{drag}} \tag{4.11}$$

where subscripts t and h stand for tail and head respectively. If we take $L_{\text{head}} = L_{\text{tail}} = L/2$ as the reference for the nodal shift, $\Delta L = 0$, generated thrust by each

wave is:

$$\begin{aligned} F_t &= -a(L/2 - \Delta L)(V+U)^2\vec{\mathbf{i}} \\ F_h &= a(L/2 + \Delta L)(V-U)^2\vec{\mathbf{i}} \end{aligned} \tag{4.12}$$

where a is a constant. Simplified expression for the net thrust generated by the two waves is:

$$F_t + F_h = (\kappa + \gamma U^2)\Delta L - \beta U \tag{4.13}$$

where $\kappa = 2aV^2$, $\gamma = 2a$ and $\beta = 2aLV$. For low-speed swimming ($U^2 \approx 0$), the last equation can be further simplified to:

$$F_t + F_h = \kappa \Delta L - \beta U. \tag{4.14}$$

For low speed swimming the drag force over the body is also negligible ($F_{\text{drag}} \approx 0$). Moreover, during steady state swimming the net force over the body has to sum to zero, $F_{\text{net}} = 0$. Thus antagonistic forces generated by two waves should balance each other according to Equation (4.11) ($F_t + F_h \approx 0$). The second-order lumped model can be used as a task-level plant model of the ribbon fin for low speed refuge-tracking:

$$m\ddot{x} + \beta\dot{x} = u(t) \tag{4.15}$$

where β is the *damping constant*, and $u(t)$ is the net thrust generated by the ribbon-fin. In the case of counter-propagating waves $u(t) = \kappa\Delta L$, where κ is the *nodal shift gain*.

4.2.3 Linear quadratic controller to track a reference trajectory

The second-order lumped model (Equation (4.15)) can be written in state space:

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu(t) \tag{4.16}$$

where $A = [0\ 1; 0\ -b/m]$ and $B = [0; 1/m]$, with state vector containing the position and velocity $\mathbf{x} = [x; \dot{x}]$. By discretizing the linear system, an optimal affine control law exists according to Table 4.4-1 from [62].

4.3 Biologically Inspired Robotic Fin Experiments

We used a biomimetic knifefish robot [24, 55] to measure forces generated by counter-propagating waves as well as assess freely swimming control strategies in one dimension. Mechanical design constraints limited us to a larger length scale and longer time scale than *Eigenmannia*. The fin consisted of 32 individually actuated rays and measures 32.60 cm in length and 3.37 cm in depth. Figure 4.1 shows a schematic of the force experiments, where the robot is suspended from an air bearing platform from above. The platform was rigidly attached to mechanical ground through a 9 N single axis force sensor (Futek Advanced Senor Technology, Irvine, CA, USA) along

the fore–aft axis. The robot was fixed in all other translational and rotational axes. The working section of the flow tunnel was 80 cm long, 22 cm wide, and 28 cm deep.

In the first set of force measurements, we varied the nodal point of counter-propagating waves along the fin from −8.15 cm to 8.15 cm in increments of 1.63 cm (0 cm indicates the middle of the fin) while the robot was suspended in still water. Force measurements were gathered at 1000 Hz and averaged over 5 seconds after initial transients had dissipated. In the second set of force measurements, we varied the flow speed of the water tunnel from 0 to 10 cm/s in increments of 0.5 cm/s while keeping the nodal point of the counter-propagating waves fixed at 0 cm (in the middle of the fin). To test sensitivity to other kinematic parameters, we repeated both sets of force experiments with varied frequencies and angular amplitudes as shown in Table 4.1 and 4.2.

For fore–aft trajectory tracking experiments, we removed the force sensor to allow the robot to swim freely forwards and backwards, as shown in Figure 4.1(B). A linear encoder provided feedback on the position of the robot along the fore–aft axis of the water tunnel. At a cycle rate of 10 Hz, a microcontroller gathered this position feedback, derived robot velocity, calculated the control signal based on the control law described previously (linear quadratic controller), and sent the control signal over a serial line to the microcontroller dedicated to control of the robot rays. Position, time, and the control signal were logged for later analysis.

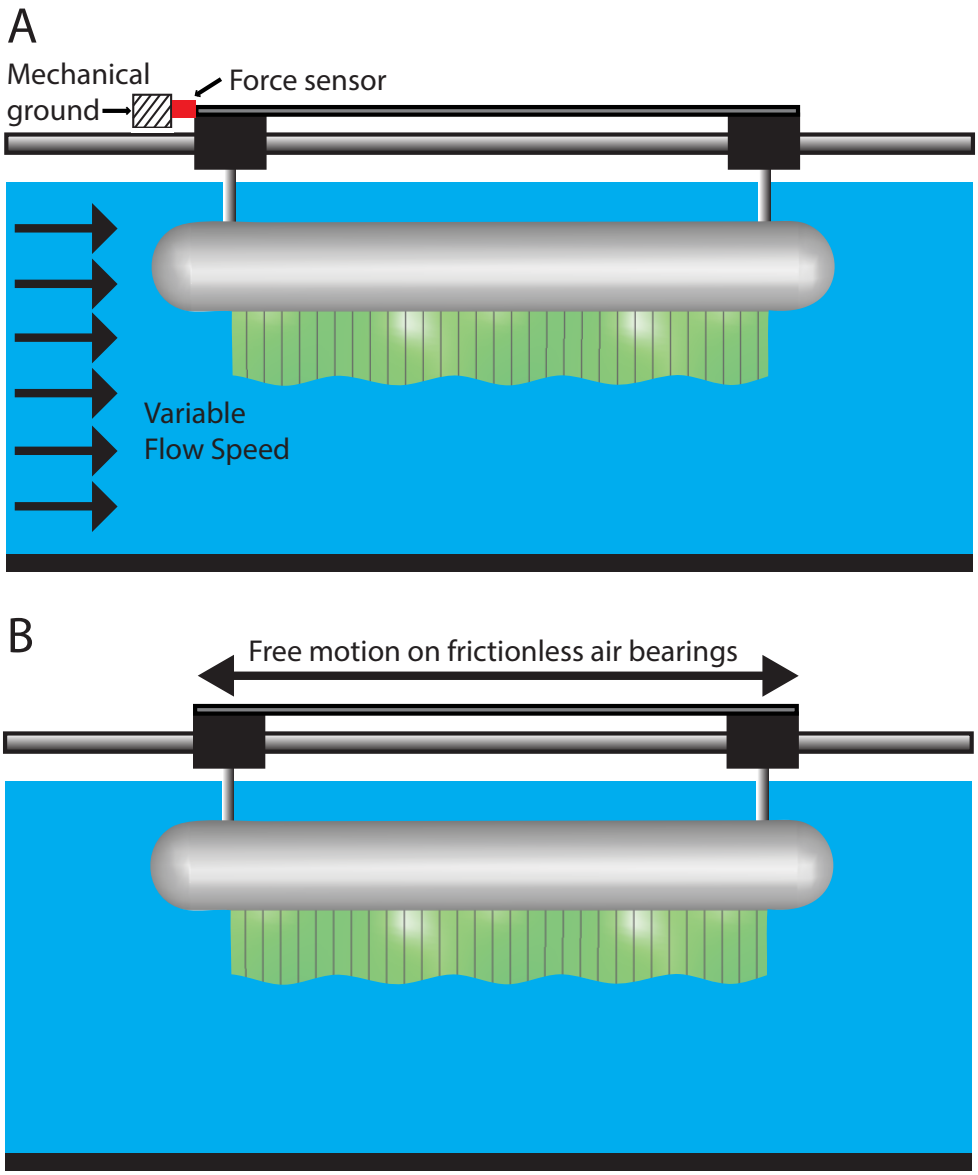

Figure 4.1: Experimental setup for the knifefish robot. (**A**) The knifefish robot was suspended into a water tunnel from a frictionless air-bearing system above. To measure force, the platform was rigidly attached to mechanical ground through a force sensor. Force measurements were collected for varying fin kinematics and flow speeds. (**B**) For the virtual refuge tracking experiments, the robot was allowed to move freely along the longitudinal axis. A linear encoder provided positional feedback of the robot. Experiments included controlling either fin oscillation frequency or nodal shift of counter-propagating waves to follow sinusoidal trajectories of varying frequency and amplitude.

4.4 Results

4.4.1 Mutually opposing forces during locomotion can eliminate the tradeoff between maneuverability and stability

The effect of nodal point position on the net thrust force generated by two inward-counter-propagating waves was investigated using a biomimetic robot (Figure 3.1(B) and Figure 4.1) and a computational model. In the first set of experiments with the biomimetic robot, the nodal point position was varied while other properties of the traveling waves were held constant (see section 4.4.2). Thrust forces generated by the two traveling waves were also predicted numerically. The measured forces as a function of nodal point shift closely match simulated forces from our model; see Figure 4.2(A). The thrust force varied linearly as a function of nodal point shift. We define the *nodal point shift gain*, κ, as the ratio of the measured net force to the nodal point shift:

$$\kappa = \frac{F_{\text{Thrust}}}{\Delta L}. \tag{4.17}$$

This parameter indicates the change in force given a unit change in nodal point position, and is used as a metric for fore-aft maneuverability of counter-propagating waves. Note that the nodal shift gain, κ, increases as a function of frequency (f) and angular amplitude of counter-propagating waves (θ) (Figure 4.6). κ increases

approximately quadratically with the frequency (Figure 4.2(B)).

We also discovered that passive damping emerges with counter-propagating waves. Specifically, a damping force opposing the direction of velocity perturbations increases linearly as a function of the speed of the animal relative to the flow. To measure this drag-like term in the robotic setup, the nodal point was held at the center of the robotic fin ($\Delta L = 0$), making the lengths of the fin dedicated to the tail wave (L_{tail}) identical to the length of fin dedicated to the head wave (L_{head}) (see section 4.7). The measured forces produced by the biomimetic robot vary linearly as a function of steady-state ambient flow and closely match simulated forces from our model; see Figure 4.2(C). Here we define the *damping constant*, β, as the ratio of the measured damping force, F, to the flow speed, U:

$$\beta = -\frac{F_{\text{Damping}}}{U}. \tag{4.18}$$

Larger values of the damping constant correspond to greater stability, in the sense that the time constant associated with recovery from perturbations is the ratio of inertia to damping [26, 46, 63]. Note that the damping constant increases with frequency (f) and angular amplitude (θ) of counter-propagating waves (Figure 4.7). In particular, the damping constant increases linearly with frequency (Figure 4.2(D)).

This damping force arises from body fore-aft velocity (longitudinal perturbations) when there are two inward-counter-propagating waves along the ribbon fin. Whole body fore-aft velocity causes asymmetries in net velocities of the counter-propagating waves (V) relative to the fluid (U). Depending on the direction of perturbation

the relative velocity for one half wave becomes $V - U$ while for the other becomes $V + U$. The resulting forces are proportional to the square of the relative velocities (see section 4.2.1). The net effect of these forces, which are individually quadratic in the relative velocity, is a net damping force that is *linear* in body fore-aft velocity. This damping force tends to reject velocity perturbations, as it opposes the direction of motion. Indeed, force measurements in a robotic experiment (explained above) reveal that such damping forces exist and vary linearly as a function of translational body velocities. Deceleration due to this passive linear damping force is proportional to the (perturbed) body velocity:

$$\ddot{x} \propto -\beta \dot{x}. \tag{4.19}$$

As a result, counter-propagating waves passively act to reject perturbations, resulting in an exponential decay of the body velocity.

As described above, the net thrust generated by two inward counter-propagating waves varies linearly as a function of nodal point shift, and the ability to change directions rapidly is captured by the nodal shift gain, κ (Figure 4.3(A)). By contrast, consider the problem of maneuvering using a single traveling wave that can reverse direction, as parameterized by the frequency, f. Here, negative frequency corresponds to a reversal of the traveling wave, thus resulting in negative thrust. As previously shown using the same biomimetic robot [24], our model indicates that force is nonlinear as a function of frequency, and is insensitive to changes in frequency near $f = 0$ (Figure 4.3(B)). Thus, using only a single traveling wave, the nonlinear relation

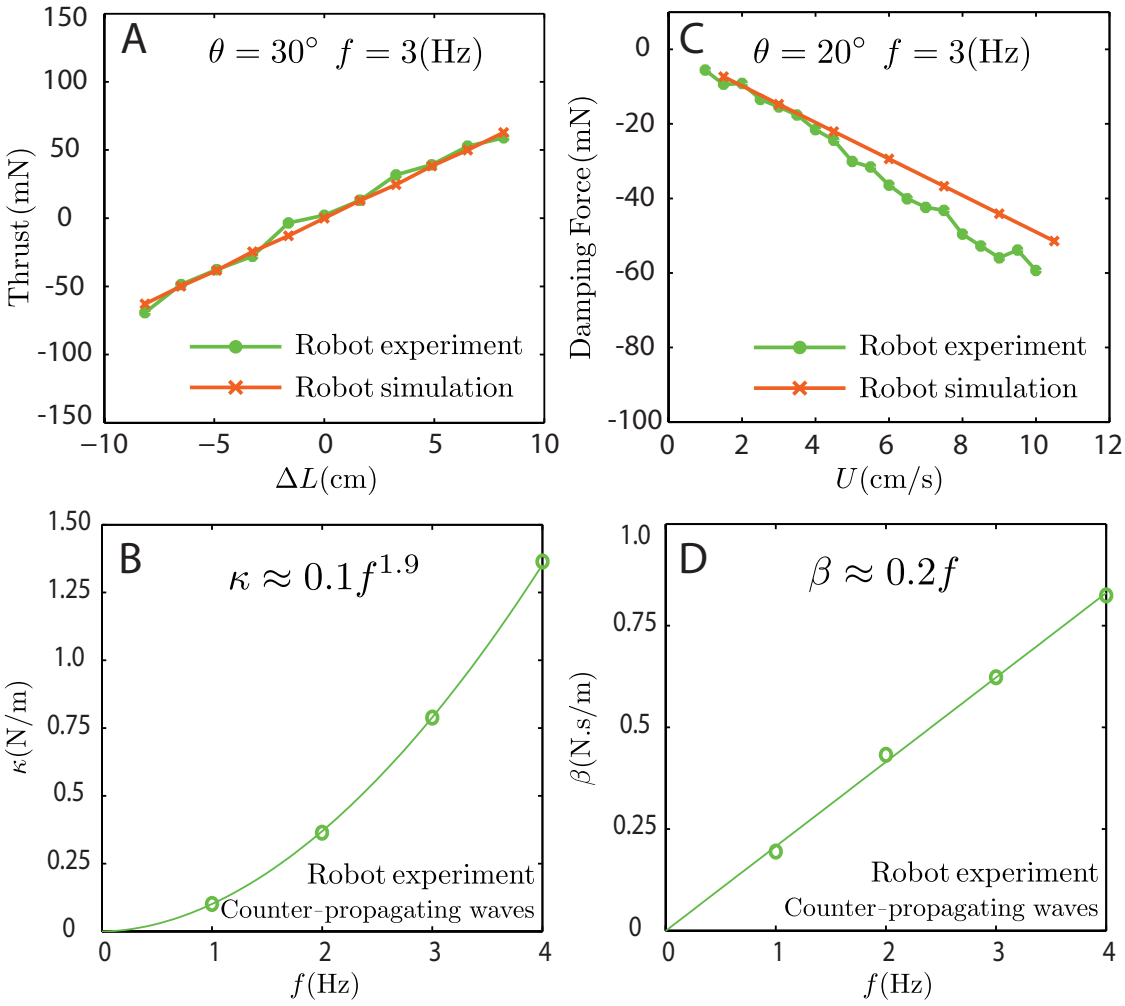

Figure 4.2: Biomimetic robot experiments and simulations. (**A**) Measured forces varied linearly as a function of nodal shift (ΔL). The slope is termed the *nodal shift gain*. (**B**) The counter-propagating waves were driven at four frequencies (see Table 4.1 (Set 1) for parameters). The nodal shift gain varied nonlinearly as a function of frequency. (**C**) Forces acting on the robotic fin varied approximately linearly as a function of steady-state flow speed when the nodal point was held in the middle of the fin ($\Delta L = 0$); the negative of the slope was termed the *damping constant*. (**D**) The damping constant varied linearly as a function of frequency (see Table 4.2 (Set 1) for parameters).

between force and the traveling wave speed (parameterized by f) creates an effect known in control systems theory as a "dead zone" [64]. In other words, modulating

the force around zero requires large changes in f for small changes in desired force, and hovering control requires rapid full fin reversal. Thus, modulating the thrust force by moving the nodal point might provide *Eigenmannia* with greater maneuverability during rapid changes in the direction of swimming when compared to changing the direction of a single traveling wave, as depicted in Figure 4.2(A-B).

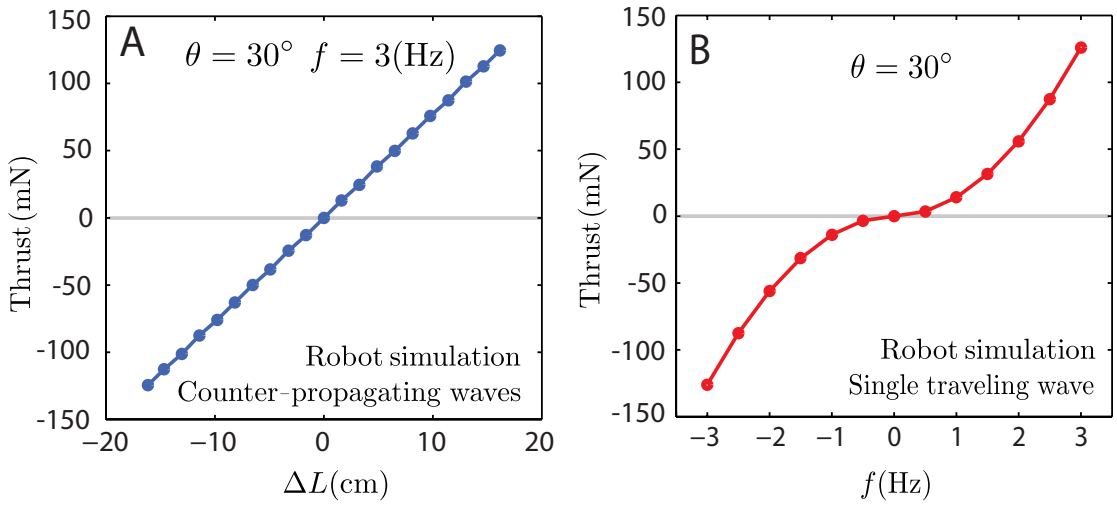

Figure 4.3: (**A-B**) Comparison of thrust generation by varying only one kinematic parameter predicted by the computational model. (A) Net thrust force is a linear function of nodal position. Nodal point is in the middle of the fin when $\Delta L = 0$. (B) Net thrust force by a single traveling wave along the fin is nonlinear with zero slope at $f = 0$, namely Force $\propto f|f|$. Negative frequency means wave direction is reversed. Note that near zero net thrust, large changes in frequency are required to generate small changes in force, since the graph has a slope of 0 at $f = 0$. The fin does not move when $f = 0$.

To test the ease of controlling rapid changes in direction in the biomimetic robot, we developed a simple lumped-parameter task-level dynamic model, or "plant", for

station keeping (Equation (4.15)) in section 4.2.2):

$$m\ddot{x} + \beta\dot{x} = F \tag{4.20}$$

where m is the robot's mass, β is the damping constant, and $F = u(t)$ is the net thrust force generated by the ribbon fin. The longitudinal position, velocity, and acceleration are denoted by x, $\dot{x}$, and $\ddot{x}$, respectively. We designed a linear quadratic tracking controller [62] to track a reference trajectory along the longitudinal axis. This is similar to the natural tracking behavior of electric knifefish [11, 16, 37]. Control inputs to the robot were chosen to be either nodal point shift (ΔL) for the counter-propagating wave strategy of thrust modulation, or frequency (f) for the unidirectional traveling wave strategy. For each desired amplitude (0.5 cm to 7.0 cm) and control strategy (single traveling wave versus counter-propagating waves), three replicate biomimetic robotic tracking experiments were conducted. Our hypothesis is that counter-propagating waves afford more maneuverability for small movements than a single traveling wave. If correct, the ratio of the control effort for using a single traveling wave compared to the control effort for using counter-propagating waves would sharply increase as the desired amplitude of the reference trajectory goes to zero.

Indeed, using both control policies (Figure 4.4(A-I, B-I)), the robot tracked the desired trajectory well (Figure 4.4(A-II, B-II)), but the ratio of the root-mean-square (RMS) of the normalized control signals, $(\frac{f}{f_{\max}})_{\text{rms}} : (\frac{\Delta L}{\Delta L_{\max}})_{\text{rms}}$, increased dramatically as the amplitude of movement decreased (Figure 4.5). That is, the nodal

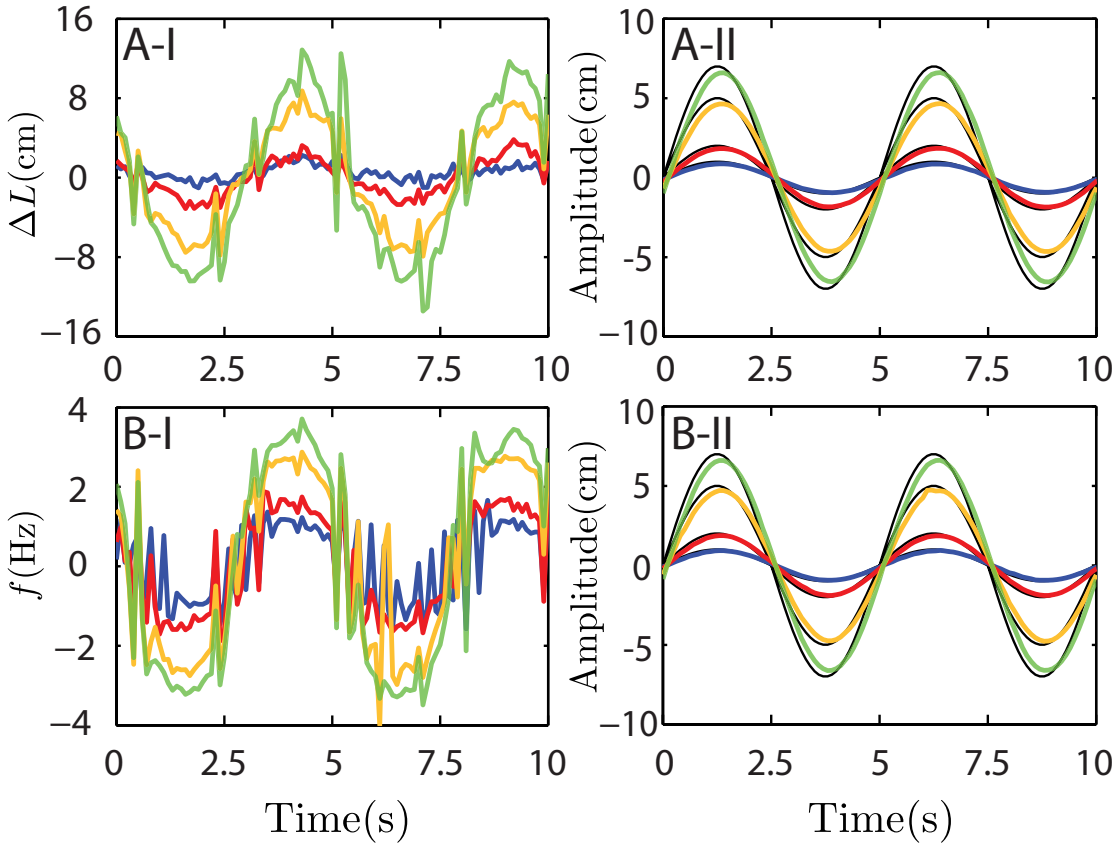

Figure 4.4: Comparison of tracking performance using two different control strategies. (**A-I, B-I**) The control signals (blue, red, orange, and green) for counter-propagating waves (ΔL) and a single traveling wave (f) are shown for four different reference trajectory amplitudes (A = 1 cm, 2 cm, 5 cm, and 7 cm, respectively). (**A-II, B-II**) The biomimetic robot positions (same color scheme) closely track the reference trajectories (black).

Table 4.1: Fin kinematic parameters for force measurement in the first robotic experiment.

Experimental set	f (Hz)	θ (deg)	# waves (L_{fin} / λ)
Set 1	1, 2, 3, 4	30	4
Set 2	3	20, 25, 30	4

point controller, compared to the unidirectional wave controller, renders the system increasingly more maneuverable as movement amplitude decreases, confirming our hypothesis (Figure 4.5). Using the validated computational model, the nodal point shift gain and damping constant corresponding to measured kinematics of *Eigenmannia* were also computed. Predicted control effort ratios for *Eigenmannia*, shown in Figure 4.5, reveal the same trend observed in biomimetic robot experiments.

4.4.2 Nodal shift gain in robot and *Eigenmannia*

Kinematic parameters used for the robotic experiment are shown in Table 4.1. In each trial, the net longitudinal force was measured as a function of nodal shift. Nodal position was varied from -8.15 cm to 8.15 cm, measured from the middle point along the fin, with 1.63 cm increments. 1.63 cm was equivalent to 5 percent of the robotic fin length. In Figure 4.6 (A-B), the measured forces are shown as a function of nodal shift. Results reveal that the thrust varies linearly as a function of nodal shift. Figure 4.6 (A-B) also reveal that the nodal shift gain increases as the temporal

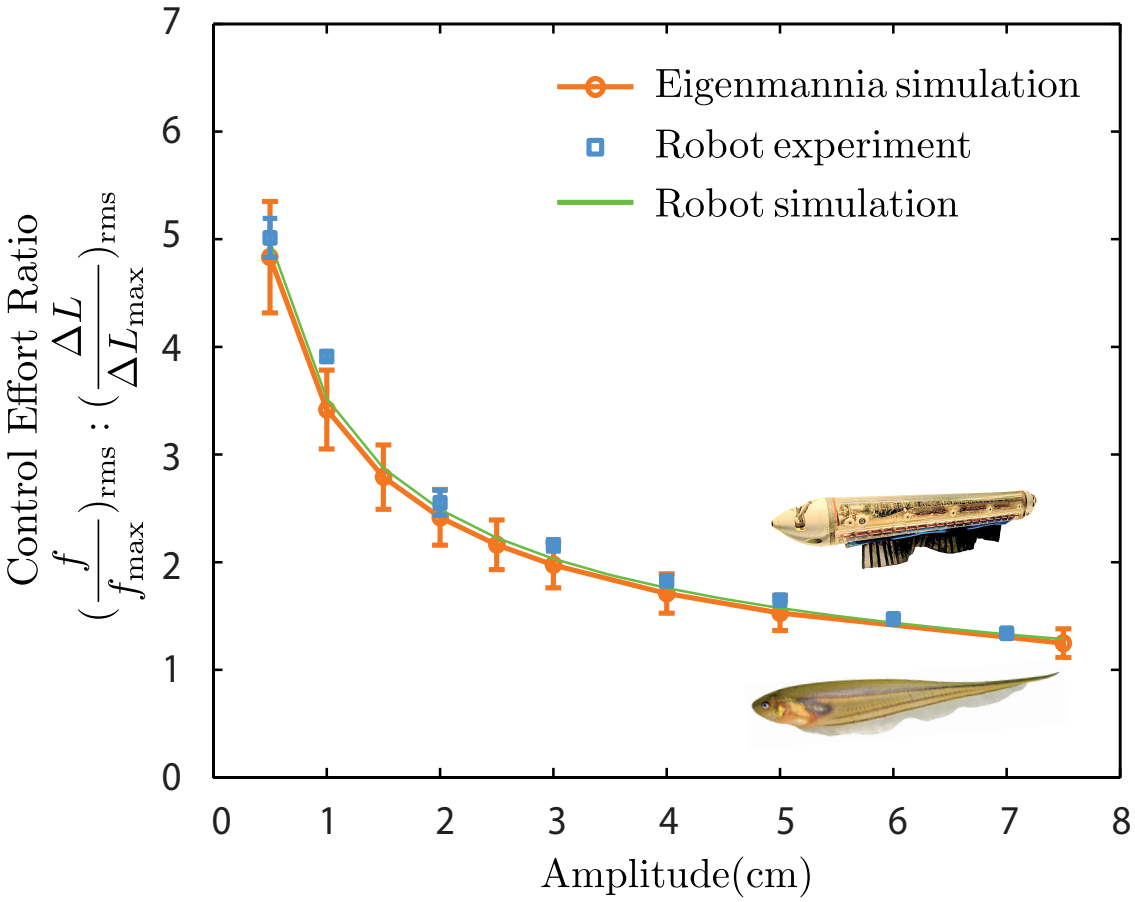

Figure 4.5: The ratio of the root-mean-square (RMS) of the normalized commanded control signals $((\frac{f}{f_{\max}})_{\text{rms}} : (\frac{\Delta L}{\Delta L_{\max}})_{\text{rms}})$ depends on the reference trajectory amplitude. The model predicts that this RMS ratio tends to infinity as the reference amplitude, A, goes to zero, strongly favoring counter-propagating waves when the goal is stable hovering ($A \approx 0$). Predicted and measured ratios for the robot closely match each other. Predicted ratios for *Eigenmannia* are based on traveling wave kinematics obtained during hovering ($U = 0$ cm/s). Uncertainty bars represent variability in kinematics of different subjects.

frequencies of counter-propagating waves and the amplitude of angular deflection of fin rays increase respectively. Nodal shift gains, corresponding to the results shown in Figure 4.6(A), as a function of temporal frequency (f) are depicted in Figure 4.6 (C). Nodal shift gain, κ, is increasing roughly quadratically as a function of frequency, f.

Finally, using the kinematics measured corresponding to hovering in biological experiments, nodal shift gain was estimated for *Eigenmannia*. In each set of simulations, the kinematic and morphological parameters of the model were set match the kinematics measured with no ambient flow trial ($U = 0$) in the biological experiments. Nodal position was varied from -10 mm to 10 mm. Simulation results for three replicates of a representative fish are depicted in Figure 4.6 (D). Forces generated by the head wave, tail wave, and the net thrust forces are shown for three replicates. Each color represents the result for one replicate (set) of data. The results reveal that the force generated by the two waves increases linearly as a function of nodal position. Simulation results for four other individual fish are similar to the results shown in Figure 4.6 (D). The nodal shift gain, κ, was 0.0209 N/m (std = 0.0084 N/m) over all replicates of data.

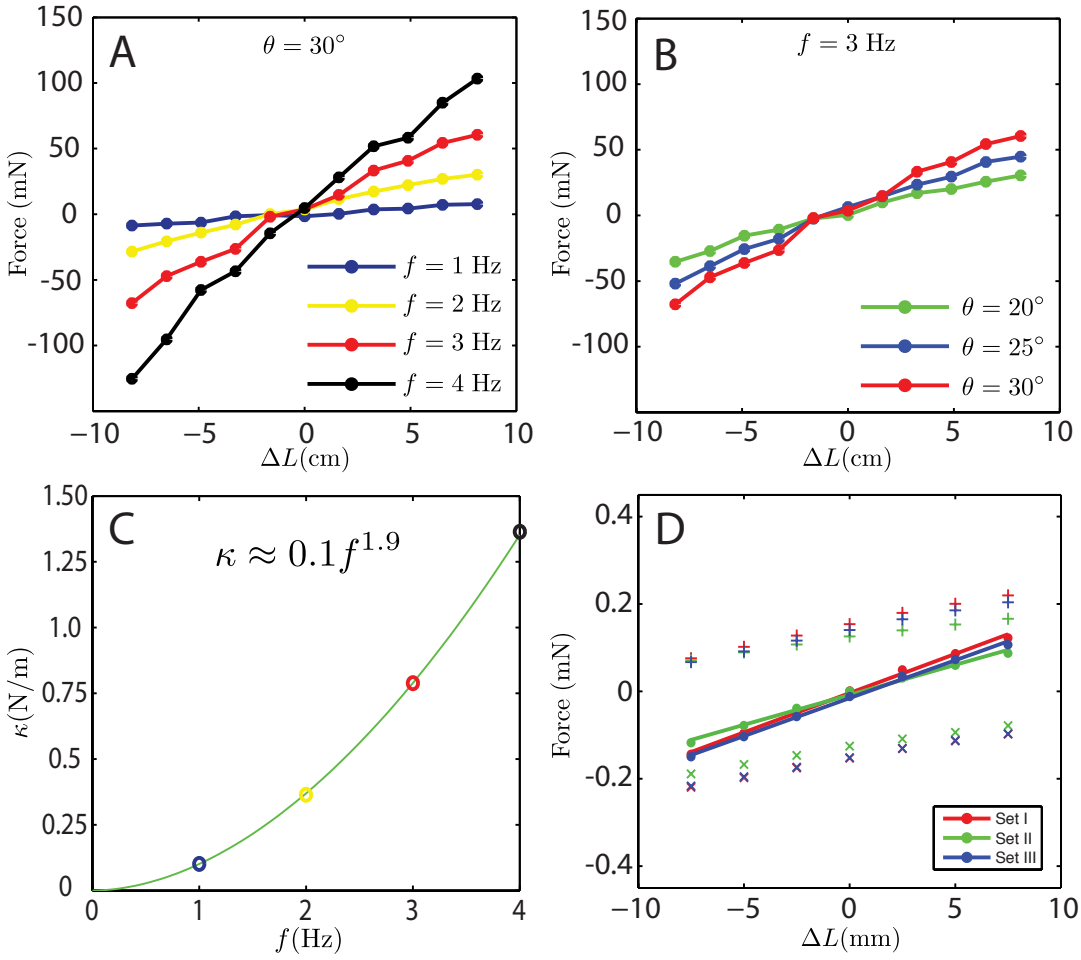

Figure 4.6: Force measurements from the robotic setup (nodal point shift gain): (**A**) For a constant angular amplitude ($\theta = 30°$), forces generated by robotic fin is shown for different frequencies. (**B**) For a constant frequency ($f = 3$ (Hz)), forces generated by robotic fin is shown for different angular amplitudes. (**C**) *Nodal shift gain* computed from a linear fit to the results shown in panel (A) are depicted as a function of frequency. κ varies nonlinearly as a function of f. Computational results: (**D**) Measured kinematics of Fish 4 from three replicates of the data during hovering (no ambient flow) are used as inputs for the computational model. Computed forces as a function of nodal shift (ΔL) are shown. Three color (red, green, and blue) correspond to three replicates (sets) of data. Forces generated by the head wave are shown with (+), forces generated by the tail wave are shown with ($\times$) and the net force produced by the two waves are shown with circles.

Table 4.2: Fin kinematic parameters for force measurement in the second robotic experiment.

Experimental set	f (Hz)	θ (deg)	# waves (L_{fin} / λ)
Set 1	1, 2, 3, 4	20	4
Set 2	3	20, 25, 30	4

4.4.3 Damping constant in robot and *Eigenmannia*

Kinematic parameters used for the robotic experiment and simulation were the same as those shown in Table 4.1. The nodal point was held at the center of the robotic fin ($\Delta L = 0$), thus the lengths of the two counter-propagating waves were equal. Ambient flow speed was varied from 0 to 10 cm/s with 0.5 cm/s increments. Only the robotic fin was submerged in this experiment. The measured forces as a function of steady-state ambient flow are shown in Figure 4.7 (A-B). From Figure 4.7 (A-B) it can also be observed that the damping constant increases as the temporal frequencies of counter-propagating waves and the amplitude of angular deflection of fin rays increase. Damping constants, corresponding to the results shown in Figure 4.7 (A), as a function of temporal frequency (f) are depicted in Figure 4.7 (C). Damping constant, β, is increasing linearly as a function of temporal frequency, f.

Finally, the damping constant was calculated for the *Eigenmannia*. Similar to the simulations explained in the previous section, in each set of simulations, kinematic

parameters of the model were set to the kinematics captured during zero flow speed ($U = 0$) in the biological experiment. The nodal position remained fixed at $\Delta L = 0$. The steady-state flow speed was varied from -5 cm/s to 5 cm/s. Predicted forces generated by each wave and the net force are shown in Figure 4.7 (D), and as shown, the damping forces increase linearly as a function of nodal position. Simulation results for four other individual fish are similar to the results shown in Figure 4.7 (D). The damping constant, β, was 0.0053 N.s/m (std = 0.0019 N.s/m) over all replicates.

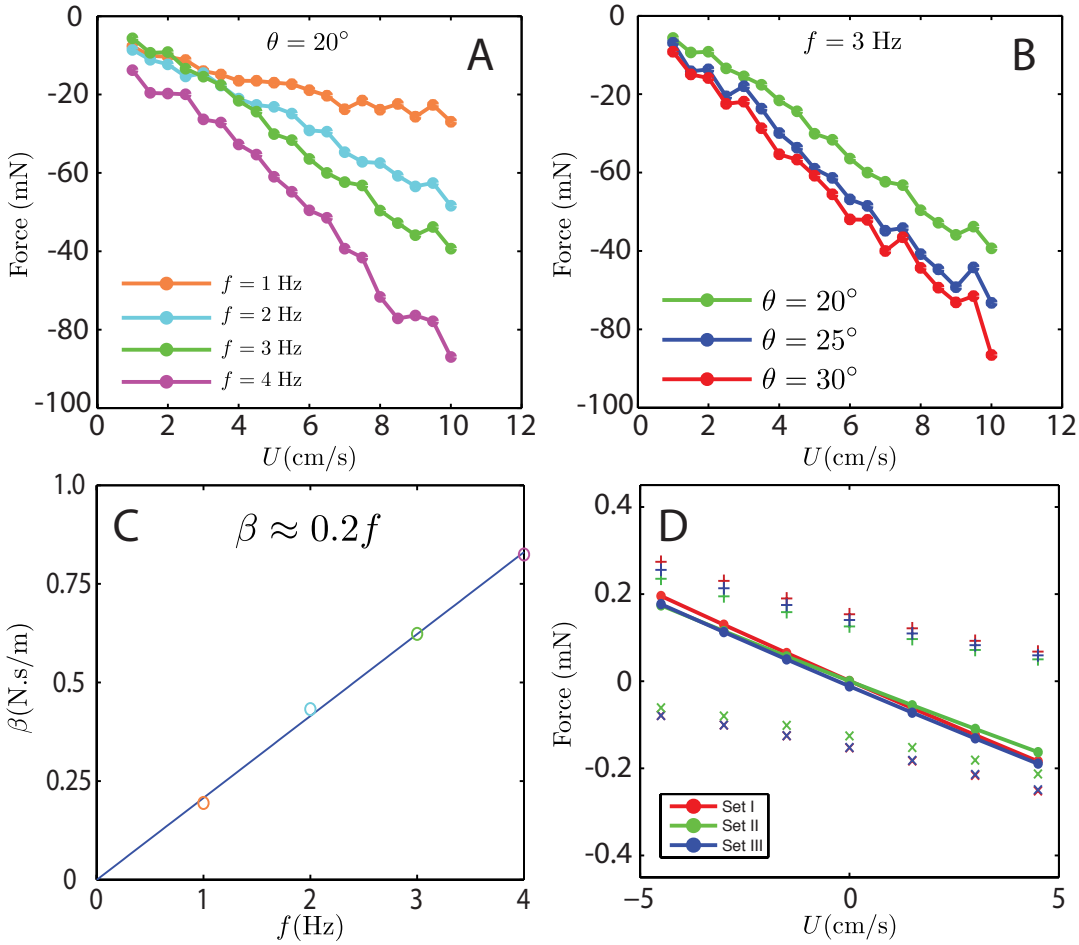

Figure 4.7: Force measurements from the robotic setup (damping constant): (**A**) For a constant angular amplitude ($\theta = 20°$), forces acting on the robotic fin are shown for different frequencies. (**B**) For a constant frequency ($f = 3$ (Hz)), forces acting on the robotic fin is shown for different angular amplitudes. (**C**) *Damping constant* computed from a linear fit to the results shown in panel (A) are depicted as a function of frequency. β varies linearly as a function of f. Computational results: (**D**) Measured kinematics of Fish 4 from three replicates of the data during hovering (no ambient flow) are used as inputs for the computational model. Computed forces over the ribbon fin are shown as a function of steady state flow speed (U). Three color (red, green and blue) correspond to three replicates (sets) of data. Forces generated by the head wave are shown with (+), forces generated by the tail wave are shown with ($\times$) and the net force produced by the two waves are shown with circles.

Chapter 5

Discussion

A key insight of the Wright brothers was that an aircraft must be both sufficiently stable to maintain its flight path and simultaneously maneuverable enough to permit its control [64, Ch. 1]. How animals manage this seemingly inescapable tradeoff [65,66] is an open question, especially since many swimming and flying animals appear to use locomotor strategies that are stable and yet facilitate the control of extraordinary maneuvers [26,38]. One possibility is that highly maneuverable animals are passively unstable, and stability is achieved solely via active feedback control using the nervous system [67].

By adopting a locomotor strategy that relies on the generation of antagonistic forces rather than a seemingly simpler strategy of moving the fin in either one direction or the other, the glass knifefish achieves a dramatic improvement in maneuverability, especially for small movements. This improvement in maneuverability is concurrent

with, as shown in Chapter 4, a significant increase in damping that enhances passive stability, although perhaps not without some energetic cost (see section 5.1). The fish could, in principle, actively stabilize itself using feedback control of either the nodal point (counter-propagating waves) or the frequency (single traveling wave). However, counter-propagating waves offer two advantages: they passively reject perturbations (increased passive stability) while also requiring substantially lower control effort (increased maneuverability). Therefore, antagonistic forces eliminate the trade-off between passive stability and maneuverability.

This strategy, which was discovered in measurements of the weakly electric fish *Eigenmannia* and tested using a biomimetic robot and a computational task-level model, may confer the same benefit in other animals that use antagonistic forces for locomotor control. Small terrestrial animals with a sprawled biomechanical posture appear to generate large lateral forces during forward running which have been postulated to enhance stability and maneuverability [35], although it remains unclear how such forces scale with body size. A mathematical model of high frequency flapping flight suggests that the antagonistic forces generated by the opposing movements of wings may similarly increase both maneuverability and stability [26]. It is interesting to note that high-frequency flapping fliers necessarily generate mutually opposing forces—they cannot readily turn these forces off during hovering. *Eigenmannia*, by contrast, are ideal for studying the role of mutually opposing forces because in these fish such forces result from a neural strategy not a biomechanical constraint.

Mounting evidence suggests that the passive design of animal morphology facilitates control, thereby reducing the number of parameters that must be managed by the nervous system [68, 69]. Here we describe a dynamical system that facilitates control by incorporating a similar design principle. Counter-propagating waves, which paradoxically appear to be a more complex behavioral strategy than the generation of simpler uni-directional waves, nevertheless simplify locomotor control. First, this strategy enhances stability and maneuverability as we have shown. Second, the modulation of the speed and direction of a single traveling wave requires the simultaneous (and instantaneous) coordination across a distributed network of spinal circuits whereas the modulation of the nodal point of two ongoing counter-propagating waves permits control via the coordination of a small number of these segmental circuits. How this motor coordination is achieved in the animal remains an open and interesting question [70]. Nevertheless, these data suggest that the dynamic design of animal morphology and their attendant neural systems are tuned [11,35,71,72] for simplified task-level control.

5.1 Energetic Cost

5.1.1 Mechanical energy during tracking and hovering

In the limit, as the tracking amplitude goes to zero (and assuming no disturbances), the mechanical work done by a single traveling wave is zero, but even for perfect hovering, counter-propagating waves are continuously doing work on the surrounding fluid. It is natural to ask whether it remains costly to use counter-propagating waves during tracking behavior.

To examine this question, we estimated the worst case mechanical energy required for a single traveling wave for the largest amplitude tracking motion compared to the energy required for simple hovering using counter-propagating waves as a conservative measure of how much more it costs the animal to use this strategy.

The instantaneous power from each infinitesimal element is given by $dP = -d\vec{\mathbf{F}} \cdot \vec{\mathbf{u}}$, where $\vec{\mathbf{u}}$ is the instantaneous velocity of the element relative to the fluid and $-d\vec{\mathbf{F}}$ is the force applied by each infinitesimal element of the fin to the fluid. Total power was estimated by integrating the dP over the fin. Mechanical work over each cycle was then estimated by integrating $P(t)$ over one period (5 s) of the tracking task (Figure 4.4).

5.1.2 Mechanical energy cost of counter-propagating waves

We estimated mechanical cost of counter-propagating waves during hovering and compared this to the cost during high-amplitude tracking using a single traveling wave.

For the fish, hovering with counter-propagating waves requires 1.1 mJ of mechanical work in 5 seconds while high-amplitude tracking using a single traveling wave requires 0.35 mJ. Likewise, for the robot, hovering with counter-propagating waves requires 1.7 J of mechanical work in 5 seconds while high-amplitude tracking using a single traveling wave requires 0.4 J. That is, for our setting, the mechanical energetic cost of counter-propagating waves is at least three times that of single traveling waves.

The mechanical energy required by each strategy is a factor which contributes to metabolic cost, but we cannot conclude that the differences in metabolic cost are commensurate to those in mechanical energy. Moreover, it is unknown whether the metabolic expenditure for either strategy is significant with respect to the metabolic budget.

5.2 A Few Loose Ends

- *What happens to the tradeoff when counter-propagating waves are no longer present?*

This study reveals that Eigenmannia can escape the trade-off between maneuverability and stability at low swimming speed by employing the counter-propagating waves strategy. Beyond 12 cm/s (or more precisely beyond the swimming speed at which nodal point disappears) the drag on the fish may provide a substantial enough opposing force to the single traveling waves thrust so as to yield a similar passively stabilizing drag force, but without further study this cannot be verified. It is likely that the fishs ability to rapidly increase/decrease swimming speed saturates at high speed, but this is also not known. In this case, the mutually opposing lateral forces e.g. created by the pectoral fins, may provide the fish with enhanced turning ability, and we hope that this study will inspire this sort of analysis in future studies.

- *Relation between the ratio of the RMS of the commanded control signals and ribbon-fin oscillatory frequency:*

  If the fish adopted a higher frequency (with all other parameters constant) for each of its two inward-counter-propagating waves, the nodal shift gain (κ) would be higher (see Figure 4.2(B)). This would further amplify the advantage of counter-propagating waves in terms of maneuverability for low amplitude tracking tasks, i.e. the RMS ratio curve (similar to the curve shown in Figure 4.5 but without normalizing the commanded control signals: f_{rms}:ΔL_{rms}) would be "higher" across all tracking amplitudes (Figure 5.1).

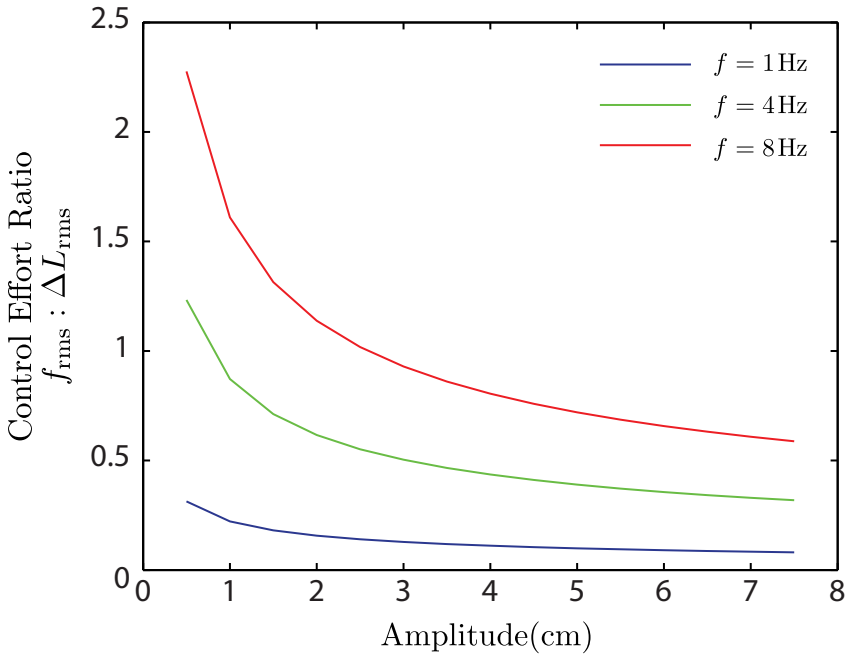

Figure 5.1: The ratio of the root-mean-square (RMS) of the commanded control signals increases at higher ribbon-fin oscillatory frequency.

Obviously, running the counter propagating waves at higher temporal frequencies would require more energy consumption during hovering but provides the fish with higher maneuverability. In other words larger nodal shift gain (κ) and larger damping constant (β) are achieved at higher energetic costs.

- *Relation between the ratio of the RMS of the commanded control signals and ribbon-fin spatial frequency:*

  How control ratio may be affected by spatial wavelength, λ, of traveling waves? It has been shown that the thrust generated by a single traveling wave varies as a function of spatial frequency [21], and Figure 5(B) in [24]). Specifically, the maximum generated thrust by a single traveling wave occurs at around

$L/\lambda = 2$ to 3. We calculated how the nodal-shift gain varies as a function of wavelength and also found that it reaches a local maximum at around $L/\lambda = 2$ to 3 (Figure 5.2). As a result, the RMS ratio of the commanded control signals would also vary as a function of L/λ as shown in Figure 5.3.

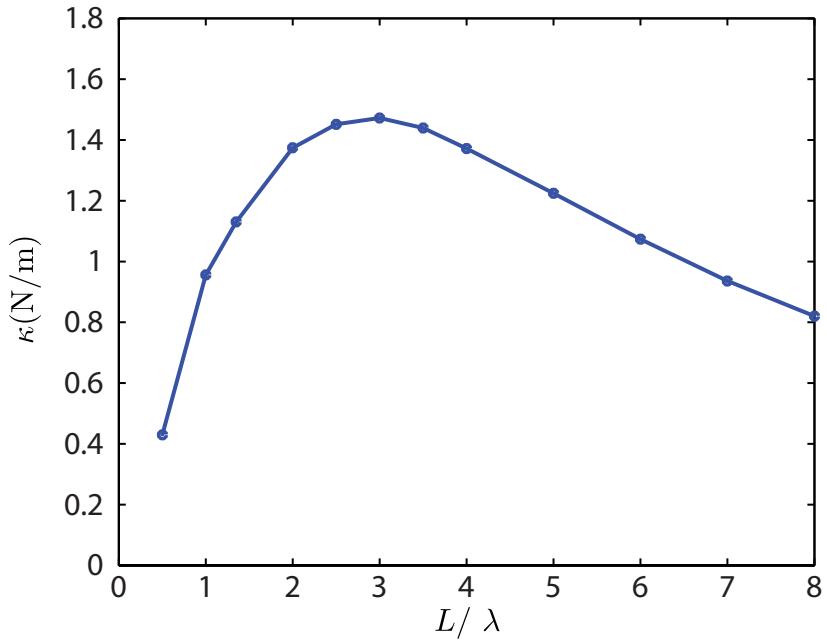

Figure 5.2: Nodal shift gain, κ, as a function of L/λ (L: fin length, λ: wavelength).

5.3 The Role of Mechanics in Decoding Sensory Systems—Revisited

The mechanics of locomotion—how motor signals are transformed into movement—dictates the control problem confronted by the nervous system [2, 4, 11, 35, 73]. The

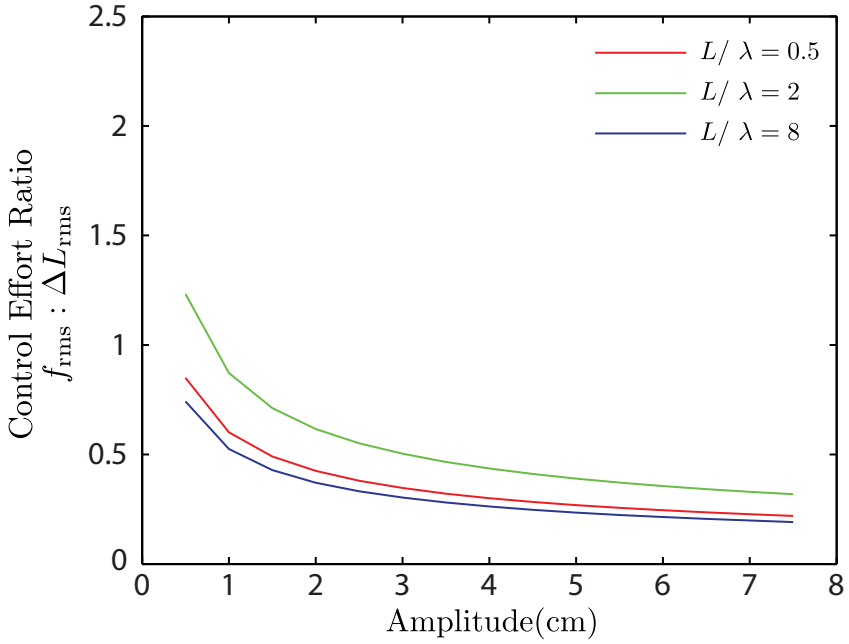

Figure 5.3: The ratio of the root-mean-square (RMS) of the commanded control signals curves as a function L/λ (L: fin length, λ: wavelength).

glass knifefish provides an excellent system for examining the interplay between mechanics and sensing. These fish perform a behavior—refuge tracking [16]—that can be modeled as a single degree of freedom behavior, greatly facilitating neuromechanical control systems modeling.

Cowan and Fortune [11] developed a control-theoretic framework for making neural control predictions given two ingredients. First, one measures the closed-loop tracking performance of the animal. Second, one uses a model of the locomotor dynamics—together with the measured closed-loop transfer function—to predict how the sensory information is processed by the nervous system (the neural controller). This process turned out to be surprisingly sensitive to the assumptions about the

locomotor dynamics. Our new analysis sheds light on this sensitivity.

5.3.1 A feedback control model of the glass knifefish

The glass knifefish, *Eigenmannia virescens*, is like an "aquatic hummingbird": it hovers in place with extraordinary precision, making rapid and nuanced adjustments to its position in response to moving stimuli. How does the nervous system process sensory information to control this behavior?

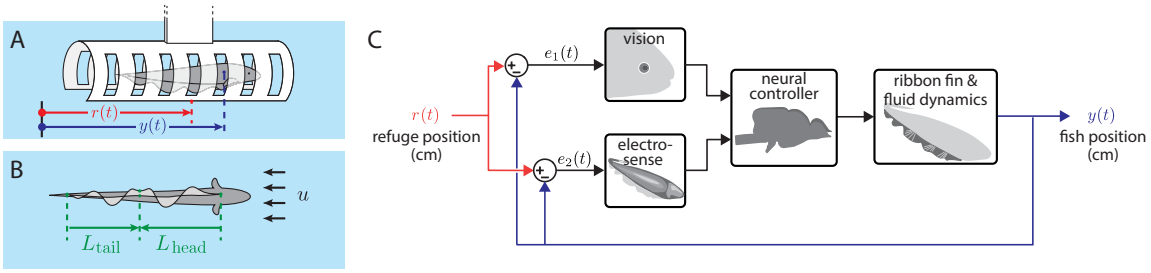

Figure 5.4: (**A**) Glass knifefish *Eigenmannia virescens* tracking a moving shuttle. (**B**) Bottom view: Glass knifefish modulating its fore-aft position by counter-propagating waves. (**C**) A block diagram depicting the knifefish's reference-tracking behavior.

It is reasonable to assume that, as the animal is swimming in the refuge, the source of sensory feedback is the *error signal*: the position of the refuge relative to the position of the fish [50]. If the fish swims forward, or the tube is shifted backward, it creates visually and electrically identical stimuli. In this way, the fish's motor output—its fore-aft position—plays an equal (but opposite) role in the generation of a sensory stimulus as does the movement of the tube itself. If we let $x(t)$ denote the

position of the fish, and $r(t)$ denote the position of the refuge, we see that the nervous system must process the difference, or *error*:

$$e(t) = r(t) - x(t). \tag{5.1}$$

To complete the closed-loop diagram in Figure 5.4, this error signal is processed by the neural controller to produce a motor command, $u(t)$, which is in turn processed by the locomotor mechanics (the so-called "plant") to produce the position, $x(t)$. In transfer function notation, we have

$$E(s) = R(s) - X(s), \tag{5.2}$$

$$U(s) = C(s)E(s), \tag{5.3}$$

and

$$Y(s) = G(s)U(s). \tag{5.4}$$

A rearrangement of these equations yields the closed-loop transfer function

$$\frac{Y(s)}{R(s)} = \frac{G(s)C(s)}{1 + G(s)C(s)} =: H(s) \tag{5.5}$$

These equations correspond to the simple closed-loop topology in Figure 5.4. It is important to recognize what is *missing* from this topological model. All of the feedback is modeled as relative to the reference. This might not be the case with vestibular feedback, for example, which would best modeled relative to an inertial reference frame.

Using single-sine frequency response data, [11] fit a closed-loop transfer function model of the following form:

$$H(s) = \frac{0.73}{0.023s^2 + 0.17s + 1}. \tag{5.6}$$

where complex frequency s is measured in rad/sec. The transfer function is dimensionless (cm/cm) because $H(s)$ represents the frequency-dependent ratio of fish-to-shuttle movement.

The central question from a neural control perspective is, how is the error signal processed in order to modulate motor output? In our modeling paradigm, this is captured by the transfer function $C(s)$. Unfortunately, $H(s)$, does not directly reveal this, because there is another unknown in Equation (5.5), namely the mechanical plant, $G(s)$. Given the input–output response $H(s)$, how sensitive is our prediction of the neural controller transfer function, $C(s)$, to different candidate structures for the mechanical plant, $G(s)$? To answer this question, [11] opened the loop algebraically:

$$C(s) = \frac{H(s)}{(1 - H(s))G(s)} \tag{5.7}$$

They then posited two possible transfer functions for the locomotor dynamics. The first was of the general form[1]

$$G_1(s) = \frac{1}{bs}, \tag{5.8}$$

where b is a damping coefficient. They referred to this model as the "kinematic plant model", because the control signal $u(t)$ is modeled to be proportional to velocity. In

[1]In their original formulation, they neglected to include the factor, b; including this parameter clarifies the model and rationalizes the units.

the time domain, this would be written as follows:

$$b\dot{x}(t) = u(t). \tag{5.9}$$

Such a model would be reasonable in a context where damping forces dominate.

On the other hand, when damping is negligible (as Cowan and Fortune suggested in their paper), one arrives at the "mechanics" transfer function

$$G_2(s) = \frac{1}{ms^2}, \tag{5.10}$$

where m is the mass. Here, $G_2(s)$ represents the following differential equation

$$m\ddot{x}(t) = u(t). \tag{5.11}$$

In this setting, $u(t)$ is proportional to acceleration.

Surprisingly, Cowan and Fortune found these two transfer functions give categorically opposite predictions regarding the transfer function of the controller. Specifically, if one assumes a kinematic plant model, of the form of Equation (5.8), one predicts that the nervous system is implementing a low-pass filter for control. This prediction is referred to as $C_1(s)$ in Figure 5.5(B-C). On the other hand if one assumes a mechanical plant model, Equation (5.10), one predicts that the nervous system is implementing a *high-pass* filter, $C_2(s)$!

Importantly, while it was clear from the data and analysis that the locomotor mechanics were critical to making a good neural prediction, Cowan and Fortune did not have a validated plant model. In other words, the transfer function models they posited for the plant, $G_1(s)$ and $G_2(s)$, were based on physical reasoning, not data.

5.3.2 A validated plant model for the glass knifefish

In these fish, fore–aft thrust force is primarily produced by undulatory motion of a long ribbon-fin along the body. Glass knifefish routinely partition the ribbon-fin into two counter-propagating waves: one traveling from head to tail (head wave) and one traveling from tail to head (tail wave) [48, 59]. These two waves meet each other at a point somewhere in the middle of the fin; we term this point "nodal point". During hovering, the two waves generate opposing forces that cancel each other out.

Observation from biological experiments revealed that glass knifefish modulates the net fore–aft force, primarily, by moving the nodal point [48]. As shown in Figure 5.5(A), nodal point is somewhere in the middle of the fin during hovering. While the fish swims forward and backward, the nodal point moves caudally and rostrally respectively. Computational task-level model and force measurements from experiments with a biomimetic robot (see Chapter 4) revealed that the net fore–aft thrust force varies linearly as a function of nodal point position. This is in contrast to the seemingly simpler strategy of producing a single traveling wave in which the thrust force exhibits a nonlinear profile as function of the speed of the traveling wave [7, 24]. Simulations validated by experimental results with the robot showed that the use of counter-propagating waves significantly improves the fore–aft maneuverability (by decreasing the control effort), and concurrently enhances the passive stability (stabi-

lization without active feedback control) by providing a damping-like force to reject perturbations, thus simplifying control.

In addition to helping elucidate the mechanism by which these fish overcome the trade-off between stability and maneuverability, the task-level "mechanical plant" developed in [7] can be reevaluated in the context of the closed-loop model, so that a more accurate neural control prediction can be made using Equation (5.7). Specifically, Sefati et al. found that

$$G_3(s) = \frac{k}{ms^2 + bs}. \tag{5.12}$$

Note that $G_1(s)$ and $G_2(s)$ used in [11] (see Equations (5.8) and (5.10), above) are limiting cases of $G_3(s)$: neglecting the inertia leads to $G_1(s)$ and neglecting the damping force leads to $G_2(s)$. This new model confirms Cowan and Fortune's hypothesis that sensory processing would be better approximated by a high-pass filter, $C_3(s)$. However, the new prediction is quite different in its details of the gain and phase relationships compared to the previous predictions, $C_1(s)$ and $C_2(s)$ (Figure 5.5(B-C)).

5.3.3 Inescapable sensitivity of the control prediction

Here, we revisit the question of the sensivity of the neural prediction. What is not clear is if Cowan and Fortune's finding—that the neural control prediction is sensitive to the plant—is general, or resulted from an idiosyncratic detail of the organism or

behavior at hand.

Given a fitted closed-loop transfer function $H(s)$, it is apparent that $C(s)$ depends inversely on $G(s)$. Specifically, note that frequency-dependent sensitivity of C as a function of G is given by

$$S(j\omega) = \frac{\partial C}{\partial G} = \left[\frac{H(j\omega)}{H(j\omega) - 1}\right] \frac{1}{G^2(j\omega)} \tag{5.13}$$

It is important to recognize that this sensitivity function is a frequency-dependent calculation. It depends on the gain and phase of both the measured closed-loop transfer function, $H(j\omega)$, as well as the plant model, $G(j\omega)$.

In general, engineered closed-loop control systems perform particularly well at low frequencies: they track slowly varying stimuli with great precision. This is also evident in many biological systems. For the fish, this means that if the shuttle were moved and then held still, the fish would ultimately recover and "catch up" with the refuge, achieving very low steady-state error.

In other words at low-to-intermediate frequencies, the fish performs excellent closed-loop tracking, ergo $H(s)$ has nearly unity gain and zero phase lag, and thus $|S(j\omega)|$, which depends on $1/|H(j\omega) - 1|$, will be large. At high frequencies, the picture is equally challenging: the sensitivity varies inversely with the square of the plant, and since mechanical systems tend to be low-pass, $|G(j\omega)| \to 0$ as $s = j\omega \to \infty$, and thus $|S(j\omega)|$ approaches infinity at high frequencies as well. By evaluating $\partial C/\partial H$, one can show that the control prediction is also quite sensitive to the fitted closed-loop transfer function.

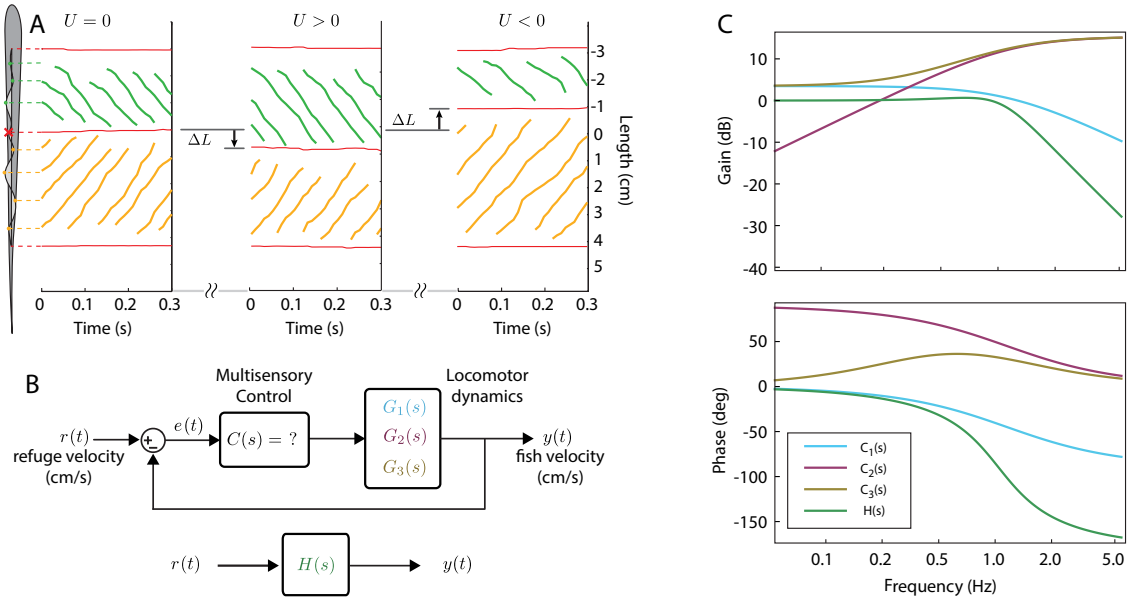

Figure 5.5: (**A**) *Eigenmannia virescens* routinely partitions its ribbon-fin into two counter-propagating waves that generate opposing forces. Both ends of the fin and the nodal point (red cross), all peaks and troughs of the head and tail waves (green circles and orange circles respectively) were tracked during station keeping at different swimming speeds. Nodal point shift, ΔL, from 0 cm/s flow speed (no ambient flow) to forward and backward swimming of a representative data set is shown. (**B**) Block diagram of the closed-loop refuge tracking behavior. Different candidate models for locomotion dynamics lead to different prediction of multi-sensory control. (**C and D**) Gain and phase relationships of closed-loop transfer function, $H(s)$, and predictions of multi-sensory control based on different plant models.

In summary, it is critical to carefully characterize the closed-loop dynamics as well as the plant model when trying to predict the control computations being performed by the nervous system.

Part II

Linear Dynamical Systems with Sparse Inputs (LDS-SI): Application in Surgical Gesture Segmentation and Classification

Chapter 6

Motivation

6.1 Data-driven System-theoretic Analysis of Complex Time-series Data

Animals and human produce extremely rich and complex behavior. But it is this richness that also makes the behavior decoding difficult. Despite this apparent complexity, many behaviors seem to result from comparatively simple, low-dimensional patterns of movement. This part of the dissertation aims to propose a new class of linear time-invariant dynamical systems to study complex biological and mechanical movements.

Linear dynamical systems are widely used to model time-series data, including dynamic textures [74], surgical video data [75], and human movement [76, 77]. For

instance, dynamic textures such as smoke, flame, etc. have been modeled as the output of a linear dynamic system with Gaussian noise (Auto Regressive Moving Average model) [74]. Simple human gaits such as walking, running, and jumping have been modeled as the output of a stationary ARMA model [76]. We argue that while stationarity may be a reasonable assumption for dynamic textures, most of biological and mechanical movements are categorically non-stationary.

On the other hand, sparse representation theory has a long history in signal processing community [78]. Sparsity embodies the notion that quite often very complex (and high-bandwidth) signals can be represented as a combination of surprisingly few basis vectors. In recent years, overcomplete representations have found widespread applications in neuroscience [79], machine learning [80], computer vision [81], data acquisition [82]. In signal processing, results from sparsity have led to the theory of compressive sensing (CS), enabling one to completely side-step the well-known Nyquist Sampling Theorem by reconstructing a signal from far fewer measurements than predicted by Nyquist sampling theorem. The idea was popularized by the invention of a "single pixel camera"—a concept developed by researchers at Rice University, in which they built a digital camera that takes a surprisingly small number of random one dimensional (i.e. single pixel) projections of the scene. Using these "one pixel images", the researchers demonstrated the successful recovery of a photographic image. This and other applications to medical image reconstruction, computer vision and machine learning have demonstrated the extraordinary applicability of the theory of

sparse representations to real-world problems and data.

What does sparsity have to do with dynamical systems in general, and decoding complex time-series data in particular? Interestingly, the recent results from sparse representation theory have not yet revolutionized linear systems theory. In principle, one could have thought of re-doing systems theory in the light of recent results on sparse representation by measuring complexity of a system using sparsity of the state (as opposed to low-rank). While some of these ideas have been explored in the context of observer design for linear systems [83–85] (see section 7.1 for more details), such approaches assume that the states, initial conditions, or innovations are sparse. In our view, this notion of sparsity for linear systems dramatically limits the class of systems that can be described. For example, if the state x_k is sparse, a state transition to $x_{k+1} = Ax_k$ need not give a sparse vector, unless A satisfies some very strong conditions.

In this part of the dissertation, we propose an alternative view to incorporate notions of sparsity into linear systems theory. In particular, we consider a class of LTI systems with unknown sparse *inputs*. The input at *each time step* is assumed to be high-dimensional, and sparse with respect to an overcomplete dictionary of inputs. Note that by changing the sparsity pattern of the input as a function of time, we can generate non-stationary inputs. Moreover, the overall behavior of the proposed system is nonlinear. In fact, linear dynamical system with switched inputs are a particular case of our model. In the context of biological and mechanical movements,

the dictionary of inputs may represent the dictionary of all possible simple behaviors. At each time step, only a few dictionary atoms are excited by the sparse input, orchestrating a specific behavior.

6.2 Roadmap and Contributions

In Chapter 7 we introduce the Linear Time-invariant Dynamical Systems with Sparse Inputs (LDS-SI). We then consider one of the fundamental problems of linear system theory: state estimation (initial condition recovery) with *unknown* inputs. We propose a convex optimization formulation to jointly recover the initial condition and the sequence of unknown, but sparse, inputs while supports and values of sparse inputs are allowed to vary arbitrarily as a function of time. We derive sufficient conditions for the perfect joint recovery. Simulation results are also presented.

In Chapter 8 we demonstrate the power of the LDS-SI framework in the analysis of complex movements during a surgical task. In particular we propose an algorithm for the segmentation and classification of surgical gestures. We end this chapter by discussions and directions for future research.

Chapter 7

LDS-SI: State estimation and input recovery

In this chapter we consider a new class of linear time-invariant dynamical systems for which the input signal is sparse, and address the fundamental problem of state estimation of such systems assuming the sparse input is also unknown. In section 7.1 we provide a summary of how the notion sparsity has been previously incorporated in the context of linear dynamical systems In section 7.2 we review compressive sensing and sparse recovery algorithms. In Section 7.3 the classical observability and state estimation problems of deterministic LTI systems are briefly discussed. In Section 7.4 we formally introduce a special class of discrete-time LTI systems with sparse inputs at each time step. A convex optimization formulation for the joint recovery of the initial condition and unknown sparse inputs is proposed. We derive the sufficient

conditions for joint recovery of sparse inputs and initial state. Simulation results and discussions are included in Section 7.5, followed by a conclusion in Section 7.6.

7.1 Linear Dynamical Systems and Sparsity

Several attempts have been made to incorporate sparsity in the context of linear dynamical systems. Most attempts can be categorized depending on how or where sparsity is imposed, e.g. sparse parameters, sparse states, or sparse inputs. In this section we provide a review of these trends.

7.1.1 LDSs with sparse parameters and sparsity in system identification

System identification and model selection applications date back to the early days of the development of sparsity-based methods and some of the most widely used methods are rooted in that literature. In this setting, sparsity in the parameter space is assumed and exploited for system identification and model order reduction (i.e., approximating a given complex system with a *simple* system of lowest order). Often sparsity inducing norms or constraints are used in constraints or in regularization in an optimization framework. One of the earliest example includes [86, 87] where system

identification and model *order* reduction are posed as the problem of choosing an efficient representation (*fewest* number of coefficients) of the frequency response of an LDS in the so-called rational wavelet basis. Another example is [88] where the problem of model order reduction is posed as a lowest *rank* matrix approximation using the so-called *nuclear* norm as a surrogate for rank. Various extensions to this approach have been proposed e.g., [89–92]. Some other examples in this category include [93–96]. In [94] it is shown that system identification of LTI ARX models and input delay estimation of "sparse systems" are possible from few observations with appropriate sparsity inducing regularization. Topology identification of large-scale sparsely connected dynamical networks can also be facilitated with sparsity inducing regularizer [95, 96].

Sparsity based methods also have been applied to system identification in other classes of dynamical systems e.g., time-varying, hybrid, switching, and nonlinear systems. In particular, identification of hybrid and switched LDSs for applications in video segmentation has received considerable attention [97, 98]. [99] presents an l_1 regularization formulation over the time-varying parameters of the model for the problem of segmentation of time-varying ARX-models. In [100] system identification for certain classes of nonlinear systems under time-variation with l_1 regularization is studied. In addition, ℓ_1 regularization over the time-varying parameters of the model has been used for the problem of segmenting time-varying ARX models [99]. In [100] system identification for certain classes of nonlinear systems under time-variation

with ℓ_1 regularization is studied. In a rather conceptually different framework [101] forms a dictionary of LDS models given a training set of time series data with an application to dynamic texture recognition, where given a new sequence of data they aim to find an LDS representation which is sparse in terms of the dictionary of LDSs. A complication with such approaches is the nonlinear structure of the relevant space of LDSs, which is difficult to take into account [102].

7.1.2 LDSs with sparse states

Recently, a number of filtering and smoothing algorithms have been proposed for recovering time-varying sparse signals whose temporal evolution ca be modeled by an LDS [83–85, 103–105]. These algorithms are mostly targeted at compressive sensing of time-varying sparse signals and they usually require limiting assumptions in the temporal evolution model such as slow changing sparsity patterns of the signal. For instance, Vaswani [83] proposed a modified Kalman filter algorithm for the estimation of time-varying spatially sparse signals with slow changes in the sparsity patterns in "real time," with applications in functional MRI. Asif et al. [104] presents a homotopy algorithm to dynamically recover a sparse signal that changes slightly between measurements. Angelosante et al. [103] regularizes the Kalman smoother with sparsity-inducing l_1 norm for tracking applications. In [85] an l_1 regularized optimization is proposed to incorporate the sparsity constraints for the evolving sparse signals as well as the error in the signal prediction (innovation). Ziniel et al. [105] presents a

probabilistic signal model for time-varying sparse signals with slowly varying support set. Smooth variations of active coefficient amplitudes are modeled as Gauss-Markov process. Tracking and smoothing of such signals are carried out through belief propagation. In a different approach, Wakin et al. [84] studied the observability of linear systems with sparse high-dimensional initial state and randomized compressive measurements.

Overall, while enforcing sparsity on the states results in promising state estimation algorithms of sparse time-varying signals, we believe this notion of sparsity is fundamentally limited. For example, if we consider an LDS $x_{k+1} = Ax_k$, then if x_k is sparse, Ax_k need not to be sparse except for a special matrix (A) (e.g., a permutation).

7.1.3 LDSs with sparse inputs

Sparse input models have mostly been used for modeling spike trains, i.e. signals that are sparse in time. For instance, blind deconvolution [106] with ℓ_1 regularization for recovering spike trains has long been applied in the context of seismic signal processing [107, 108]. However, such approaches and more recent ones have been limited to single-input single-output systems, usually with finite impulse response [107–110]. More recently, in [111] a similar approach has been proposed for modeling multivariate time series of human actions as the output of an LDS driven by a one dimensional spike train. Moreover, in an alternating minimization framework both the train of input spikes and the LDS model are learnt. This approach is an example of blind sys-

tem identification or deconvolution. The main limitation of this approach is that the input class is impoverished enough and cannot generate complex dynamic behavior (whereas in our proposed linear time-invariant dynamical system with sparse inputs much more complex dynamics could be modeled by using a rich input dictionary). Some other works (e.g. [85, 112]) consider sparse inputs as noise rather stimulus, i.e., an undesired signal to be suppressed. To the best of our knowledge, imposing sparsity at the input of MIMO LDSs while including and learning a dictionary of basis for the input (thus allowing a rich class of inputs) is a novel approach and has not appeared in the literature so far.

7.2 Compressive Sensing and Sparse Recovery

Compressive sensing (CS) and sparse signal recovery gained significant attention in recent years, for good reason: the theory of compressive sensing states that some sparse signals can be exactly and robustly recovered from an underdetermined and possibly noisy set of measurements. Consider a real-valued signal $x \in \mathbb{R}^m$. The signal x can be represented in a basis consisting of m vectors in $\mathbb{R}^m$: $\Psi = [\psi_1, \psi_2, \ldots, \psi_m] \in \mathbb{R}^{m \times m}$:

$$x = \sum_{i=1}^{m} \psi_i s_i = \Psi s \tag{7.1}$$

where $s = [s_1, s_2, \ldots, s_m]^T$. Here, s represents the same signal relative to the basis defined by the columns of Ψ. Sparse representation theory is predicated on the idea that most real signals admit a sparse representation with respect to some properly chosen basis. More specifically, given a signal $x \in \mathbb{R}^m$, a basis (or dictionary) Ψ, x represented as $x = \Psi s$, where the signal s is S-sparse, namely it has S non-zero elements, $\|s\|_0 \leq S \ll m$. To fix notation, $\text{supp}(s) = \{i | s_i \neq 0\}$ is the set of indices corresponding to non-zero entries, $\|s\|_0 = |\text{supp}(s)|$. Consider a linear measurement of the entries of the signal x: $y = \phi^T x$. This measurement can be viewed as the inner product of the measurement vector ϕ and the signal x: $y = \langle \phi, x \rangle$. Assuming we have p measurement vectors ($\{\phi_i\}_{i=1}^p, p < m$), the underdetermined *sensing matrix*, Φ, is constructed by taking the ϕ_i^T's as its rows:

$$y = \Phi x = \Phi \Psi s = \Theta s \tag{7.2}$$

Equation 7.2 is an underdetermined system of equations and in general there exists infinite number of solutions but under some remarkably general conditions on the sensing matrix Φ, it turns out that one can recover s exactly from far fewer samples ($p \ll m$) of the signal than predicted by the "Nyquist sampling theorem." Concretely, the sparse solution to the underdetermined system of equations $y = \Theta s$ can be found by solving the following optimization problem:

$$P_0 : \min_{s \in \mathbb{R}^m} \|s\|_0 \quad \text{subject to} \quad y = \Theta s. \tag{7.3}$$

In general this is a non-convex optimization and an NP hard combinatorial problem.

There are two main approaches to solve such a non-convex optimization problem: 1) heuristic greedy algorithms such as *Orthogonal Matching Pursuit* (OMP) [87], 2) convex relaxation methods such as *Basis Pursuit* (BP) [78]. In nutshell, greedy algorithms such as OMP attempt to directly solve the P_0 problem by finding the columns of Θ matrix (known as the atoms of the dictionary in sparse overcomplete representation) that have the highest correlation with the measurement. The second category of approaches (BP algorithm [78]):

$$P_1 : \min_{s \in \mathbb{R}^m} \|s\|_1 \quad \text{subject to} \quad y = \Theta s. \tag{7.4}$$

Three main categories of theoretical guarantees establish that under appropriate assumptions, the convex P_1 problem is equivalent to the non-convex P_0 problem: restricted isometry property [113, 114]; exact recovery condition [115]; and mutual coherence [116]. Theoretical guarantees are provided for stable and exact recovery in the presence of noise [116, 117]

The standard techniques in CS hold for signals that are sparse in the standard basis ($\Psi = I$) or in some proper orthonormal basis. Rauhut et al. [118] showed that similar techniques could be applied to recover signals that are sparse with respect to a (possibly) overcomplete dictionary. More recently, Candes et al. [119] provided theoretical guarantees adopted to overcomplete and redundant (coherent) dictionaries. In the following subsections two recovery conditions for stable and exact recovery of sparse signals are briefly introduced.

7.2.1 Restricted Isometry Property (RIP)

The restricted isometry property (RIP) serves as a measure of orthonormality of matrices when operating on sparse vectors. For all vectors with sparsity level S (represented by x_S), $\delta_S(\Theta)$ is the smallest constant that the following inequalities hold:

$$(1 - \delta_S)\|x_S\| \leq \|\Theta x_S\| \leq (1 + \delta_S)\|x_S\| \tag{7.5}$$

Several bounds on RIP constant have been established for stable and exact recovery of solutions to P_0 problem by solving the convex optimization problem, P_1 [113]. Computing the restricted isometry constant of a matrix is an NP-hard problem. It has been shown that random matrices with independently identically distributed (i.i.d.) Gaussian distributions satisfy the RIP [120] with high probability.

7.2.2 Mutual coherence

Mutual coherence of a matrix Θ is defined as the maximum absolute value of coherence between normalized columns:

$$\mu(\Theta) = \max_{i,j,i \neq j} \frac{\theta_i^T \theta_j}{\|\theta_i\| \|\theta_j\|} \tag{7.6}$$

In contrast to RIP constant, mutual coherence of a matrix can be easily computed. The S-sparse solution to the P_0 problem can be obtained by solving the convex problem P_1, if the coherence of matrix A satisfies the following [116]:

$$\mu(\Theta)(2S - 1) < 1 \tag{7.7}$$

In words, the angle between the normalized columns of the matrix A should be above some threshold. Both of these criteria discussed above are sufficient conditions for exact recovery of the sparse solution.

7.3 State Estimation for Deterministic LTI Systems

Linear time-invariant (LTI) systems are widely used to model various time-series data, including dynamic textures [74], surgical video data [75], and human movement [77]. Consider the general model of linear dynamical systems:

$$\begin{aligned} x_{k+1} &= Ax_k + \Psi u_k, \quad x_k \in \mathbb{R}^n, u_k \in \mathbb{R}^m \\ y_k &= Cx_k, \qquad\qquad y_k \in R^p. \end{aligned} \tag{7.8}$$

Here, $A \in \mathbb{R}^{n\times n}$ is called the dynamic matrix, $C \in \mathbb{R}^{p\times n}$ is the observation matrix, and u_k is the input signal.

For the linear system defined in (7.8), let

$$Y_N = [y_0^T, y_1^T, y_2^T, \ldots, y_{N-1}^T]^T \in \mathbb{R}^{Np} \tag{7.9}$$

denote all measurements up to time N. Likewise, let

$$U_N = [u_0^T, u_1^T, u_2^T, \ldots, u_{N-2}^T]^T \in \mathbb{R}^{(N-1)m} \tag{7.10}$$

denote the input sequence. One can easily show that

$$Y_N = \mathcal{O}_N x_0 + \Gamma_N U_N \tag{7.11}$$

where $\mathcal{O}_N$ is the observability matrix:

$$\mathcal{O}_N = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{N-1} \end{bmatrix} \tag{7.12}$$

and

$$\Gamma_N = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ C\Psi & 0 & \cdots & 0 \\ CA\Psi & C\Psi & \cdots & 0 \\ \vdots & \vdots & \ddots & \\ CA^{N-2}\Psi & CA^{N-3}\Psi & \cdots & C\Psi \end{bmatrix}. \tag{7.13}$$

State estimation is one the fundamental problems in linear dynamical systems. The state estimation problem typically boils down to recovering the sequence of states from inputs and outputs over time, i.e. $\{u_k, y_k\}_{k=0}^{N-1}$. Indeed a system is typically defined to be *observable* if one can recover x_0 given the measurements and input. In can be easily verified from (7.11) that in the cases where the sequence of inputs and outputs are known, the initial condition x_0 can be recovered if the rank of the observability matrix $\mathcal{O}_N$ is n.

What if the inputs are not known, as assumed in this chapter? Provided that there are fewer outputs than inputs, i.e. $p < m$, recovering the inputs is an ill-posed problem, in which case it is not clear if and how to reduce this to a standard state

estimation problem, which requires knowing the inputs to solve (7.11).

7.4 Linear Time-invariant Dynamical Systems with Sparse Inputs

Compressive sensing theory applies to the identification of sparse signals with a small number of measurements, while observer theory focuses on recovering system state (or initial conditions). How can these two ideas be combined to estimate the initial conditions and the input signals? Techniques developed in the field of compressed sensing and sparse recovery have been mostly applied to static linear problem. The framework we propose extends sparse representation theory to a dynamical context.

In this section, we introduce a new class of linear dynamical systems that the input u_k at *each time step* is sparse with respect to an overcomplete dictionary of inputs Ψ, namely, $\|u_k\|_0 \leq S$ ($S \ll m$). Formally linear time-invariant dynamical systems are systems of the form:

$$\begin{aligned} x_{k+1} &= Ax_k + \Psi u_k, \quad x_k \in \mathbb{R}^n, u_k \in \mathbb{R}^m, \|u_k\|_0 \leq S \\ y_k &= Cx_k, \qquad\qquad y_k \in R^p. \end{aligned} \tag{7.14}$$

Support and value are not constrained over time in our model. We let n and m be possibly large. Sparsity of the input at each time step means that at any given time only a few columns of the basis Ψ get excited, but in as few as m/S steps, all of the

columns could be excited, depending on how the support varies over time.

The overall behavior of LDS-SI systems is nonlinear. In fact, we can think of the proposed LTI systems with S-sparse inputs of dimension m as an LTI system with switched inputs of dimension S, hence a switched LTI system, where the number of discrete states is $\binom{m}{S}$.

In this section, we first propose a framework to jointly recover the initial condition and sparse inputs to LTI systems. We further state a theorem and derive sufficient conditions for the correctness of the perfect joint recovery of the non-sparse initial condition and sparse inputs to LTI systems. Next we assume that the initial condition is known (or without loss of generality is set to zero) and discuss the step-by-step recovery of sparse inputs.

7.4.1 Joint recovery of the initial condition and sparse inputs

In the classical linear-systems-theory setting, one assumes that the parameters of the model (Γ_N and $\mathcal{O}_N$) and sequences of inputs and outputs (U_N and Y_N) are known, and then estimating the initial conditions boils down to inverting the matrix, $\mathcal{O}_N$, which can be done if and only if the observability matrix O_N has rank n.

However, here we assume that the sequence of inputs, U_N, is *also unknown.* Assuming that $p < m$ (fewer measurements than inputs) then (7.11) is an under-

determined problem and in general an infinite number of solutions exist: there are $Nm + n$ unknowns and only Np measurements ($Nm + n > Np$). However, in this chapter we consider a special set of inputs to the linear dynamical system, namely input signals, u_k, that are sparse at each time step ($k = 0, 1, 2, \ldots, N-1$). In other words we study the inputs that are either sparse in the standard basis or have a sparse representation with respect to an overcomplete dictionary, Ψ [116].

As discussed in Section 7.2, in general, finding a sparse solution involves l_0 minimization (see (7.3)), which is a non-convex and NP-hard optimization problem. To overcome this, we appeal by analogy to the l_1 relaxation approach to recover sparse signals, and propose the following convex optimization problem to jointly recover the sparse inputs and non-sparse initial condition:

$$\min_{U_N, x_0} \|U_N\|_1 \quad \text{subject to} \quad Y_N = \mathcal{O}_N x_0 + \Gamma_N U_N, \tag{7.15}$$

where

$$\|U_N\|_1 = \sum_{k=0}^{N-1} \|u_k\|_1 \tag{7.16}$$

and x_0 and U_N are both unknown. This formulation is new in that it proposes a means by which to recover the state and unknown, but sparse, inputs.

Prior knowledge about the structured sparsity patterns can lead to more effective structured sparsity-inducing norms in the optimization formulation.

The comparison of the proposed optimization problem in (7.15)–(7.16) and the standard observability, and the standard sparse recovery problems sheds light on the conditions required for perfect joint recovery. In the standard observability problem

(assuming the sequence of inputs and outputs are known), the rank of the observability matrix must be equal to n. In the context of the standard sparse recovery (assuming the initial condition and the sequence of outputs are known), stable and exact recovery of unknown sparse inputs is guaranteed if the matrix Γ satisfies conditions such as RIP and mutual coherence (see Section 7.2.1 and 7.2.2), although these conditions are sufficient, and not necessary.

Although in general the sparsity level of inputs at each time step is allowed to be time-varying, let us assume the sparsity level of inputs remains constant over time, namely $\|u_i\|_0 = S_i = S$. A simple counting argument suggests that a solution to (7.15)–(7.16) will exist only if $pN \geq n + NS$, where S is the sparsity level of inputs at each time step, p is the number of measurements at each step, and n is the dimension of dynamical system. Therefore, we expect the smallest number of steps to be on the order of $N \geq n/p + NS/p$. This is very intuitive since, loosely speaking, n/p is the minimum number of steps to recover x_0 and NS/p is the minimum number of steps to recover U_N.

More formally, we derive sufficient conditions for joint state estimation and sparse recovery as follows.

Proposition 1. *Let Π be the projection to the orthogonal complement of the column space of the observability matrix. If the observability matrix $\mathcal{O}_N$ is full rank and the projected matrix $\Pi\Gamma_N$ is incoherent, x_0 and U_N can be uniquely recovered from Y_N.*

Proof. The projection, Π, has the following form:

$$\Pi = I - \mathcal{O}(\mathcal{O}^T\mathcal{O})^{-1}\mathcal{O}^T \tag{7.17}$$

where $\mathcal{O}_N$ is replaced by $\mathcal{O}$ for simplicity of the notation, and I, is the identity matrix of appropriate size. We first project both sides of (7.11) onto the orthogonal complement of the column space of the observability matrix ($\mathcal{O}_N$). Premultiplying the both sides of the equation (7.11) by the projection mapping, Π, results in the following equation:

$$\min_{U_N} \|U_N\|_1 \quad \text{subject to} \quad Y_\Pi = \Gamma_\Pi U_N, \tag{7.18}$$

where $Y_\Pi = \Pi Y_N$, and $\Gamma_\Pi = \Pi\Gamma$. This formulation can be viewed as a batch recovery of a sequence of sparse input signals. It attempts to recover the unknown inputs while globally satisfying all measurements constraints in the past N steps. Notice that, in this case, the recovery of the input is a standard l_1 minimization problem, where Y_Π are the measurements, Γ_Π is the dictionary, and U_Π is the sparse vector to be recovered. As a consequence, sufficient conditions for the correctness of the recovery of U_Π follow directly from the RIP or incoherence conditions applied to Γ_Π (see section 7.2.1 and 7.2.2).

Once U_Π is recovered, and assuming the system is observable, x_0 readily recovered using standard results from linear systems theory. □

7.4.2 Zero initial condition

Assuming the initial condition is known, or without loss of generality is set to zero, the convex optimization problem to recover the sparse inputs takes the following form:

$$\min_{U_N} \|U_N\|_1 \quad \text{subject to} \quad Y_N = \Gamma_N U_N \tag{7.19}$$

Similarly to the joint recovery case, sufficient conditions for the correctness of the recovery of U_N follow directly from the RIP or incoherence conditions applied to Γ_N.

An alternative to batch recovery is to recover each input sequentially as each new output becomes available. Although step-by-step recovery is not equivalent to batch recovery, it is potentially more efficient in terms of computational cost. Considering the general scenario in which the input signal at each time step, u_i, is sparse with respect to (possibly) an overcomplete dictionary (Ψ), step-by-step sparse recovery of the unknown input signal is formulated as follows:

$$\hat{u}_k = \arg \min_{u_k} \|u_k\|_1 \quad \text{subject to} \quad \tilde{y}_{k+1} = C\Psi u_k \tag{7.20}$$

where $\tilde{y_1} = y_1$ and

$$\tilde{y}_{k+1} = y_{k+1} - \sum_{j=0}^{k-1} CA^{k-j}\Psi\hat{u}_j. \tag{7.21}$$

for $k = \{1, 2, \ldots, N-1\}$.

Recovery of sparse inputs in (7.20) differs from the standard compressive sensing problem as the dictionary, Ψ, is not necessarily an orthonormal basis. As discussed in the Section 7.2, theoretical guarantees for stable recovery of signals that are sparse

with respect to an overcomplete dictionary have been provided [118, 119]. Rauhut et al. [118] showed under what conditions of the sensing matrix, C, and the overcomplete dictionary, Ψ, the global restricted isometry constant of the composed matrix, $C\Psi$, satisfies the perfect recovery condition.

7.4.3 Remarks on step-by-step recovery

As discussed in Section 7.4.2, when the initial condition is assumed to be known, a step-by-step solution to (7.20)–(7.21) for recovering the sparse inputs potentially offers a computational speed-up over batch recovery method. This improvement in the computational cost is achieved by sequentially solving an optimization problem with far fewer variables and constraints compared to the batch recovery. However, the overall performance of the step-by-step recovery hinges on stable and exact recovery of the input signals at each time step. Failure in recovering the input signal in any one step results in the propagation of the error for the rest of the steps. The challenge of capitalizing on the efficiency of step-by-step recovery, while not being hampered by a failure along the way, is an important problem.

7.5 Simulation Results

Values for the number of steps, $N = 50$, and the dimension of the inputs at each time step, $m = 50$, were kept fixed in all experiments. Supports and values of

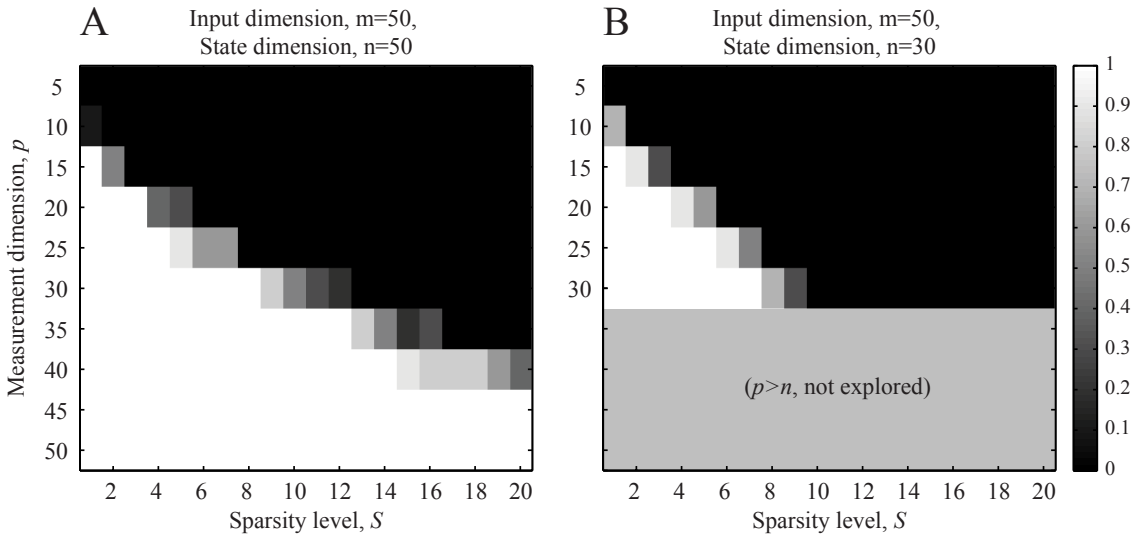

Figure 7.1: Estimate of probability across 10 simulated replicates of perfect joint recovery as posed in Section 7.4. Probability (scale on right) is given as a function of the sparsity level of the input at each time step, and the measurement dimension. (A) $n = m = 50$. Dimension of the dynamical model equals to the number of inputs at each time step. (B) $30 = n < m = 50$. Ψ is an overcomplete dictionary.

the input were allowed to vary from step to step. The dimension of the dynamical system was varied from 10 to 50 with increments of 10, i.e. $n = 10, 20, 30, 40, 50$. For each choice of n, the dimension of the measurements was varied from 5 to n with increments of 5, i.e. $p = 5, 10, \ldots, n$. For each combination of n and p, sparsity level of the inputs at each time step was varied from 1 to 20, i.e. $S = 1, 2, 3, \ldots, 20$. This range of sparsity allows for up to 40% of the entries of u_k to be nonzero at each time step ($S_{\max}/m = 20/50 = 0.4$).

We simulated (7.14) to generate times series for a wide range of parameters of the LTI system with sparse inputs. Entries of the transition matrix, $A \in \mathbb{R}^{n \times n}$, were i.i.d. Gaussian random variables with mean zero and standard deviation $1/\sqrt{n}$. It has been

shown that the distribution of eigenvalues of these random matrices obey the circular law as $n \to \infty$ [121]. For A matrices generated in our experiments, sampled matrices with eigenvalues of maximum modules greater than 0.9 are discarded to enforce stability of the sampled transition matrices. The dictionary matrix Ψ was generated in the same way as A, but not enforced to be stable. In all experiments, entries of the measurement matrix, $C \in \mathbb{R}^{p \times n}$, were independently identically distributed (i.i.d.) standard normal random variables. The value of each entry of the non-sparse initial condition, x_0, is uniformly sampled on $(-5, 5)$. For the input signal with sparsity level S at time step k, $\text{supp}(u_k)$ is a set of S integer numbers sampled uniformly on $[1, m]$ ($\|\text{supp}(u_k)\| = S$). The value of non-zero entries of u_k, are also uniformly sampled on $(-5, 5)$.

Each replicate of the experiment consisted of 600 sets of simulations. Ten replicates of the experiment were carried out. In all simulations we used the CVX software package to solve the optimization problems [122, 123].

7.5.1 Joint recovery of the initial condition and sparse unknown input

The probability of the perfect joint recovery as a function of the sparsity level of the inputs at each time step, S, and the dimension of measurements at each time step, p, is shown in Figure 7.1. The probability of perfect joint recovery is estimated using

10 simulated replicates. For a given number of measurements, p, and for the range parameters explored in this study, the ratio of the dimension of the dynamics to the number of inputs, n/m, does not significantly change the level of sparsity that admits perfect recovery. This can be seen by comparing the results in Figure 7.1(A) and (B) for $p \leq 30$ (where both systems were simulated). Note that this ratio, n/m, is the ratio of the number of rows to the number of columns of the dictionary matrix, Ψ. Of course, as n gets very small compared to the number of inputs, the overcomplete dictionary becomes highly coherent (as an extreme example when $n = 1$, $\mu(\Psi) = 1$) and it becomes impossible to exactly recover the sparse input.

Note that when $p = n$ joint recovery becomes trivial in the sense that the initial condition can be recovered independently from the inputs, directly from the first measurement, $y_0 = Cx_0$, provided the observation matrix, C, has rank n.

7.5.2 Mutual coherence

The mutual coherence of the matrices, Ψ, $C\Psi$, and Γ as a function of the dimension of the measurements at each time step is shown in Figure 7.2. Error bars show one standard deviation variation across 10 replicates of simulated data. For a fixed choice of n and m, the coherence of the randomly sampled dictionary matrix, $\mu(\Psi)$, does not statistically change as a function of p (shown in green). For a fixed choice of m (number of inputs), as n decreases, the dictionary, Ψ, becomes more coherent. As a result, for a given dimension of the measurements, the coherence of $C\Psi$ also

increases as n decreases. From the structure of the Γ matrix (see (7.13)), it can be easily verified that its mutual coherence, $\mu(\Gamma)$ (shown in red), is always bounded from below by $\mu(C\Psi)$ (shown in blue). Note that for almost all simulations, the coherence of Γ is larger than about 0.6. Based on the sufficient recovery condition in (7.7), this implies that the maximum sparsity level we can tolerate to guarantee recovery, independent of the number of measurements, is about 1.3. Experimentally and for the joint recovery of the initial condition and sparse inputs, this sufficient condition turns out to be quite conservative as shown in Figure 7.1.

7.5.3 Zero initial condition

As discussed in Section 7.4.2, when the initial condition is known (or equivalently is set to zero), the convex optimization problem takes the form in (7.19). Our numerical experiments (not shown in detail) suggest that this assumption allows the convex optimization to recover the unknown inputs with a slightly higher level (typically +1) of sparsity of the unknown inputs at each time step.

7.5.4 Discussion on optimization formulation

One can jointly recover the intitial candition and input by first projecting both sides of the equation onto the orthogonal complement of the column space of the observability matrix ($\mathcal{O}_N$) as described in Section 7.4.1. This approach appears promis-

ing because one can first solve (7.18) to recover the unknown inputs, U_N, and then solve (7.11) to recover the initial condition. Simulation results reveals that when (7.15)–(7.16) fails to jointly recover the initial condition and unknown sparse input, (7.18) also fails to recover the unknown input perfectly. This result can be explained by comparing the coherence of the Γ (before projection) and Γ_Π (after projection). As shown in Figure 7.2, the coherence of Γ_Π (black) is always greater than the coherence of Γ (see Figure 7.3).

7.6 Conclusions and Future Work

In this chapter, we proposed a convex optimization formulation to jointly recover the initial condition and unknown but sparse inputs of a linear dynamical system. While this formulation allows the sparsity of the initial condition, and state-transition matrices that preserve sparsity, it does not require these potentially limiting assumptions. Simulation results show that recovery of sparse inputs are achievable even for signals that are sparse with respect to an overcomplete dictionary.

In this study, we did not assume any structured sparsity pattern in input signals. One possible example is the case in which non-zero entries of control input are clustered. In such scenarios, the performance of the optimization solver may be improved by the use of norms that induce sparsity at the group level (e.g. l_1/l_2 norm). Structured sparsity-inducing norms other than l_1 can be formulated for the specific

problem at hand [124]. Moreover, sparse representations are shown to be useful for classification of unlabeled data [80, 81].

In next chapter, we aim to demonstrate the power of the LDS-SI in the analysis of complex motions and extend the results to the segmentation and classification of surgical gestures.

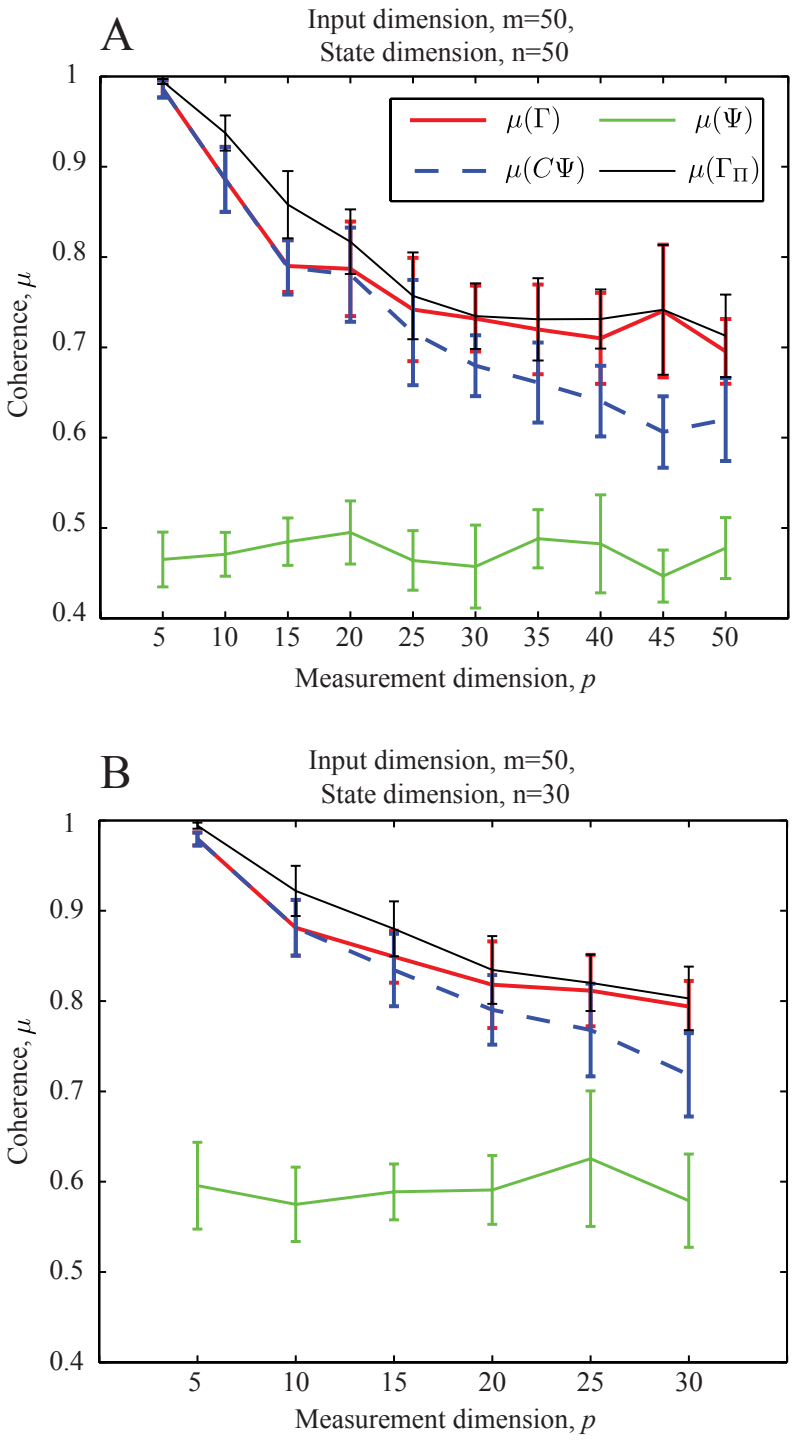

Figure 7.2: Mutual coherence of several matrices (see legend) as a function of measurement dimension, p. The coherence of the Γ matrix is always bounded from below by the coherence of $C\Psi$. (A) n=50. (B) n=30.

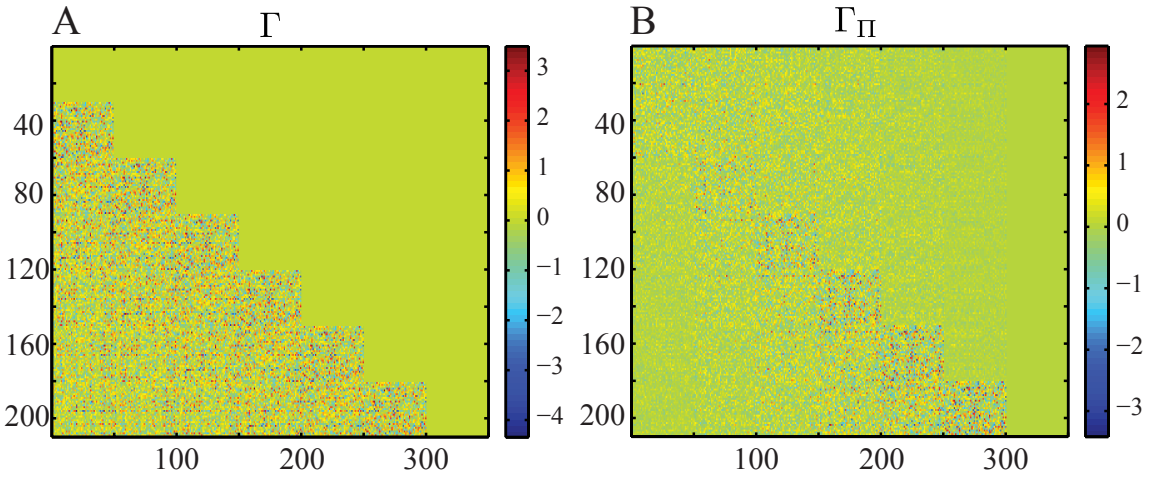

Figure 7.3: Visualization of the entries of Γ and Γ_{Π} of a representative simulation set for $m = 50, N = 7, n = 50, p = 30$. Note that in this example, after projection onto the orthogonal complement of the column space of $\mathcal{O}_N$, the colors in the first few blocks (50 columns per block) are more "muted" than before projection, suggesting that coherence is compromised as verified numerically (see Figure 7.2).

Chapter 8

LDS-SI: Application in Surgical Gesture Segmentation and Classification

8.1 Introduction

Technological advances in Robotic Minimally Invasive Surgery (RMIS) have enabled the capture of rich time-series data including the kinematics of slave robot during a surgical task. Assessment of surgical technical performance and skill, and surgical training in robotic surgery can greatly benefit from automatic segmentation and recognition of the surgical gestures. In this chapter, we demonstrate that even a simple linear dynamical model with sparse inputs can be effectively used to analyze

complex surgical gestures recorded by da Vinci robotic surgical system. Given the kinematic time-series data from the da Vinci robotic surgical system, and assuming the simplest linear dynamical system with sparse inputs, an overcomplete dictionary of inputs is learned and sparse inputs are computed using K-SVD. An SVM classifier is learned from simple features extracted from sparse inputs that carry local statistics of the surgical gesture. Experiments on a database of "suturing" task motions acquired by da Vinci robotic surgery system reveals that the proposed method performs better than the state-of-the-art methods that only use kinematics data.

8.1.1 Prior work

Previous work on surgical gesture classification in RMIS are mainly based on kinematic data recorded by the robot such as position of the robot tools, angles between robot joints, and translational and rotational velocities of both joints and tooltips. Many prior studies have quantified and analyzed the global measurements such as time to completion of a task [125, 126], the distance travelled [126], and force and torque signatures [126–128] for classification of surgical tasks. These approaches are generally easy to implement but they completely ignore the fact that a surgical task such as suturing can be decomposed to a number of surgical gestures. In recent years several studies have attempted to provide a more detailed description of a surgical task by decomposing it into a set of pre-defined atomic gestures called *surgemes* [129–134]. Examples of different surgemes in suturing include "reach needle", "insert needle",

"pull suture" etc. (see Figure 8.1).

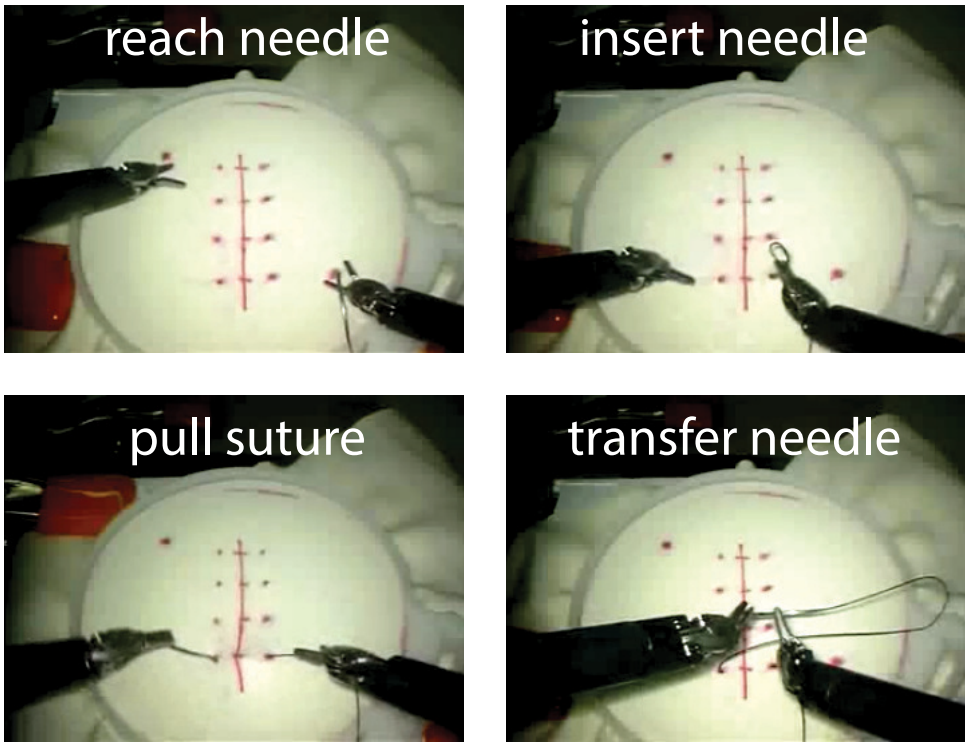

Figure 8.1: Examples of four surgemes in the suturing task.

Automatic segmentation and recognition of surgemes can facilitate automatic skill classification of trainees based on how well each of the surgical gestures have been performed in a particular surgical task. The most widely used dynamical-system based model is probabilistic Hidden Markov Model (HMM) and its variations [129–131, 133–137]. One limitation of HMM is that they model temporal dynamics with a fixed number of discrete states. However many activities involve continuous motion which may be better modeled using Linear Dynamical Systems (LDSs). Recently a surgical gesture classification method using LDSs has been proposed [75].

The method assumes that the surgical task trial is already segmented to gestures and it aims to classify the segmented data. The kinematics data or image intensities of a video frames are modeled as the output of linear dynamical systems with Gaussian noise. For the system identification they follow a sub-optimal PCA-based identification method used in modeling dynamic textures [74]. After fitting LDS models to manually segmented data they choose a dissimilarity metric for comparing the all the pairwise distances between LDSs. Based on the chosen metric a classifier k-Nearest Neighbors (k-NN) or SVMs with radial basis function (RBF) kernels) is trained for classifying novel kinematics and/or video sequences. While this approach has a significant performance in classifying the gestures, it assumes the segmentation of the data in the test set is known.

8.2 LDS-SI for Joint Segmentation and Classification of Surgical Gestures

In section 8.2.1 we first describe how we model the sequence of kinematics data as the output of a very simple linear dynamical system with sparse inputs. In section 8.2.2 we describe the joint dictionary learning and sparse coding using K-SVD. In section 8.2.3 a Support Vector Machine (SVM) classifier is trained using a histogram representation built from the sparse inputs.

8.2.1 LDS-SI model

In this section, we demonstrate that even an extremely simple linear dynamical system with sparse inputs can be effectively used for the analysis of the complex surgical gestures recorded by da Vinci robotic surgical system and performs better than the state-of-the-art methods for joint segmentation and classification of surgical gestures.

Let $\{y_k\}_{k=1}^N$ be a sequence of 78-dimensional kinematics data (see Section 8.3 for dataset description). We model these observations as the output of the following extremely simple linear dynamical system with sparse inputs:

$$\begin{aligned} x_{k+1} &= ax_k + \Psi u_k, \quad x_k \in \mathbb{R}^n, u_k \in \mathbb{R}^m, \|u_k\|_0 \leq S \\ y_k &= Cx_k, \qquad\qquad y_k \in R^p. \end{aligned} \tag{8.1}$$

where a is a scalar and describes the dynamics of the hidden state $x \in \mathbb{R}^n$, $C \in \mathbb{R}^{p \times n}$ $(n < p)$ is the observation matrix, $\Psi \in \mathbb{R}^{n \times m}$ is the overcomplete dictionary of inputs, and u_k is the sparse input signal at time k. We hypothesize that the dictionary Ψ will represent the dictionary of input signals corresponding to all possible surgemes. In other words we hypothesize that at any given time k, the pattern by which the atoms of the dictionary are activated captures useful information for identifying the specific action being performed at time k. This pattern is determined by the nonzero entries of the input signal at time instant k.

Without formally addressing the problem of joint system identification and sparse dictionary learning we employ the sub-optimal PCA-based identification method pro-

posed in [74] to estimate the observation matrix, C, and the hidden state, x_k. This step can be viewed as a dimensionality reduction of the raw data as the temporal evolution of the hidden state is completely ignored. Let

$$Y = [y_1, y_2, \cdots, y_N] \in \mathbb{R}^{p \times N} \tag{8.2}$$

denote the concatenation of all observations $\{y_k\}_{k=1}^{N}$. Assuming $Y = U\Sigma V^T$ is the compact singular value decomposition (of order $n < p$) of the Y matrix, the observation matrix and the hidden states are estimated as follows:

$$\hat{C} = U \quad \text{and} \quad \hat{X}_1^N = \Sigma V^T, \tag{8.3}$$

where $X_1^N = [x_1, x_2, \cdots, x_N] \in \mathbb{R}^{n \times N}$.

To evaluate the effect of dynamics in surgical gesture classification and to enforce the stability of the model, we let the scalar parameter, a, take a value on $[0, 1]$.

8.2.2 Dictionary learning and sparse coding

After estimating the observation matrix, C, hidden states, x_k, and choosing a constant value for a (from $[0, 1]$ interval) in Equation (8.1), we need to jointly learn an overcomplete dictionary, Ψ, and find the sparse inputs, u_k. Let $d_k = x_{k+1} - ax_k$ and

$$D = X_2^N - aX_1^{N-1} = [d_1, d_2, \ldots, d_{N-1}] \in \mathbb{R}^{n \times (N-1)} \tag{8.4}$$

denote the concatenation of all $\{d_k\}_{k=1}^{N-1}$. Likewise, let

$$U = [u_1, u_2, \ldots, u_{N-1}] \in \mathbb{R}^{m \times (N-1)} \tag{8.5}$$

denote the concatenation of all inputs $\{u_k\}_{k=1}^{N-1}$. The problem of joint dictionary learning and sparse coding can be formulated as finding the Ψ and U matrices in the following optimization problem:

$$\min_{\Psi,U} \|D - \Psi U\|_F \quad \text{subject to} \quad \|u_k\|_0 \leq S \quad \text{for all } k = 1, 2, \ldots, N-1 \tag{8.6}$$

where S is the maximum sparsity level of input at time k.

Sparse dictionary learning is a well-studied problem in the signal processing community [79, 138–142]. Two popular algorithms for sparse dictionary learning are K-SVD [140] and MOD (method of optimal direction) [139]. The optimization problem in (8.6) is nonconvex due to product ΨU and NP-hard but it is convex in either Ψ or U when the other is fixed. Both K-SVD and MOD algorithms take an alternating minimization approach by iteratively fixing one and optimizing the objective over the other parameter. When the dictionary is fixed, the problem in (8.6) reduces to finding the sparse coding of the columns of D matrix with respect to the dictionary. The solution can be approximated by l_1 minimization methods such as Basis Pursuit (BS) or Orthogonal Matching Pursuit (OMP) [78, 87]. K-SVD and MOD methods have different strategies for updating the dictionary. MOD fixes all the sparse codes and updates the entire dictionary by directly minimizing the cost in (8.6). K-SVD updates the dictionary differently by sequentially updating the columns of dictionary as well as their corresponding nonzero coefficients in the sparse code. In this work we use K-SVD algorithm to jointly learn the dictionary (Ψ) and find the sparse inputs (U).

Note that when $C = I$ (identity matrix of size $p \times p$) and $a = 0$ in Equation (8.1), the optimization in (8.6) is the same as sparse dictionary learning of the raw data itself.

8.2.3 Training

Given n_{train} training trials $\{Y^i\}_{i=1}^{n_{\text{train}}} \in \mathbb{R}^{p \times N_i}$ (where N_i is the number of data points in trial i), and their corresponding labelings $\{\mathcal{L}\}_{i=1}^{n_{\text{train}}}$, let

$$\mathcal{Y} = [Y^0, Y^1, \ldots, Y^{n_{\text{train}}}] \in \mathbb{R}^{p \times N_t} \tag{8.7}$$

denote the concatenation of all training trials, where $N_t = \sum_{i=1}^{n_{\text{train}}} N_i$. We first estimate a common observation matrix C and estimate the hidden states by computing the compact SVD of $\mathcal{Y}$ as described in section 8.2.1. After choosing a constant value for the scalar parameter, a, we learn one common dictionary for the entire training set and compute the sparse inputs by considering the following optimization cost function and using K-SVD algorithm:

$$\min_{\Psi, \mathcal{U}} \|\mathcal{D} - \Psi\mathcal{U}\|_F \quad \text{subject to} \quad \|u_k\|_0 \leq S \quad \text{for all } k \tag{8.8}$$

where

$$\mathcal{D} = [D^1, D^2, \ldots, D^{n_{\text{train}}}] \in \mathbb{R}^{n \times (N_t - n_{\text{train}})} \tag{8.9}$$

(see section 8.2.2 for the definition of the D matrix) and

$$\mathcal{U} = [U^1, U^2, \ldots, U^{n_{\text{train}}}] \in \mathbb{R}^{m \times (N_t - n_{\text{train}})}. \tag{8.10}$$

Once the dictionary is learned and sparse inputs are computed, a classifier based on the sparse inputs can be trained. Typically a soft-voting of sparse-code magnitudes is used [75] for classification. Intuitively, this can be seen as the way in which each sparse input, u_k, votes for the dictionary atoms. Formally, a vector of assignment "votes", v_k of size m is computed as follows:

$$v_k(i) = |u_k(i)|/\|u_k\|_1 \text{for all } i = 1, 2, \ldots, m. \tag{8.11}$$

As mentioned in section 8.2.1, we hypothesize that the pattern by which the dictionary atoms are activated at each time step, plays an informative role in the analysis of complex actions. At each time step, this pattern is determined by the nonzero entries of the input signal.

As an alternative to soft voting that may better extract features given the sparse input u_k, we compute a vector of assignment "votes", $v_k = [v_k(1), v_k(2), \ldots, v_k(2m)]^T$, of size $2m$ by hard-voting of sparse codes' signs as follows:

$$\begin{pmatrix} v_k(2i-1) \\ \\ v_k(2i) \end{pmatrix} = \begin{cases} \binom{1}{0} & \text{if } u_k(i) > 0 \\ \\ \binom{0}{1} & \text{if } u_k(i) < 0 \quad \text{for all } i = 1, 2, \ldots, m \\ \\ \binom{0}{0} & \text{if } u_k(i) = 0 \end{cases} \tag{8.12}$$

This serves as the binary vote for the sign by which each dictionary atom has been activated.

Regardless of the voting strategy, and to better encode the temporal information of the sparse inputs, we apply a temporal windowing. At each time step k, we build

a histogram by sum-pooling the assignment "votes" in a window of size w:

$$h(k) = \sum_{i \in W} v_i \quad \text{for all } k \text{ in the training set} \tag{8.13}$$

where W ($|W| = w$) is a set of appropriate indices in a window around time step k. Once the histograms for the entire data in the training set are built, a Support Vector Machine (SVM) classifier with RBF kernel is trained.

8.3 Experiments

8.3.1 Dataset description

We use the California dataset described in [131, 143]. The California dataset includes eight subjects with different robotic surgical experiences: novice, intermediate and expert. We evaluate the performance of the proposed method on joint segmentation and classification of surgical gestures in suturing. Each surgeon performs 4 to 5 trials. Each trial lasts about 2 minutes and the kinematic data of both master and slave manipulators of the da Vinci robotic surgical system is recorded at a constant rate of 30 Hz. Kinematic data consists of 78 motion variables including positions and velocities of both master and slave manipulators. All trials in the California dataset corresponding to suturing are manually segmented to 10 surgical gestures. Figure 8.2 shows a manually labeled trial and the surgical gestures are listed in the caption.

8.3.2 Experiment setup

We consider two different test setups for surgical gesture classification. Setup 1 is the *leave-one-supertrial-out* (LOSO) in which we leave one trial of each subject for testing, and use the remaining trials for training. Setup 2 is the *leave-one-user-out* (LOUO) where we leave all trials corresponding to one subject for testing and use all the trials from the remaining users for training.

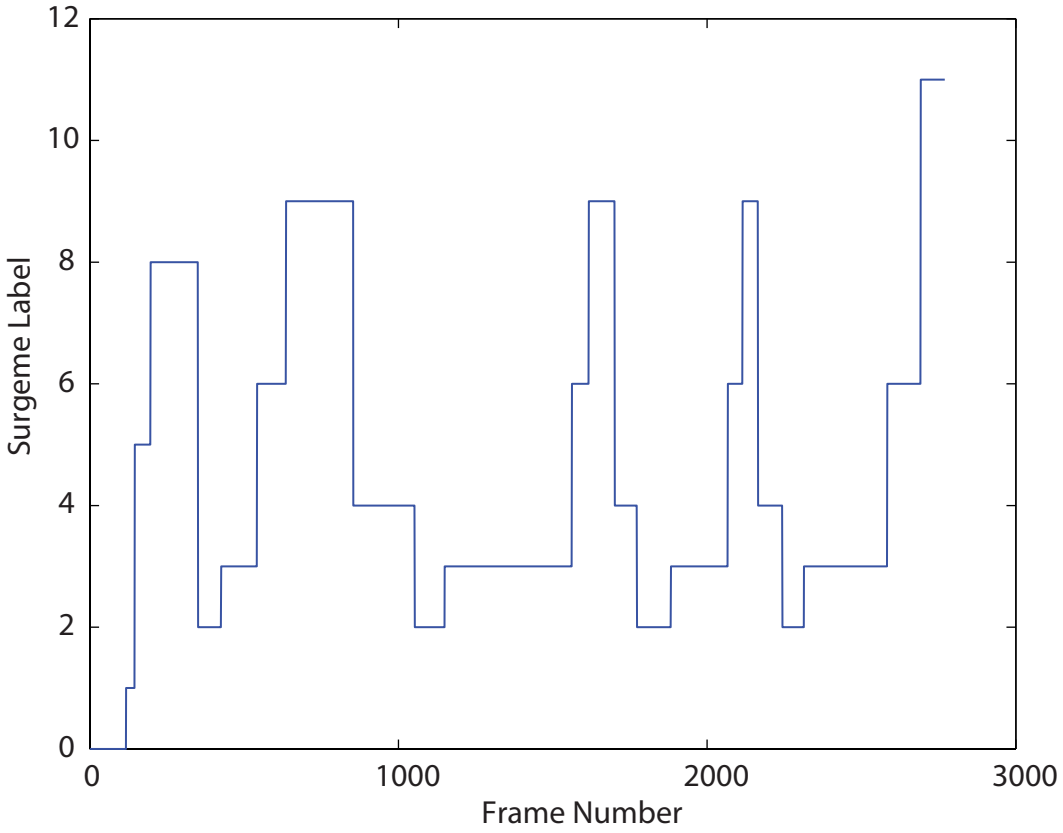

Figure 8.2: Sample surgeme time series. List of surgemes: 0. Idle motion, 1. Reach needle, 2. Position needle, 3. Insert/push needle through tissue, 4. Transfer needle, 5. Move to center with needle (right hand), 6. Pull suture with left hand, 7. Pull suture with right hand, 8. Orienting needle, 9. Right hand assisting left in tightening suture, 10. Loosen more suture, 11. Drop suture (end of trial).

8.4 Results

We first evaluate the performance of our proposed voting assignment in (8.12) as an alternative to the soft-voting assignment of the sparse codes' magnitudes in (8.11). We start with the following configuration: $a = 0, n = 25, w = 1, S = 10$. We evaluate the performance on LOSO setup for different sizes of the dictionary. The results are tabulated in Table 8.1. The results reveal that the hard-voting assignment of sparse codes' signs improves the performance of the classifier. For the rest of the experiments we use the hard-voting assignment of sparse codes' signs in (8.12).

For a thorough evaluation of our proposed method one should run experiments for all possible combinations of the parameters: dynamical model parameter a, the window size w, dictionary size m, and the sparsity level of the inputs S. Instead we start with a set of initial values for each of the parameters and we evaluate the effect of each parameter by keeping all others fixed. We first evaluate the performance of our proposed method on LOSO setup (see section 8.3 for setup description) and after finding the most promising set of parameters, we evaluate the performance on LOUO setup.

- *Effect of the dynamical model:* We start by evaluating the effect of the scalar parameter, a, that governs the very simple temporal evolution of the model. We start with $n = 25, w = 21$, $m = 40$, $S = 4$. Interestingly, the classification rate peaks at around $a = 0.7$ (Figure. 8.3(A)), suggesting that even the simplest

Table 8.1: Suturing task, LOSO setup ($a = 0, n = 25, w = 1, S = 10$). Classification rate for different voting assignments.

Suturing task	m=50	m=100	m=200
Soft-voting of magnitudes	45.52%	43.54%	35.91%
Hard-voting of signs	68.46%	66.43%	64.32%

dynamical model can increase the classification rate. We set $a = 0.7$ for the rest of the experiments.

Note that $a = 0$ completely ignores the dynamical model and (8.1) reduces to sparse dictionary learning of the raw data or its projection to a lower dimension. Sparse dictionary learning of the raw 78-dimensional kinematics data results in 58.3% classification rate for the following set of parameters: $a = 0, w = 21, m = 40, S = 4$ and $C = I$ (identity matrix of size 78×78).

- *Effect of window size:* Results shown in Figure. 8.3(B) reveals that while window size affects the classification rate, the performance is relatively stable around $w = 21$.

- *Effect of dictionary size and sparsity level of inputs:* Results shown in Figure. 8.3(C) reveals that number of dictionary atoms also improves the performance and the classification rate is stable for $m \geq 100$. Lastly for $m = 200$, we evaluated the effect of the sparsity level S. As shown in Figure 8.3(D) the classification rate peaks at $S = 8$.

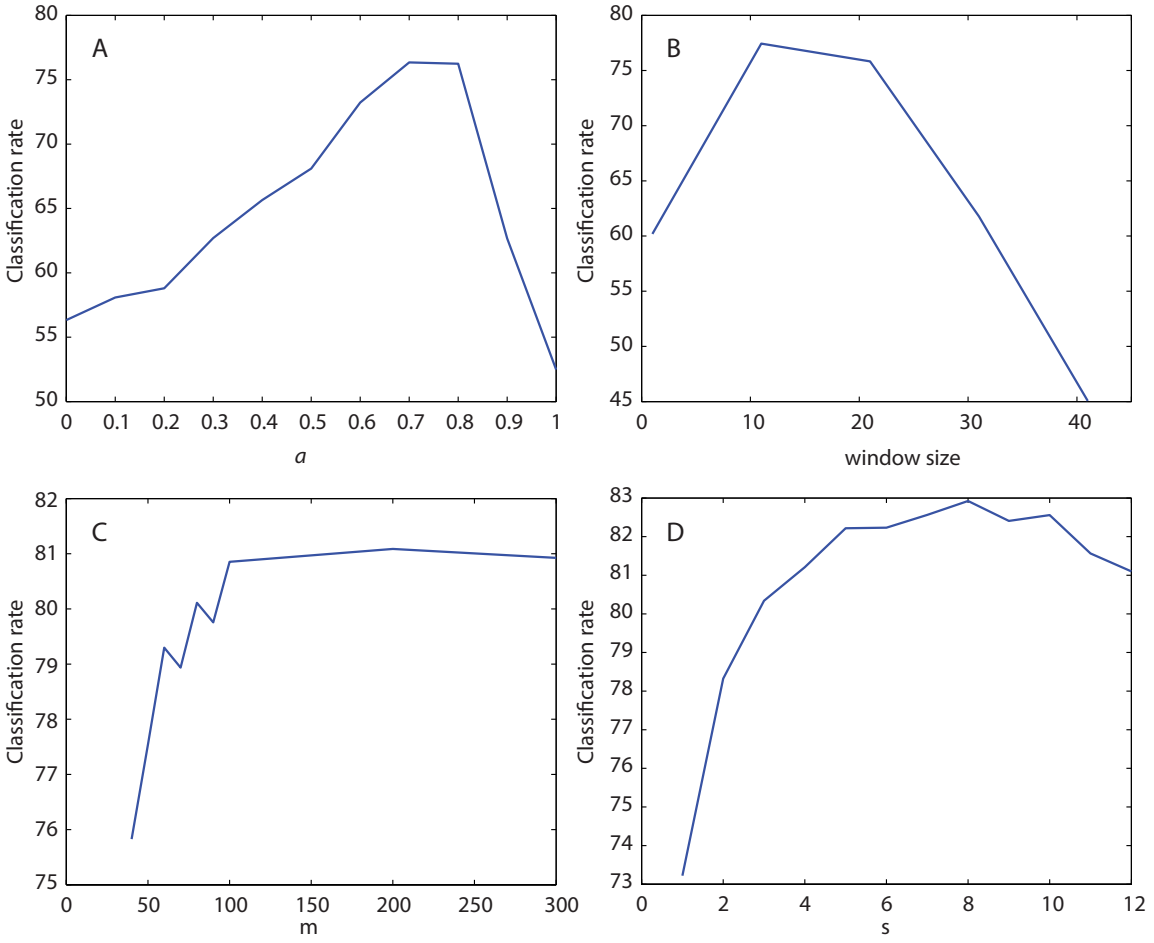

Figure 8.3: Effect of the dynamical model parameter a (**A**), window size w (**B**), dictionary sizes m (**C**), sparsity level of inputs S (**D**), on the classification rate for the suturing task in LOSO setup.

The performance for both LOSO and LOUO setups are tabulated in Table 8.2. Our results (highlighted in boldface) are better than the state-of-the-art results in LOSO setup as well as the more challenging setup of LOUO. All the methods listed in Table 8.2 are only using the kinematics data for learning the model and training the classifier. Moreover the segmentation of the test set is assumed to be unknown in all the methods listed in the table.

Table 8.2: Average gesture classification rate without assuming known segmentation (only kinematics data).

Suturing task	S-HMM [136]	CRF	MsM-CRF [137]	FA-HMM [133]	S-LDS [133]	LDS-SI
LOSO	81.1%	81.62%	80.99%	78.27%	80.79%	**82.92**%
LOUO	67.8%	68.65%	69.03%	57.2%	67.1%	**71.58**%

Results shown in Table 8.2 correspond to the following set of parameters: $a = 0.7$, $w = 21$, $m = 200$, $S = 8$. Confusion matrix in Figure 8.4 shows the performance of the classifier for all gestures.

8.5 Discussion

We demonstrated the application of LDS-SI framework in the analysis of a complex high-dimensional time-series data, namely kinematics data recorded by da Vinci robotic surgical system. The proposed method performed better than the state-of-the-art methods in surgical gesture segmentation and classification.

Without formally addressing the problem of system identification, we considered a very simple linear dynamical system with sparse inputs. We learned a common overcomplete dictionary of inputs. We have shown that the pattern by which the dictionary atoms are activated capture the characteristics of different surgical gestures and can be effectively used for segmentation and classification of the data.

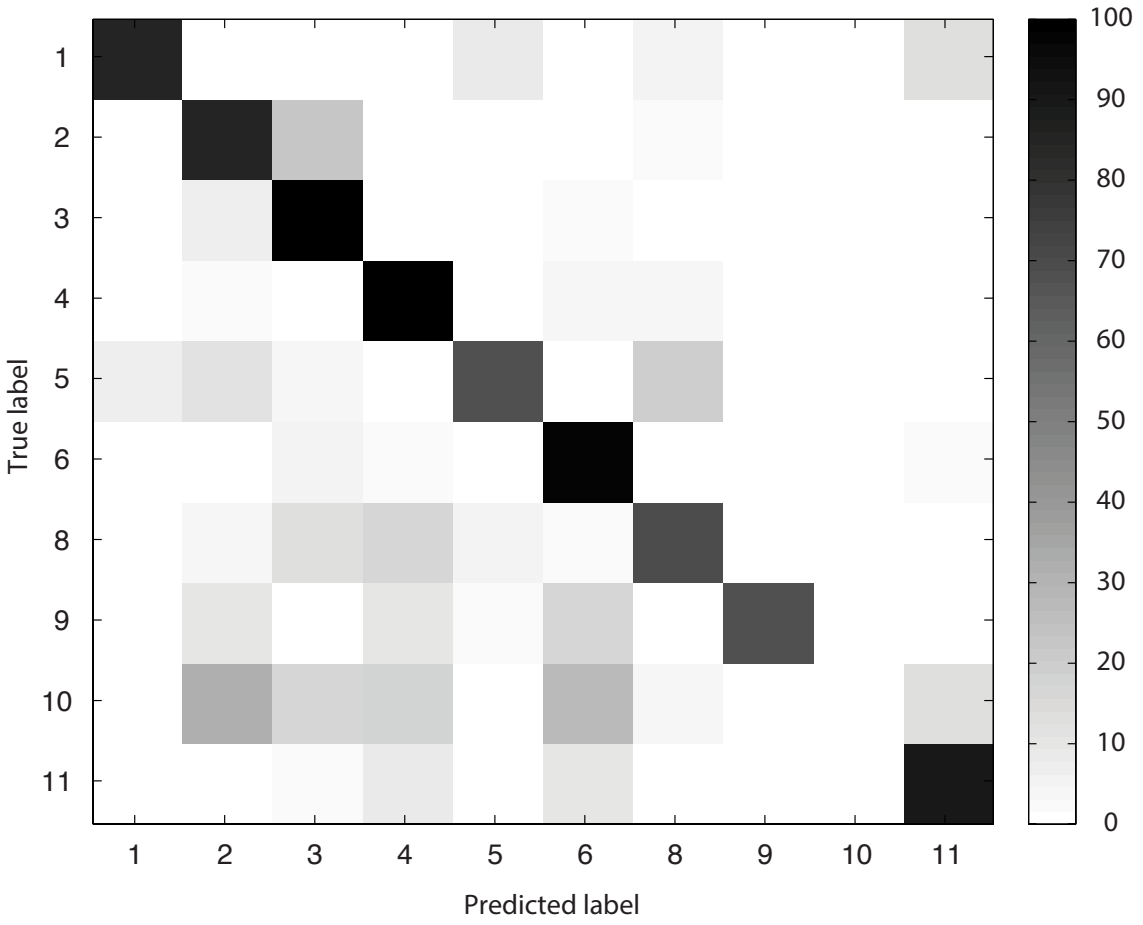

Figure 8.4: Confusion matrix corresponding to LOSO setup for suturing task.

8.6 Limitations and Directions for Future Research

- *Joint System Identification and Dictionary Learning:* In the presented work we did not formally address the system identification problem and instead we considered a very simple linear dynamical model to demonstrate the power of our

proposed framework in the analysis of surgical data. In general, this might be a major limitation for applying the method to other data sets. Future direction of research should address the problem of joint system identification and dictionary learning. More specifically, given multiple output sequences $\{Y^i\}_{i=1}^{n_{\text{train}}}$, the goal is to identify the state-transition matrix A, an overcomplete dictionary of inputs Ψ, the observation matrix C, without knowing the initial states $\{x_0^i\}_{i=1}^{n_{\text{train}}}$ or the sequence of inputs $\{U^i\}_{i=1}^{n_{\text{train}}}$. This is a blind identification problem. Classical system identification algorithms such as N4SID [144] assume that the sequence of inputs to the system is also observed or stationary zero-mean Gaussian process. One possible approach for addressing the blind identification problem is an alternating minimization framework by combining the classical system identification techniques and dictionary learning algorithms.

- *Structured dictionary learning and structured sparse inputs:* In this work we did not assume any structured sparsity patterns such as group sparsity pattern for the inputs. Recent works have demonstrated that dictionary learning and sparse coding algorithms can be efficiently modified and provide practical benefits when a prior knowledge about a particular structure exists [124, 141, 145–147].

  For instance, Zelnik-Manor et al. [141] proposed an algorithm for designing dictionaries that admits block-sparse representations. The algorithm (BK-SVD+SAC), which is a natural extension of K-SVD algorithm, provides a more accurate representation for the signals that can be well approximated as a union

of subspaces.

For the purpose of LDS-SI, one can think of a dictionary of inputs Ψ that can be decomposed into L smaller dictionaries $\Psi = [\Psi_1, \Psi_2, \ldots, \Psi_L]$, where each dictionary Ψ_i or a specific combination of them may represent a dictionary of inputs for a specific action or surgical gesture. When such a structure is learned, group-sparse coding of inputs may provide a simpler solution for directly decoding complex actions.

Chapter 9

Conclusion

In this dissertation we utilized two different modeling approaches to study complex biological and mechanical movement from two different perspectives. In Part I we primarily focused on a template *physics-based* modeling approach to shed light on a long-standing question in animal locomotion: why do animals often produce substantial forces in directions that do not directly contribute to movement? We examined the weakly electric knifefish, a well-suited model system to investigate the relationship between mutually opposing forces and locomotor control. We used slow-motion videography to study the ribbon-fin motion and developed a physics-based template model at the task-level for tracking behavior. Using the developed physics-based model integrated with experiments with a biomimetic robot, we showed that mutually opposing forces generated by the ribbon-fin improves the fore-aft maneuverability (by decreasing the control effort), and concurrently enhances the passive

stability (stabilization without active feedback control) by providing a damping-like force to reject perturbation, thus simplifies control. The results may also inspire the exploration of a new set of strategies for the design and control of robots.

In Part II, we proposed a more general *data-driven, system-theoretic* framework for decoding complex behaviors. Specifically, we introduced a new class of linear time-invariant dynamical systems with sparse inputs (LDS-SI). In the proposed framework, at each time instant, the input to the system is a sparse linear combination of just a few atoms of an overcomplete dictionary of inputs. In the context of complex behaviors, the dictionary may represent the dictionary of all possible simple behaviors. We studied a fundamental problem of state estimation with *unknown* inputs. We proposed a convex optimization formulation for the joint recovery of initial condition and inputs. We derived sufficient conditions for perfect joint recovery. Finally we demonstrated the power of the proposed framework in the analysis of complex gestures in robotic surgery.

Modeling complex biological and mechanical movements plays a critical role in mutual interaction between biology and robotics fields. On the one side, richness and diversity in animal locomotion are a great source of inspiration in engineering design of future robots. On the other side, neuromechanical hypotheses can be tested on bio-inspired and bio-mimetic robots. At a higher level, developing more unified frameworks for modeling, decoding, and analysis of complex behaviors in both biological and mechanical movements (with applications such as automatic activity recognition)

are essential for ongoing advances in cyber-physical systems.

Bibliography

[1] R. Blickhan and R. J. Full, "Similarity in multilegged locomotion: Bouncing like a monopode," *J. Comp. Physiol. A*, vol. 173, no. 5, pp. 509–517, Nov. 1993.

[2] P. Holmes, R. J. Full, D. Koditschek, and J. Guckenheimer, "The dynamics of legged locomotion: Models, analyses, and challenges," *SIAM Rev.*, vol. 48, no. 2, pp. 207–304, Feb. 2006.

[3] N. J. Cowan, J. Lee, and R. J. Full, "Task-level control of rapid wall following in the American cockroach," *J. Exp. Biol.*, vol. 209, no. 9, pp. 1617–1629, 2006.

[4] T. L. Hedrick and A. K. Robinson, "Within-wingbeat damping: dynamics of continuous free-flight yaw turns in manduca sexta," *Biol. Letters*, vol. 6, no. 3, pp. 422–425, 2010.

[5] R. J. Full and D. E. Koditschek, "Templates and anchors: Neuromechanical hypotheses of legged locomotion," *J. Exp. Biol.*, vol. 202, pp. 3325–3332, 1999.

[6] E. D. Tytell, C.-Y. Hsu, T. L. Williams, A. H. Cohen, and L. J. Fauci, "In-

teractions between internal forces, body stiffness, and fluid environment in a neuromechanical model of lamprey swimming," *Proceedings of the National Academy of Sciences*, vol. 107, no. 46, pp. 19 832–19 837, 2010.

[7] S. Sefati, I. D. Neveln, E. Roth, T. Mitchell, J. B. Snyder, M. A. MacIver, E. S. Fortune, and N. J. Cowan, "Mutually opposing forces during locomotion can eliminate the tradeoff between maneuverability and stability," *Proc. Nat. Acad. Sci.*, vol. 110, no. 47, pp. 18 798–18 803, 2013.

[8] J. Schmitt and P. Holmes, "Mechanical models for insect locomotion: dynamics and stability in the horizontal plane I. Theory." *Biol. Cybern.*, vol. 83, pp. 501–515, 2000.

[9] R. Altendorfer, D. E. Koditschek, and P. Holmes, "Stability analysis of legged locomotion models by symmetry-factored return maps," *The International Journal of Robotics Research*, vol. 23, no. 10-11, pp. 979–999, 2004.

[10] R. Altendorfer, N. Moore, H. Komsuoglu, M. Buehler, H. B. Brown Jr, D. McMordie, U. Saranli, R. Full, and D. E. Koditschek, "Rhex: a biologically inspired hexapod runner," *Autonomous Robots*, vol. 11, no. 3, pp. 207–213, 2001.

[11] N. J. Cowan and E. S. Fortune, "The critical role of locomotion mechanics in decoding sensory systems," *J. Neurosci.*, vol. 27, no. 5, pp. 1123–1128, 2007.

[12] G. J. Rose and J. G. Canfield, "Longitudinal tracking responses of the weakly

electric fish, *Sternopygus*," *J. Comp. Physiol. A*, vol. 171, no. 6, pp. 791–798, Jan. 1993.

[13] O. Ekeberg, "A combined neuronal and mechanical model of fish swimming," *Biol. Cybern.*, vol. 69, no. 5-6, pp. 363–374, Oct. 1993.

[14] M. Frye and M. Dickinson, "Fly Flight, A Model for the Neural Control of Complex Behavior," *Neuron*, vol. 32, no. 3, pp. 385–388, Nov. 2001.

[15] K. Nishikawa, A. A. Biewener, P. Aerts, A. N. Ahn, H. J. Chiel, M. A. Daley, T. L. Daniel, R. J. Full, M. E. Hale, T. L. Hedrick, A. K. Lappin, T. R. Nichols, R. D. Quinn, R. A. Satterlie, and B. Szymik, "Neuromechanics: an integrative approach for understanding motor control," *Integr. Comp. Biol.*, vol. 47, no. 1, pp. 16–54, Jul 2007.

[16] E. Roth, K. Zhuang, S. A. Stamper, E. S. Fortune, and N. J. Cowan, "Stimulus predictability mediates a switch in locomotor smooth pursuit performance for *Eigenmannia virescens*." *J. Exp. Biol.*, vol. 214, no. 7, pp. 1170–1180, Apr. 2011.

[17] L. A. Miller, D. I. Goldman, T. L. Hedrick, E. D. Tytell, Z. J. Wang, J. Yen, and S. Alben, "Using computational and mechanical models to study animal locomotion," *Integr. Comp. Biol.*, vol. 52, no. 5, pp. 553–575, 2012.

[18] R. Mittal, "Computational modeling in biohydrodynamics: trends, challenges,

and recent advances," *Oceanic Engineering, IEEE Journal of*, vol. 29, no. 3, pp. 595–604, 2004.

[19] Z. J. Wang, J. M. Birch, and M. H. Dickinson, "Unsteady forces and flows in low reynolds number hovering flight: two-dimensional computations vs robotic wing experiments," *J. Exp. Biol.*, vol. 207, no. 3, pp. 449–460, 2004.

[20] H. Luo, R. Mittal, X. Zheng, S. A. Bielamowicz, R. J. Walsh, and J. K. Hahn, "An immersed-boundary method for flow–structure interaction in biological systems with application to phonation," *Journal of computational physics*, vol. 227, no. 22, pp. 9303–9332, 2008.

[21] A. A. Shirgaonkar, O. M. Curet, N. A. Patankar, and M. A. Maciver, "The hydrodynamics of ribbon-fin propulsion during impulsive motion." *J. Exp. Biol.*, vol. 211, no. Pt 21, pp. 3490–3503, Nov. 2008.

[22] G. V. Lauder, E. J. Anderson, J. Tangorra, and P. G. Madden, "Fish biorobotics: kinematics and hydrodynamics of self-propulsion," *J. Exp. Biol.*, vol. 210, no. 16, pp. 2767–2780, 2007.

[23] S. Sefati, I. Neveln, M. A. MacIver, E. S. Fortune, and N. J. Cowan, "Counter-propagating waves enhance maneuverability and stability: a bio-inspired strategy for robotic ribbon-fin propulsion," in *Proc IEEE Int Conf Biomed Robot Biomech.*, 2012, pp. 1620–1625.

BIBLIOGRAPHY

[24] O. M. Curet, N. A. Patankar, G. V. Lauder, and M. A. MacIver, "Mechanical properties of a bio-inspired robotic knifefish with an undulatory propulsor," *Bioinspir. Biomim.*, vol. 6, no. 2, p. 026004, 2011.

[25] W. B. Dickson, A. D. Straw, and M. H. Dickinson, "Integrative model of drosophila flight," *AIAA J.*, vol. 46, no. 9, pp. 2150–2164, 2008.

[26] T. L. Hedrick, B. Cheng, and X. Y. Deng, "Wingbeat Time and the Scaling of Passive Rotational Damping in Flapping Flight," *Science*, vol. 324, no. 5924, pp. 252–255+, 2009.

[27] J. P. Dyhr, K. A. Morgansen, T. L. Daniel, and N. J. Cowan, "Flexible strategies for flight control: an active role for the abdomen," *The Journal of experimental biology*, vol. 216, no. 9, pp. 1523–1536, 2013.

[28] J. Lighthill, *Mathematical Biofluiddynamics*, ser. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial Mathematics, 1975, vol. 17.

[29] R. W. Blake, *Fish Locomotion*. Cambridge Univ. Press, 1983.

[30] ——, "Swimming in electric eels and knifefishes," *Can J Zool*, vol. 61, pp. 1432–1441, 1983.

[31] G. V. Lauder and P. G. A. Madden, "Learning from fish: Kinematics and exper-

imental hydrodynamics for roboticists," *International Journal of Automation and Computing*, vol. 3, no. 4, p. 325, 2006.

[32] C. T. Farley and T. C. Ko, "Mechanics of locomotion in lizards." *J. Exp. Biol.*, vol. 200, no. 16, pp. 2177–2188, 1997.

[33] R. J. Full and M. S. Tu, "Mechanics of a rapid running insect: two-, four-and six-legged locomotion," *J. Exp. Biol.*, vol. 156, no. 1, pp. 215–231, 1991.

[34] R. Blickhan and R. J. Full, "Similarity in multilegged locomotion: Bouncing like a monopode," *J. Comp. Physiol. A*, vol. 173, no. 5, 1993.

[35] M. H. Dickinson, C. T. Farley, R. J. Full, M. A. R. Koehl, R. Kram, and S. Lehman, "How animals move: an integrative view," *Science*, vol. 288, pp. 100–106, 2000.

[36] J. D. H. Sprayberry and T. L. Daniel, "Flower tracking in hawkmoths: behavior and energetics," *J. Exp. Biol.*, vol. 210, no. 1, pp. 37–45, 2007.

[37] G. J. Rose and J. G. Canfield, "Longitudinal tracking responses of the weakly electric fish, sternopygus," *J. Comp. Physiol. A*, vol. 171, no. 6, pp. 791–798, 1993.

[38] P. W. Webb, "Stability and maneuverability," *Fish Physiol.*, vol. 23, pp. 281–332, 2005.

BIBLIOGRAPHY

[39] P. W. Webb and A. G. Fairchild, "Performance and maneuverability of three species of teleostean fishes," *Can. J. Zool.*, vol. 79, no. 10, pp. 1866–1877, 2001.

[40] R. J. Full, T. Kubow, J. Schmitt, P. Holmes, and D. Koditschek, "Quantifying dynamic stability and maneuverability in legged locomotion," *Integr. Comp. Biol.*, vol. 42, no. 1, pp. 149–157, 2002.

[41] S. B. Emerson and M. A. R. Koehl, "The interaction of behavioral and morphological change in the evolution of a novel locomotor type: "flying" frogs," *Evolution*, pp. 1931–1946, 1990.

[42] D. Weihs, "Stability versus maneuverability in aquatic locomotion," *Integr. Comp. Biol.*, vol. 42, no. 1, pp. 127–134, 2002.

[43] R. Von Mises, *Theory of flight*. Dover Publications, 1959.

[44] C. A. Marchaj and Z. A. Marchaj, *Aero-hydrodynamics of sailing*. Dodd, Mead, 1980.

[45] S. A. Brandt, *Introduction to aeronautics: a design perspective*. AIAA, 2004.

[46] T. L. Hedrick, "Damping in flapping flight and its implications for manoeuvring, scaling and evolution," *J. Exp. Biol.*, vol. 214, no. 24, pp. 4073–4081, 2011.

[47] D. L. Jindrich and R. J. Full, "Many-legged maneuverability: dynamics of turning in hexapods," *J. Exp. Biol.*, vol. 202, no. 12, pp. 1603–1623, 1999.

BIBLIOGRAPHY

[48] S. Sefati, E. S. Fortune, and N. J. Cowan, "Counter-propogating waves in the ribbon fin of *Eigenmannia virescens* enhance maneuverability," in *Soc Int Comp Biol*, 2010.

[49] S. Sefati, T. Mitchell, E. Fortune, and N. Cowan, "An experimentally validated fluid dynamic model of ribbon-finned propulsion reveals how thrust is controlled by counter-propagating waves," in *Soc Int Comp Biol*, vol. 51, 2011, pp. E126–E126.

[50] N. J. Cowan, M. M. Ankarali, J. P. Dyhr, M. S. Madhav, E. Roth, S. Sefati, S. Sponberg, S. A. Stamper, E. S. Fortune, and T. L. Daniel, "Feedback control as a framework for understanding tradeoffs in biology," *Integr. Comp. Biol.*, vol. 54, no. 2, pp. 223–237, 2014.

[51] R. Ruiz-Torres, O. M. Curet, G. V. Lauder, and M. A. MacIver, "Kinematics of the ribbon fin in hovering and swimming of the electric ghost knifefish," *J. Exp. Biol.*, vol. 216, no. 5, pp. 823–834, 2013.

[52] J. S. Albert, *Species diversity and phylogenetic systematics of American knifefishes (Gymnotiformes, Teleostei)*. Division of Ichthyology, Museum of Zoology, University of Michigan, 2001.

[53] M. Sfakiotakis, D. M. Lane, and J. B. C. Davies, "Review of fish swimming modes for aquatic locomotion," *IEEE J. Ocean. Eng.*, vol. 24, no. 2, pp. 237–252, 1999.

BIBLIOGRAPHY

[54] J. Lighthill and R. Blake, "Biofluiddynamics of balistiform and gymnotiform locomotion. part 1. biological background, and analysis by elongated-body theory," *J. Fluid Mech.*, vol. 212, no. 1, pp. 183–207, 1990.

[55] O. M. Curet, N. A. Patankar, G. V. Lauder, and M. A. Maciver, "Aquatic manoeuvering with counter-propagating waves: a novel locomotive strategy." *J. R. Soc. Interface*, vol. 8, no. 60, pp. 1041–1050, 2011.

[56] A. Hitschfeld, S. A. Stamper, K. Vonderschen, E. S. Fortune, and M. J. Chacron, "Effects of restraint and immobilization on electrosensory behaviors of weakly electric fish," *ILAR J.*, vol. 50, no. 9, pp. 361–372, 2009.

[57] G. J. Rose and J. G. Canfield, "Longitudinal tracking responses of *Eigenmannia* and *Sternopygus*," *J. Comp. Physiol. A*, vol. 173, pp. 698–700, 1993.

[58] T. L. Hedrick, "Software techniques for two-and three-dimensional kinematic measurements of biological and biomimetic systems," *Bioinspir. Biomim.*, vol. 3, p. 034001, 2008.

[59] R. Ruiz-Torres, O. M. Curet, G. V. Lauder, and M. A. MacIver, "Kinematics of the ribbon fin in hovering and swimming of the electric ghost knifefish," *J. Exp. Biol.*, vol. 216, no. 5, pp. 823–834, 2013.

[60] M. Epstein, J. E. Colgate, and M. A. MacIver, "A biologically inspired robotic ribbon fin," in *Proceedings of IEEE/RSJ International Conference on Intelli-*

gent Robots and Systems. Workshop on Morphology, Control, and Passive Dynamics, 2005.

[61] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 1: Classical Results and Geometric Methods (Applied and Numerical Harmonic Analysis)*, 1st ed. Birkhäuser Boston, Sep. 2009.

[62] F. L. Lewis and V. L. Syrmos, *Optimal control.* Wiley, 1995.

[63] S. N. Fry, R. Sayaman, and M. H. Dickinson, "The aerodynamics of free-flight maneuvers in drosophila," *Science*, vol. 300, no. 5618, pp. 495–498, 2003.

[64] K. J. Astrom and R. M. Murray, *Feedback systems: an introduction for scientists and engineers.* Princeton University Press, 2008.

[65] H. Kitano, "Biological robustness." *Nat. Rev. Genet.*, vol. 5, no. 11, pp. 826–837, Nov. 2004.

[66] M. E. Csete and J. C. Doyle, "Reverse engineering of biological complexity." *Science*, vol. 295, no. 5560, pp. 1664–1669, Mar. 2002.

[67] J. M. Smith, "The importance of the nervous system in the evolution of animal flight," *Evolution*, vol. 6, no. 1, pp. 127–129, 1952.

[68] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, no. 5712, pp. 1082–1085, 2005.

BIBLIOGRAPHY

[69] R. J. Full and D. E. Koditschek, "Templates and anchors: neuromechanical hypotheses of legged locomotion on land," *J. Exp. Biol.*, vol. 202, no. 23, pp. 3325–3332, 1999.

[70] L. H. Ting, J. M. Macpherson *et al.*, "A limited set of muscle synergies for force control during a postural task," *J. Neurophysiol.*, vol. 93, no. 1, pp. 609–613, 2005.

[71] J. B. Snyder, M. E. Nelson, J. W. Burdick, and M. A. MacIver, "Omnidirectional sensory and motor volumes in electric fish," *PLoS Biol.*, vol. 5, no. 11, p. e301, 2007.

[72] T. L. Hedrick and A. K. Robinson, "Within-wingbeat damping: dynamics of continuous free-flight yaw turns in manduca sexta," *Biol. Letters*, vol. 6, no. 3, pp. 422–425, 2010.

[73] M. A. MacIver, N. A. Patankar, and A. A. Shirgaonkar, "Energy-information trade-offs between movement and sensing," *PLoS Comp. Biol.*, vol. 6, no. 5, p. e1000769, May 2010.

[74] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, "Dynamic textures," *International Journal of Computer Vision*, vol. 51, no. 2, pp. 91–109, 2003.

[75] L. Zappella, B. Béjar, G. Hager, and R. Vidal, "Surgical gesture classification

from video and kinematic data," *Medical image analysis*, vol. 17, no. 7, pp. 732–745, 2013.

[76] A. Bissacco, A. Chiuso, Y. Ma, and S. Soatto, "Recognition of human gaits," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 2. IEEE, 2001, pp. II–52.

[77] A. Veeraraghavan, A. K. Roy-Chowdhury, and R. Chellappa, "Matching shape sequences in video with applications in human movement analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1896–1909, 2005.

[78] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM journal on scientific computing*, vol. 20, no. 1, pp. 33–61, 1998.

[79] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by v1?" *Vision research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[80] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 2790–2797.

[81] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face

recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.

[82] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 21–30, 2008.

[83] N. Vaswani, "Kalman filtered compressed sensing," in *15th IEEE International Conference on Image Processing, 2008. ICIP 2008.* IEEE, 2008, pp. 893–896.

[84] M. B. Wakin, B. M. Sanandaji, and T. L. Vincent, "On the observability of linear systems from random, compressive measurements," in *49th IEEE Conference on Decision and Control (CDC).* IEEE, 2010, pp. 4447–4454.

[85] A. Charles, M. S. Asif, J. Romberg, and C. Rozell, "Sparsity penalties in dynamical system estimation," in *45th Annual Conference on Information Sciences and Systems (CISS).* IEEE, 2011, pp. 1–6.

[86] Y. Pati, R. Rezaiifar, P. S. Krishnaprasad, and W. P. Dayawansa, "A fast recursive algorithm for system identification and model reduction using rational wavelets," in *The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers.* IEEE, 1993, pp. 35–39.

[87] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition,"

in *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*. IEEE, 1993, pp. 40–44.

[88] M. Fazel, H. Hindi, and S. P. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *American Control Conference, 2001. Proceedings of the 2001*, vol. 6. IEEE, 2001, pp. 4734–4739.

[89] Z. Liu, A. Hansson, and L. Vandenberghe, "Nuclear norm system identification with missing inputs and outputs," *Systems & Control Letters*, vol. 62, no. 8, pp. 605–612, 2013.

[90] A. Hansson, Z. Liu, and L. Vandenberghe, "Subspace system identification via weighted nuclear norm optimization," in *IEEE 51st Annual Conference on Decision and Control (CDC)*. IEEE, 2012, pp. 3439–3444.

[91] D. Sadigh, H. Ohlsson, S. S. Sastry, and S. A. Seshia, "Robust subspace system identification via weighted nuclear norm optimization," *arXiv preprint arXiv:1312.2132*, 2013.

[92] M. Fazel, T. K. Pong, D. Sun, and P. Tseng, "Hankel matrix rank minimization with applications to system identification and realization," *SIAM Journal on Matrix Analysis and Applications*, vol. 34, no. 3, pp. 946–977, 2013.

[93] A. Chiuso and G. Pillonetto, "Learning sparse dynamic linear systems using

stable spline kernels and exponential hyperpriors," in *Neural Information Processing Systems NIPS*, 2010, pp. 397–405.

[94] B. M. Sanandaji, T. L. Vincent, M. B. Wakin, R. Tóth, and K. Poolla, "Compressive system identification of lti and ltv arx models," in *Conference on Decision and Control and European Control Conference (CDC-ECC)*. IEEE, 2011, pp. 791–798.

[95] D. Napoletani, T. D. Sauer *et al.*, "Reconstructing the topology of sparsely connected dynamical networks," *Physical Review-Section E-Statistical Nonlinear and Soft Matter Physics*, vol. 77, no. 2, p. 26103, 2008.

[96] B. M. Sanandaji, T. L. Vincent, and M. B. Wakin, "Exact topology identification of large-scale interconnected dynamical systems from compressive observations," in *American Control Conference (ACC)*. IEEE, 2011, pp. 649–656.

[97] N. Ozay, M. Sznaier, C. Lagoa, and O. Camps, "A sparsification approach to set membership identification of a class of affine hybrid systems," in *47th IEEE Conference on Decision and Control*. IEEE, 2008, pp. 123–130.

[98] L. Bako, "Identification of switched linear systems via sparse optimization," *Automatica*, vol. 47, no. 4, pp. 668–677, 2011.

[99] H. Ohlsson, L. Ljung, and S. Boyd, "Segmentation of arx-models using sum-of-norms regularization," *Automatica*, vol. 46, no. 6, pp. 1107–1111, 2010.

BIBLIOGRAPHY

[100] N. Kalouptsidis, G. Mileounis, B. Babadi, and V. Tarokh, "Adaptive algorithms for sparse system identification," *Signal Processing*, vol. 91, no. 8, pp. 1910–1919, 2011.

[101] B. Ghanem and N. Ahuja, "Sparse coding of linear dynamical systems with an application to dynamic texture recognition," in *International Conference on Pattern Recognition (ICPR)*. IEEE, 2010, pp. 987–990.

[102] B. Afsari and R. Vidal, "The alignment distance on spaces of linear dynamical systems," in *IEEE 52nd Annual Conference on Decision and Control (CDC)*, 2013, pp. 1162–1167.

[103] D. Angelosante, S. I. Roumeliotis, and G. B. Giannakis, "Lasso-Kalman smoother for tracking sparse signals," in *Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*. IEEE, 2009, pp. 181–185.

[104] M. S. Asif and J. Romberg, "Dynamic updating for l_1 minimization," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 421–434, 2010.

[105] J. Ziniel, L. C. Potter, and P. Schniter, "Tracking and smoothing of time-varying sparse signals via approximate belief propagation," in *Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*. IEEE, 2010, pp. 808–812.

BIBLIOGRAPHY

[106] S. S. Haykin, *Blind deconvolution.* PTR Prentice Hall Englewood Cliffs, 1994.

[107] J. F. Claerbout and F. Muir, "Robust modeling with erratic data," *Geophysics*, vol. 38, no. 5, pp. 826–844, 1973.

[108] H. L. Taylor, S. C. Banks, and J. F. McCoy, "Deconvolution with the l_1 norm," *Geophysics*, vol. 44, no. 1, pp. 39–52, 1979.

[109] M. S. O'Brien, A. N. Sinclair, and S. M. Kramer, "Recovery of a sparse spike time series by l_1 norm deconvolution," *Signal Processing, IEEE Transactions on*, vol. 42, no. 12, pp. 3353–3365, 1994.

[110] D. Luengo, I. Santamaría, J. Ibáñez, L. Vielva, and C. Pantaleón, "A fast blind simo channel identification algorithm for sparse sources," *Signal Processing Letters, IEEE*, vol. 10, no. 5, pp. 148–151, 2003.

[111] M. Raptis, K. Wnuk, and S. Soatto, "Spike train driven dynamical models for human actions," in *Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, 2010, pp. 2077–2084.

[112] A. Charles and C. Rozell, "Dynamic filtering of sparse signals using reweighted l_1," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, 2013.

[113] E. J. Candes and T. Tao, "Decoding by linear programming," *Information Theory, IEEE Transactions on*, vol. 51, no. 12, pp. 4203–4215, 2005.

[114] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9, pp. 589–592, 2008.

[115] J. A. Tropp, "Just relax: Convex programming methods for identifying sparse signals in noise," *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1030–1051, 2006.

[116] D. L. Donoho, M. Elad, and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Transactions on Information Theory*, vol. 52, no. 1, pp. 6–18, 2006.

[117] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on pure and applied mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.

[118] H. Rauhut, K. Schnass, and P. Vandergheynst, "Compressed sensing and redundant dictionaries," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2210–2219, 2008.

[119] E. J. Candes, Y. C. Eldar, D. Needell, and P. Randall, "Compressed sensing with coherent and redundant dictionaries," *Applied and Computational Harmonic Analysis*, vol. 31, no. 1, pp. 59–73, 2011.

[120] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A simple proof of the

restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.

[121] A. Edelman, "The probability that a random real gaussian matrix has k real eigenvalues, related distributions, and the circular law," *Journal of Multivariate Analysis*, vol. 60, no. 2, pp. 203–232, 1997.

[122] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," 2008.

[123] M. C. Grant and S. P. Boyd, "Graph implementations for nonsmooth convex programs," in *Recent advances in learning and control*. Springer, 2008, pp. 95–110.

[124] F. Bach, R. Jenatton, J. Mairal, G. Obozinski *et al.*, "Structured sparsity through convex optimization," *Statistical Science*, vol. 27, no. 4, pp. 450–468, 2012.

[125] V. Datta, S. Mackay, M. Mandalia, and A. Darzi, "The use of electromagnetic motion tracking analysis to objectively measure open surgical skill in the laboratory-based model," *Journal of the American College of Surgeons*, vol. 193, no. 5, pp. 479–485, 2001.

[126] T. N. Judkins, D. Oleynikov, and N. Stergiou, "Objective evaluation of ex-

pert and novice performance during robotic surgical training tasks," *Surgical endoscopy*, vol. 23, no. 3, pp. 590–597, 2009.

[127] C. Richards, J. Rosen, B. Hannaford, C. Pellegrini, and M. Sinanan, "Skills evaluation in minimally invasive surgery using force/torque signatures," *Surgical Endoscopy*, vol. 14, no. 9, pp. 791–798, 2000.

[128] Y. Yamauchi, J. Yamashita, O. Morikawa, R. Hashimoto, M. Mochimaru, Y. Fukui, H. Uno, and K. Yokoyama, "Surgical skill evaluation by force data for endoscopic sinus surgery training system," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI*. Springer, 2002, pp. 44–51.

[129] J. Rosen, B. Hannaford, C. G. Richards, and M. N. Sinanan, "Markov modeling of minimally invasive surgery based on tool/tissue interaction and force/torque signatures for evaluating surgical skills," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 5, pp. 579–591, 2001.

[130] J. Rosen, M. Solazzo, B. Hannaford, and M. Sinanan, "Task decomposition of laparoscopic surgery for objective evaluation of surgical residents' learning curve using hidden markov model," *Computer Aided Surgery*, vol. 7, no. 1, pp. 49–61, 2002.

[131] C. E. Reiley and G. D. Hager, "Task versus subtask surgical skill evaluation of robotic minimally invasive surgery," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI*. Springer, 2009, pp. 435–442.

[132] B. Varadarajan, C. Reiley, H. Lin, S. Khudanpur, and G. Hager, "Data-derived models for segmentation with application to surgical assessment and training," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI*. Springer, 2009, pp. 426–434.

[133] B. Varadarajan, *Learning and inference algorithms for dynamical system models of dextrous motion*. The Johns Hopkins University, 2011.

[134] J. J. Leong, M. Nicolaou, L. Atallah, G. P. Mylonas, A. W. Darzi, and G.-Z. Yang, "Hmm assessment of quality of movement trajectory in laparoscopic surgery," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006*. Springer, 2006, pp. 752–759.

[135] C. Loukas and E. Georgiou, "Surgical workflow analysis with gaussian mixture multivariate autoregressive (gmmar) models: a simulation study," *Computer Aided Surgery*, vol. 18, no. 3-4, pp. 47–62, 2013.

[136] L. Tao, E. Elhamifar, S. Khudanpur, G. D. Hager, and R. Vidal, "Sparse hidden markov models for surgical gesture classification and skill evaluation," in *Information Processing in Computer-Assisted Interventions*. Springer, 2012, pp. 167–177.

[137] L. Tao, L. Zappella, G. D. Hager, and R. Vidal, "Surgical gesture segmentation and recognition," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2013*. Springer, 2013, pp. 339–346.

[138] M. Lewicki and T. Sejnowski, "Learning overcomplete representations," *Neural computation*, vol. 12, no. 2, pp. 337–365, 2000.

[139] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5. IEEE, 1999, pp. 2443–2446.

[140] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[141] L. Zelnik-Manor, K. Rosenblum, and Y. C. Eldar, "Dictionary optimization for block-sparse representations," *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2386–2395, 2012.

[142] A. Agarwal, A. Anandkumar, P. Jain, P. Netrapalli, and R. Tandon, "Learning sparsely used overcomplete dictionaries via alternating minimization," *arXiv preprint arXiv:1310.7991*, 2013.

[143] C. E. Reiley, H. C. Lin, B. Varadarajan, B. Vagvolgyi, S. Khudanpur, D. Yuh, and G. Hager, "Automatic recognition of surgical motions using statistical modeling for capturing variability," *Studies in health technology and informatics*, vol. 132, p. 396, 2008.

[144] P. Van Overschee and B. De Moor, "Subspace identification for linear systems: Theory, implementation," *Methods*, 1996.

[145] R. Jenatton, J. Mairal, F. R. Bach, and G. R. Obozinski, "Proximal methods for sparse hierarchical dictionary learning," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 487–494.

[146] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *Signal Processing, IEEE Transactions on*, vol. 58, no. 6, pp. 3042–3054, 2010.

[147] E. Elhamifar and R. Vidal, "Block-sparse recovery via convex optimization," *IEEE Transactions on Signal Processing*, vol. 60, no. 8, pp. 4094–4107, 2012.

Vita

Shahin Sefati received his B. Sc. degree in Mechanical Engieering from Sharif University of Technology, Tehran, Iran, in 2008. He then joined the LIMBS (Locomotion in Mechanical and Biological Systems) lab at Johns Hopkins University to pursue his Ph. D. under the supervision of Prof. Noah J. Cowan. He received his M. Sc. in Mechanical Engineering from Johns Hopkins University in 2011. His research interests lie at the intersection of robotics, dynamical systems and control, machine learning and sparse representations.